The growing complexity of modern technical facilities and devices, technological lines and industrial systems together with the need for meeting high reliability and control quality requirements constitutes a challenge for many engineering fields, particularly automatic control, technical diagnostics, computer science, electrical metrology and electrical engineering. The aim of the present collective monograph is concise presentation of selected research methods that effectively solve problems related to metrology, process modelling, automatic control and diagnostics, digital systems design as well as industrial power electronics from the viewpoint of the requirements of modern technical processes and devices. The book is a result of research works carried out at the Faculty of Electrical Engineering, Computer Science and Telecommunications of the University of Zielona Góra over many years.

Wydawnictwa Komunikacji i Łączności recommends this book to PhD students and engineers interested in issues related to metrology, modelling, control systems design and diagnostics. The interdisciplinary content may be found useful in the context of both automatic control and computer science as well as electrical engineering and power electronics.

Wydawnictwa Komunikacji
i Łączności
www.wkl.com.pl

ISBN 978-83-206-1644-6

MEASUREMENTS • MODELS • SYSTEMS AND DESIGN

WKŁ

FACULTY OF ELECTRICAL ENGINEERING
COMPUTER SCIENCE
AND TELECOMMUNICATIONS

MEASUREMENTS MODELS SYSTEMS AND DESIGN

Edited by Józef Korbicz

WKŁ

The authors of particular chapters are with the University of Zielona Góra. They are employees of the four institutes of the Faculty of Electrical Engineering, Computer Science and Telecommunications:

- Institute of Computer Engineering and Electronics,
- Institute of Control and Computation Engineering,
- Institute of Electrical Engineering,
- Institute of Electrical Metrology.

Editor

Józef Korbicz has been a full-rank professor of automatic control at the University of Zielona Góra, Poland, since 1994. He currently heads the Institute of Control and Computation Engineering (ICCE). His present research interests include soft computing and analytical methods and their applications to fault detection and isolation (FDI). The primary aim of his research group is to contribute towards the diagnostics of dynamical systems. His research projects in this field have been financed by the Polish State Committee for Scientific Research/Ministry of Science and Higher Education, and by the European Commission within the 4th and the 5th Framework Programme.

Professor Korbicz founded the *International Journal of Applied Mathematics and Computer Science* (AMCS) and up to now he has been the Editor-in-Chief. He is a senior member of the IEEE, and a member of the IFAC TC on *SAFEPROCESS.* He was the IPC chairman of the IFAC Symposium on *SAFEPROCESS* held in Beijing, China, in 2006. Since 2003 he also has been the vice-chairman of the Committee on Automatic Control and Robotics of the Polish Academy of Sciences.

Monograph published
on the occasion of the 40th anniversary
of the Faculty of Electrical Engineering,
Computer Science and Telecommunications
UNIVERSITY OF ZIELONA GÓRA
1967–2007

MEASUREMENTS MODELS SYSTEMS AND DESIGN

To Friends and Partners
of the Faculty of Electrical Engineering,
Computer Science and Telecommunications
of the University of Zielona Góra

MEASUREMENTS MODELS SYSTEMS AND DESIGN

Edited by Józef Korbicz

Wydawnictwa Komunikacji i Łączności
Warszawa

Editor

Prof. Józef Korbicz
University of Zielona Góra
Institute of Control and Computation Engineering
ul. Podgórna 50, 65-246 Zielona Góra, Poland
e-mail: J.Korbicz@issi.uz.zgora.pl

Computer typesetting

Editorial Office of the International Journal of Applied Mathematics and Computer Science, AMCS
Beata Bukowiec

Financed by

University of Zielona Góra,
Faculty of Electrical Engineering, Computer Science and Telecommunications

ISBN 978-83-206-1644-6

Wydawnictwa Komunikacji i Łączności sp. z o.o.
http://www.wkl.com.pl, e-mail: wkl@wkl.com.pl

First edition. Warszawa 2007.

FOREWORD

For the past 40 years I have been observing with great pleasure and high regard the dynamic development of the Faculty of Electrical Engineering, Computer Science and Telecommunications as well as the entire University of Zielona Góra. The success of the Faculty is a result of the great commitment and effort of its staff, with whom I have been closely cooperating for many years. Some of them, including Prof. Józef Korbicz, Prof. Krzysztof Gałkowski and Prof. Marian Adamski – just to mention a few, have also been my friends. It is with great respect and admiration that I have been observing their success and involvement in the development of the Faculty and the entire University.

I have the great honour and pleasure to recommend to the reader the present valuable monograph, which summarises the Faculty's many years' research works and international cooperation within the *COPERNICUS* project, the 5th Framework Programme as well as research projects conducted together with French, German and English universities.

The monograph presents in a synthetic and original way selected methods of solving effectively problems related to metrology, process modelling, digital control systems design, diagnostics and power electronics.

In the first part of the monograph (Chapters 1–6), particular attention is focused on the design of measurement systems and circuits, the estimation of temporal parameters of distributed measurement and control systems, and the configuration of sensor networks.

The second part of the monograph (Chapters 6–9) is devoted to selected issues of automatic control and technical diagnostics, computational intelligence methods and their application to diagnostic systems. The use of fuzzy logic and artificial neural networks for modelling non-linear processes deserves particular attention.

The third part of the monograph (Chapters 10–15) presents in an interesting way the issues of information processing and digital systems design, digital image analysis and identification methods as well as optimal design of control systems.

The closing part of the monograph (Chapters 16–19) is concerned with power electronic converters of electrical current. A considerable amount of attention is paid to transition processes in power electronic feedback converters.

On the occasion of the 40th anniversary of the foundation of the Faculty, I would like to wish all of its staff members continuous success in their research and educational activities. May the coming years and decades be most successful for you! I also hope for a wonderful, dynamic development of the Faculty and the University of Zielona Góra for the sake of Polish academic research.

Warsaw, January 2007

Tadeusz KACZOREK
Honorary Doctor
of the University of Zielona Góra

MULTIFACETED DISCOURSE ON INFORMATION TECHNOLOGIES

1. Introduction

As a long-standing friend of the Faculty of Electrical Engineering, Computer Science and Telecommunications of the University of Zielona Góra, which in June 2007 celebrates the 40th anniversary of its foundation, I have the honour of adding a foreword to this great monograph published on the above-mentioned occasion and containing the most valuable research works of the Faculty's staff. I perceive this task as a reward and honour, although fulfilling it has turned out to be quite difficult, since the wide range of subject matters contained in the book considerably impedes finding a concept that could be exposed as being representative of the entire monograph. Pondering upon what links the four headwords composing the title of the book: *Measurements, Models, Systems and Design*, I have come to the conclusion that the "common denominator" searched for is the fact that both the title headwords and the content of the 19 chapters touch upon information technologies. Therefore, in this foreword, I take the liberty of presenting my own view on the content of the monograph as well as its main threads seen from the viewpoint of information technologies, being the leitmotif and a factor blending together this multiauthored and multithreaded research monograph.

Information technologies are presently one of the key issues in the development of contemporary engineering, although their role is by no means limited to that. Those technologies so deeply penetrate the social tissue of countries, nations, continents and the whole world that it is becoming more and more common to perceive the future society as *information society*. That will probably be the future of our politics, culture and civilization, although one must admit that the term is interpreted differently by different researchers, while various politicians, willingly exploiting the idea of *information society* as a trendy and support-providing synonym of progress and development, in much the same diversified fashion mark the path presumably leading to that somewhat mythical society of the future. In the present book, written by outstanding engineers and researchers representing technical sciences, the idea of information technology does not in fact appear overtly, but information as such – very much so. Therefore, – shall precede further deliberations on the content and value of the book with a brief discussion of the concept of information.

2. Information as an indispensable part of reality

Information is nowadays one of the most frequently used terms. Innumerable computers collect and process information, the Internet and other communications systems transfer it, while political scientists claim that in knowledge-based economy information will be a most valuable carrier, analogous to great estates in the feudal system and financial resources in capitalist economy. We are plied with information – not always of the highest quality – by the media, information – not always true – destabilizes authorities, various information services are the apple of somebody's eye or a thorn in somebody else's side. In a word – information is a trendy idea.

Yet if one tried to obtain a very reliable and precise definition of what information is, it would turn out that most of those who popularly use this term in every-day life were unable to provide such an explanation. What is more, although there have already been written many clever treaties on mutual relations between such concepts as data, information and knowledge, this area is by no means an unquestionable one. Even the Web's *Wikipedia*, renowned for precise and friendly (clear) definitions of various complicated ideas, provides us with a definition of information full of references to other notions which reads as follows: *Information as a concept bears a diversity of meanings, from everyday usage to technical settings. Generally speaking, the concept of information is closely related to notions of constraint, communication, control, data, form, instruction, knowledge, meaning, mental stimulus, pattern, perception, and representation.* This definition is neither easy nor "friendly", while that fact that it is further elaborated on throughout several pages of the encyclopedia certainly does not help to understand what information really **is**.

Of course, I will not even attempt proposing in this foreword any new definitions of such a difficult concept, as the task would be as much risky as it is pointless, but I will discuss a fact that often escapes the attention of both popular users of the idea of information and specialist on various types of information technologies. Namely, information is an indispensable and extremely important part of the world we live in, with or without computers, cellular phones, the Internet and other IT products. **Information** is an ingredient of existence that is as elementary and necessary as the surrounding **matter** and the **energy** that keeps it moving and transforming.

To illustrate the thesis I will use a figure from a book of mine[1], published 30 years ago but still by all means up-to-date (Fig. 1). Part *a* of the schema presents a diagram of a simple water turbine, which can use energy provided by water and conduct some particular useful work. If, however, there is no energy (in the form of flowing water), the movement will stop and the turbine will become useless, which is presented in Part *b* of the schema. Energy on its own, without the physical content, is useless as well – if there is no turbine, the energy of the flowing water cannot be used (Part *c*). From the diagram presented in Part *d* of the figure it is obvious that the studied device can be useless also in a situation when both the energy (water) and the physical content (turbine) are in place – the entire system will not be working

[1] Kulik C. and Tadeusiewicz R. (1974): *Elements of Economic Cybernetics.* — Course book of the Cracow University of Economics, (in Polish).

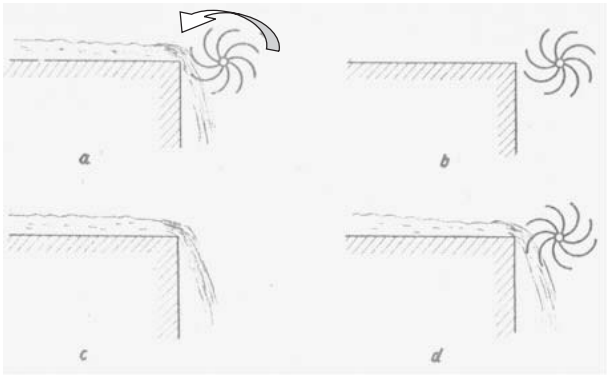


Fig. 1. Relations between matter, energy and information using the example of a simple engineering system (water turbine). Detailed discussion can be found in the text.

since the turbine is not mounted properly. The proper way of mounting it stands in this case for *information*. If it is missing, the physical and energy contents, although present, will not be able to fulfill their tasks.

3. Information as a common element of the published research results of the staff of the University of Zielona Góra

Moving from the general – although greatly simplified – deliberations, provided in the preceding part as a **digression**, on to the proper aim of the foreword, which is presenting and discussing in brief the content of the monograph, I would like to stress that the dominating part of information processes is a key linking and blending together those somehow thematically diversified works into one coherent piece. Let us look closely at what the authors focus on in particular chapters.

In the book, according to its title, one can distinguish at the beginning a cluster of works related to **metrology**. The issue can be perceived through details – one can, for example, analyse new measurement methods or new sensor and converter constructions, design measurement systems, improve measurement signal processing techniques, and scientifically examine all of those and other problems that are parts of widely understood measurement technology. Yet if wishing to approach all those issues in a more general way, one should first of all note that, carrying out any of the above-mentioned tasks, we will always be dealing in fact with an **information** process. Every measurement is a method of obtaining some important (quantitative or qualitative) information. Each signal processing algorithm is in fact aimed at isolating and clearing (from disturbances and noise expressing the omnipresent tendency for an increase in entropy) the components of useful information. Each analysis of measurement results and each decision-making process (e.g., related to diagnostics) on the basis of measurements and registration of various signals definitely have an informative character. Therefore, despite the multidirectionality of the works on measurements and methods of processing their results, collected in the book, we are

dealing here with a multifaceted discourse of many outstanding researchers, which is united by a drive to obtain reliable information.

Another title and content part of the discussed book is related to **modelling** methodology, which is one of the best-developed techniques for obtaining information on those features of analysed elements of reality that are not directly observable or measurable. When modelling various systems, we are always dealing with two existences with different physical and energy characteristics (I am intentionally relating here to the closing paragraphs of the previous section of the foreword). These existences, which constitute the task of modelling and the basis for model evaluation, are respectively the system under consideration (physical, technical, biological, social, economic, etc.) and its idealised abstract form known as the model. An original system bears many unique features and characteristics – for example, it holds a specific position in space and time, which may not be occupied by any other system. It possesses physical and energy components which cannot be duplicated without incurring considerable costs. None of those unique features of the original system can be found in its model, which can in turn have a symbolic character (notion model of a system without formalisation elements), take the form of abstract mathematical notations, or be implemented in a computer system as a simulation model. Despite, however, the fundamental differences between the original system and its model, there exists a link which makes the model a useful prognostic and inspection tool for the monitoring or control of the real system. This link between the model and the object of modelling is located in the very sphere of information. In a real object there are physical existences, while in a model – abstractive symbolic ones (including here also datasets in the registers of a digital machine simulating a given system). However, the **relations** between the physical existences in a real system and the abstractive ones in a model can be united with an isomorphic dependency due to which, from the viewpoint of **information**, there exists between the model and the system an affinity so close that the examination of the model can provide useful conclusions regarding the system.

The third thread in the discussed book is a widely understood notion of **systems**. As is well known, the term can be applied actually to any unique object since, according to the most general definition, a system is a set of interwoven elements. Therefore, this broad term can indicate both particular technical devices and living organisms or their pieces, as well as social structures, economic regularities or even philosophical concepts. Of course, depending on what systems we focus on, their form may differ, yet because the book considered discusses mainly systems of automatic control and technical diagnostics as well as information processing systems and electrical energy converters, we are dealing here again with a dominating information motif. The essence of processes found in the discussed systems consists in passing on or processing information, and most of the tools and methods considered contain an IT element. Therefore, the book recommended here has informative connotations in this context as well, which shows and proves again how extensive and multilayered a term **information** is.

And last, but not least, component of this rich book is related to the concept of **design**. The essence of design is abstractive creation of existences priorly absent and

conceived by the designer with their knowledge, imagination and – what is becoming increasingly important – the beneficial IT tools. After the design process the project is implemented, which means engaging various physical components: machinery, electronic parts, software, etc., but this is already project implementation and no longer the design process. At the design stage the final product is a concept, a concrete and detailed yet still abstract vision of the object being designed. The concept, vision, technical or methodological drawing, block diagram of the algorithm, specification of parts and materials and many other factors that we may employ during design are all **information pieces**. Thus the designer, regardless of what and how they are designing, is actually the creator of a new piece of **information**.

4. Closing remarks

The above discussion shows that although the recommended anniversary monograph of the Faculty of Electrical Engineering, Computer Science and Telecommunications of the University of Zielona Góra contains 19 independent works of separate teams, covering a wide spectrum of various research projects conducted at the Faculty, it does possess a uniting link. Namely, all of the book chapters relate to one concept and correspond to one subject matter by touching upon **information**. Because each author had perceived and described this fundamental concept from a different viewpoint and with reference to different needs, they eventually created a most valuable **multifaceted discourse on information**, or, more precisely, a discourse on information technologies. It is my belief that in times of knowledge-based economy, *information society* and omnipresent information technologies, the monograph is worth adding to our book collection to refer to it on various occasions, as it truly contains knowledge of highest order.

Cracow, January 2007

Ryszard TADEUSIEWICZ
Honorary Doctor
of the University of Zielona Góra
`http://www.agh.edu.pl/uczelnia/tad/`

PREFACE

The growing complexity of modern technical facilities and devices, technological lines and industrial systems together with the need for meeting high reliability and control quality requirements constitutes a challenge for many engineering fields, particularly automatic control, technical diagnostics, computer science, electrical metrology and electrical engineering. The aim of the present collective monograph is concise presentation of selected research methods that effectively solve problems related to metrology, process modelling, automatic control and diagnostics, digital systems design as well as industrial power electronics from the viewpoint of the requirements of modern technical processes and devices.

The monograph is composed of 19 chapters, which can be divided into several parts. The first one, comprising six chapters (1–6), is concerned with issues related to measurements and the design of measurement systems and circuits. This part discusses, among other things, the correction of distortions in input circuits of measurement systems, the design of voltage and current calibrators, the determination of temporal parameters of distributed measurement and control systems as well as advanced mathematical issues regarding the design of sensor network configurations for distributed parameter systems.

The second part (Chapters 6–9) presents selected problems of automatic control and technical diagnostics. Attention is mainly paid to intelligent computation methods and possibilities of their effective application, particularly in diagnostic systems. The possibility of employing fuzzy logic and artificial neural networks for the modelling of non-linear processes and their usage in fault detection systems are discussed. Taking into account the complexity of optimisation processes when designing diagnostic systems, evolutionary methods of solving those are presented. A separate chapter discusses multidimensional (nD) systems and repetitive processes together with selected methods of their analysis and control.

The subsequent part of the monograph is composed of Chapters 10–15, which discuss the issues of information processing and digital systems design. Methods of quantum information processing with application to cryptography as well as selected methods of digital image analysis and identification together with three-dimensional wavelet compression of visual sequences are studied. Separate chapters are devoted to the design of reconfigurable logic controllers implemented using modern FPGA

logic structures and hardware description languages. This part also discusses optimal design of control systems using programmable logic devices of the PAL and PLA types, and others.

The closing part of the monograph is concerned primarily with power electronic converters of electrical current. AC converters and research into transition processes in feedback power electronic converters are discussed, together with issues related to electromagnetic compatibility of systems with power electronic converters.

The book is a result of research works carried out at the Faculty of Electrical Engineering, Computer Science and Telecommunications of the University of Zielona Góra (up till 2001 – the Technical University of Zielona Góra) over many years. The authors are employees of the four institutes composing the Faculty: the Institute of Computer Engineering and Electronics, the Institute of Electrical Engineering, the Institute of Electrical Metrology, and the Institute of Control and Computation Engineering. The research and application results presented in the monograph were to a large extent obtained within numerous Faculty research projects financed by the Polish State Committee for Scientific Research/Ministry of Science and Higher Education in the years 1993-2006 as well as the *COPERNICUS* (1997–1999) and 5th Framework Programme projects financed by the European Union. Some results were obtained in the framework of international research projects within bilateral agreements with France (*POLONIUM* programme) and the UK (*British Council programme*). Other research projects were financed, for example, by the University of Hong Kong (2003–2004).

The book is recommended for PhD students and engineers interested in issues related to metrology, modelling, control systems design and diagnostics. The interdisciplinary content may be found useful in the context of both automatic control and computer science as well as electrical engineering and power electronics.

Zielona Góra, February 2007

Józef KORBICZ
`http://www.weit.uz.zgora.pl/users/J_Korbicz/`

Contents

Chapter 1

MEASUREMENT AND REPRODUCTION OF A COMPLEX VOLTAGE RATIO WITH THE APPLICATION OF DIGITAL SIGNAL PROCESSING ALGORITHMS

Ryszard RYBSKI*, Janusz KACZMAREK*

1.1. Introduction

The measurement and reproduction of the complex voltage ratio is one of the most important processes fundamental for the operation principles of devices and systems for the measurement of such electric quantities as power, phase angle or impedance. In the last years, the natural tendency connected with digital signal processing methods applied in measurements circuits has been observed. This results from, among other things, the accessibility of first-class sampling voltmeters, data acquisition cards, programmable generators and calibrators of sinusoidal signals, as well as the functional software on the market. Owing to that, it is possible to create, in a relatively simple way, measurement systems with good metrological properties (Bell, 1990; Callegaro and D'Elia, 2001; Ilic and Butorac, 2001; Kaczmarek and Rybski, 1995; Ramm *et al.*, 1999; Rybski and Kaczmarek, 1997). Research had been done in many scientific centers, whose results will permit to answer the fundamental question of to what extent the methods of digital signal processing can be applied in high-accuracy measurement circuits of the AC current, and to what degree they can make essential supplementing of accurate classical analog measurement circuits, as well as to what degree they can replace these circuits. Answers to the questions posed this way will have great practical meaning. Circuits performing digital signal processing algorithms are relatively easy to integrate. Hence, it will be possible to construct systems whose functions and properties will be changed in a significant degree by a change of the software. Circuits of this type can perform very different functions, from autonomous

* Institute of Electrical Metrology
e-mails: {r.rybski, j.kaczmarek}@ime.uz.zgora.pl

measuring devices, through data acquisition cards and intelligent nodes of distributed measurement systems, to signals conditioning circuits integrated with sensors.

At the beginning of the 1980s, there was a growing interest in the circuits in which DSP methods were applied to the measurement and reproduction of the complex voltage ratio. The application range of DSP methods is strictly connected withe the growing possibilities of their technical realization (e.g. by using A/D and D/A converters with high resolution, signal processors, etc.). The rapid development of equipment possibilities accompanies the development of new DSP algorithms and methods, as well as the extension of the application area of the measurements of electric and nonelectric quantities. For example, specialized high accuracy measurement systems based on digital signal generation and sampling method are designed in the area connected to the calibration of measurement instruments of impedance, power and phase angle. In the future, these systems are supposed to enable the remote calibration of devices at a user's place via the Internet (the so-called *e-calibration*).

Simultaneously, a growing area of application in digital measurements of the complex voltages ratio, which are not connected to the standards and calibration, e.g. for determining the frequency characteristics of functional blocks of measurement circuits (amplifiers, current-to-voltage transducers (Locci and Muscas, 2001; Sasdelli *et al.*, 1998)) and in measurements of nonelectric quantities (e.g. measurements of linear displacement by using transformer sensors (Crescini *et al.*, 1998; Rybski and Krajewski, 2003)) is observed.

For many years, the authors have been into both the theoretical and experimental research oriented towards accurate measurements of electric quantities with DSP algorithms. Among others, there are the following research topics: methods and circuits for the reproduction and measurement of the complex voltage ratio, the calibration of methods and circuits, as well as their application in impedance comparators and AC power calibrators. Some research results from the above area are presented in this chapter.

1.2. Digital sine-wave sources for the reproduction of the complex voltage ratio

1.2.1. Complex voltage ratio

The complex ratio of two AC voltages can be reproduced in AC current measurement circuits by using the impedance voltage divider (Fig. 1.1(a)), the inductive voltage divider (Fig. 1.1(b)), two voltage sources connected in series (Fig.1.1c), or the voltage instrumentation amplifier (Fig. 1.1(d)) (Skubis, 1995). Figure 1.1(e) shows two digital voltage sine-wave sources, DSVS1and DSVS2, connected in series, which provide the voltage, $\underline{V}_1$ and $\underline{V}_2$ with adjustment possibilities of the amplitude, frequency and initial phase. A case when $f_1 = f_2 = f$ is exclusively analyzed in this paper. Moreover, it is assumed that the internal impedance of sources is negligibly small.

The complex voltage ratio on the outputs of the sources is equal to the ratio of their complex amplitudes $\underline{V}_1$ and $\underline{V}_2$, and is given by means of the equation

$$\underline{K}_V = \frac{\underline{V}_1}{\underline{V}_2}. \tag{1.1}$$

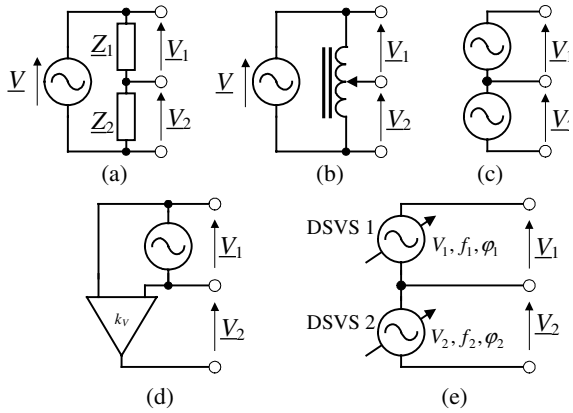


Fig. 1.1. Circuits reproducing the complex voltages ratio: (a) impedance voltage divider, (b) inductive voltage divider, (c) two voltage sources connected in series, (d) voltage instrumentation amplifier, (e) two digital voltage sources connected in series

It is possible to show the complex voltage ratio in the form

$$\underline{K}_V = \frac{\underline{V}_1}{\underline{V}_2} = \frac{V_1 e^{j\varphi_1}}{V_2 e^{j\varphi_2}} = \frac{V_1}{V_2} e^{j(\varphi_1 - \varphi_2)} = K_V e^{j\varphi_V}, \tag{1.2}$$

where $\underline{V}_1$ and $\underline{V}_2$ are amplitudes of voltages $\underline{V}_1$ and $\underline{V}_2$, φ_1, φ_2 represent the initial phase angle of the voltages $\underline{V}_1$ and $\underline{V}_2$, K_V is the magnitude of the complex voltage ratio, and φ_V is the argument of the complex voltage ratio.

1.2.2. Digital sine-wave sources

1.2.2.1. Sinusoidal voltage generation based on direct digital synthesis techniques

The most frequently applied methods of frequency synthesis are as follows:

- direct synthesis – four fundamental operations: multiplication, division, addition and subtraction are used in this method; the tasks are carried out by the quartz oscillator on the frequency of a generated signal.
- indirect synthesis – synchronized oscillators with the application of the synchronizing phase loop as well as programmed frequency dividers are used in this method.
- digital frequency synthesis, also known as direct digital synthesis, which can be characterized in the following way:
  - the frequency of the output signal is exclusively determined by mathematical processing (binary operations) and clock impulses from a reference quartz generator,
  - the generated sinusoidal signal is given, in the preliminary stage of the synthesis, in the form of the sequence of binary numbers (samples),
  - the sequence of binary numbers is converted into the analog form in the next stage of the synthesis.

In the digital method of frequency synthesis, a time continuous function $x(t)$ is determined in the range of time $0 \leq t < T$, the root is divided into a finite number of N identical time periods (Fig. 1.2). Referring to the beginning of each range Δt, a value of the function $x(t)$ is assigned to the entire range (as a sample) $x(n\Delta t)$.

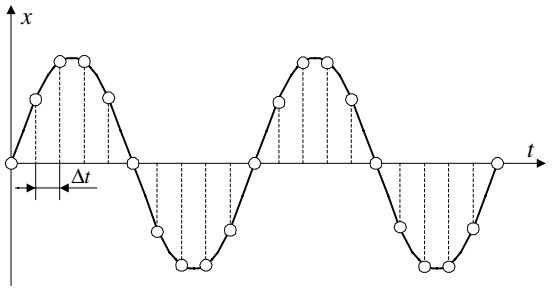


Fig. 1.2. Uniformly sampled sinusoidal function

It is possible to assign the phase increment $\omega\Delta t$ to each time period Δt and then, in reference to the harmonic signals, it is possible to show the function $x(n\Delta t)$ in the form

$$x_1\left(n\Delta t\right) = X\sin\left(\omega n\Delta t + \phi\right), \tag{1.3}$$

as well as

$$x_2\left(n\Delta t\right) = X\cos\left(\omega n\Delta t + \phi\right), \tag{1.4}$$

where X, ω and ϕ are signal parameters: the amplitude, frequency (radian per second) and initial phase angle, respectively.

Using Euler's equation and assuming that $X = 1$, $\phi = 0$, it is possible to write the functions (1.3) and (1.4) in a different form:

$$\sin\left(\omega n\Delta t\right) = \mathrm{Im}\left[e^{j\omega n\Delta t}\right], \tag{1.5}$$

$$\cos\left(\omega n\Delta t\right) = \mathrm{Re}\left[e^{j\omega n\Delta t}\right]. \tag{1.6}$$

Assuming that the frequency f is determined by the relation

$$f = \frac{1}{N\Delta t}, \tag{1.7}$$

the function

$$e^{j\omega n\Delta t} = e^{j\frac{2\pi}{N}n} \tag{1.8}$$

represents the complex amplitude of the sinusoidal signal with the unity amplitude and zero initial phase, and it can be shown on a complex plain in the unity circle form (Fig. 1.3).

Using the vector graph (Fig. 1.3) and taking the earlier assumption of the constancy of function in every of n time intervals, it is possible to construct the timing diagram of the function (1.8). The diagram is presented in Fig. 1.4.

Digital synthesis of frequency assumes that a binary number referring to the value of a sampled sinusoidal function is assigned to each time interval. Through periodical

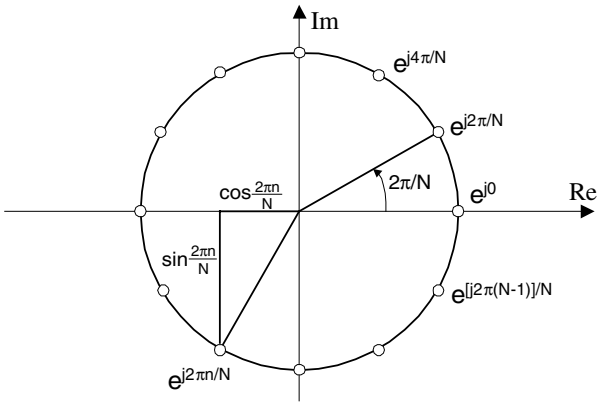


Fig. 1.3. Graphical representation of a complex sinusoidal signal

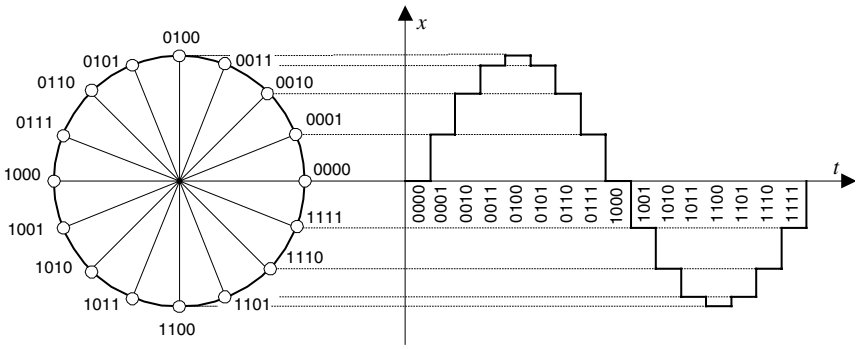


Fig. 1.4. Applying digital frequency synthesis to create a sinusoidal signal

repetition of their values, the stepwise function, approximating the desired sine-wave, is yielded.

In the practical realization of the digital frequency synthesis method, the values of samples of the sinusoidal function are most often put into the semiconductor memory. The speed of reading the samples from the memory (angle speed of the vector in the Fig. 1.3) decides about the frequency of the generated signal. A digital sinusoidal signal on the memory output is converted into the analog form (voltage signal) via a digital-to-analog converter (Fig. 1.5).

The task of the phase quantization circuit relies on the way of changes in the argument of the sinusoidal function (phase increment) according to the algorithm applied, which is appropriate for the described method of synthesis.

The signal containing information about the phase of the generated signal and changing to the beat of clock impulses with the frequency f_C is processed by the converter phase/amplitude to the digital sinusoidal signal. In most cases, the semiconductor memory is used as the phase/amplitude converter. In such a case, the output signal of the phase quantization circuit is used to address storage cells that contain appropriate values of the *sine* function. Specialized integrated circuits containing in their

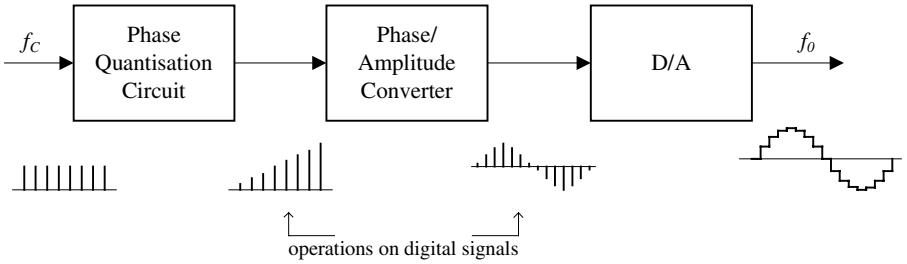


Fig. 1.5. Principles of operation of the voltage source with digital frequency synthesis

structure the phase quantization circuit and the phase/amplitude converter (DDS integrated circuits) have been accessible on the market for some time. They permit the generation of the sinusoidal voltage in the form of a sequence of digital words representing consecutive values of the sinusoidal function. Additionally, the complete integrated programmable generators of the voltage sine-wave (with any shape of the waveform on the generator output), which the method of the digital frequency synthesis is based on, are produced. The complete realization of the operations carried out by the phase quantization circuit and the phase/amplitude converter is also possible by means of software. In this case, the DSP processor carries out all necessary tasks. The high computational efficiency of digital signal processors makes it possible to calculate the value of the sample of the sinusoidal function in real time.

Two essential methods of direct digital frequency synthesis on account of determining the way of the incremental phase and the way of the phase quantization circuit solution are distinguished (Ciglaric *et al.*, 2002; Lapuh and Svetik, 1997):

- direct frequency division method,
- phase accumulation method.

Method of direct frequency division. An address counter acts as the phase quantization circuit in the method of direct frequency division (Fig. 1.6).

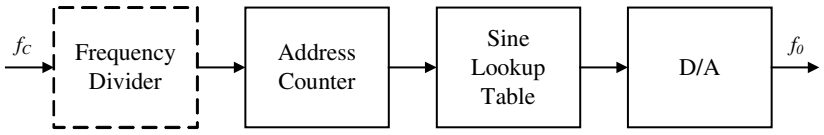


Fig. 1.6. Block diagram of the voltage source based on the method of direct frequency dividing

The size of the address counter, equal to the number M of address bits of the memory that contains the sine function values, determines the number of samples by the period. The frequency of the output signal is calculated from the formula

$$f_0 = \frac{f_C}{M}. \tag{1.9}$$

For a given value of M, it is possible to change the frequency of the output signal by changing the frequency of clock impulses. A divider of the frequency usually

applies in this case. A constant number of samples – independently of the frequency of the generated signal – in the period is a characteristic feature of this method. Hence, the method is called sometimes the waveform with the *Constant Number of Samples* (CNS) per period.

Method of phase accumulation. The principle of operation of the voltage source, to which the method of the phase accumulation was applied, is shown in Fig. 1.7. The change in the argument of the generated function is being carried out by the phase accumulator. Clock impulses with the f_C frequency cause cyclic increment, by a certain value of F_A of the digital F word entered in a frequency register, of the contents of the L-bits register-accumulator.

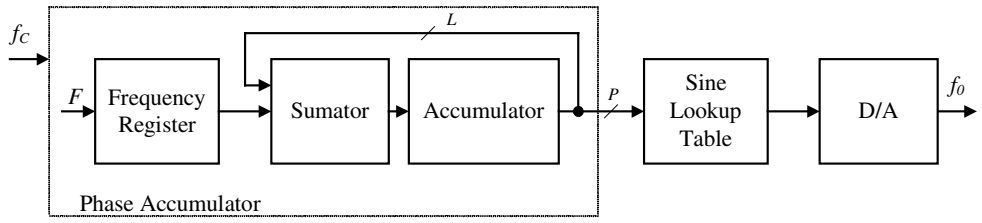


Fig. 1.7. Block diagram of the voltage source based on the method of phase accumulation

Each clock impulse causes a change in the state of the phase accumulator by the value

$$\Delta\phi = F_A \Delta\phi_{\min} = F_A \frac{2\pi}{2^L}, \tag{1.10}$$

where $\Delta\phi$ is the phase increment of the generated signal responding to the F_A value of the digital word, $\Delta\phi_{\min}$ is the lowest possible increment of phase of the generated signal, L is the size of the phase accumulator. The most significant P bits of the phase accumulator are used to address the memory containing samples of the sine function.

The frequency of the generated output signal equals

$$f_0 = F_A \frac{f_C}{2^L}. \tag{1.11}$$

A constant value of the sampling impulse duration of each sample is a characteristic feature of the phase accumulation method. Hence, the method is called also the waveform with *Constant SamplingTime* (CST).

Sinusoidal voltage generation by the digital frequency synthesis method is an operation as a result of which a stepwise waveform is produced approximating the sine-wave. The desirable quality of approximation depends on the required metrological parameters resulting from the expected application. On the other hand, the choice of the method of digital synthesis, suitable selection of the characteristic for its parameters, especially the number of samples in the period as well as the accuracy of their digital representation and, finally, the parameters of the electronic circuits applied in the practical realization, are decisive for the quality of generated voltage.

In digital sources based on digital synthesis, the fundamental advantages are very high resolution of the settings of the frequency (about μHz) and phase (below 0.01°),

good frequency, phase and amplitude stabilities, digital control of the basic parameters of the generated signal, the possibility of synchronising the generated voltage signal with signals controlling the operation of different devices and measurement circuits, easiness of sine-wave generation with very small frequencies. Limitations in the application of the digital synthesis method result, above all, from the presence of a higher harmonic in the spectrum of the generated signal and relatively small value of its maximum frequency. However, the weight of these limitations can be different depending on the destination of the generator. It is possible, particularly in the range of lower frequencies, to generate voltage signals with small distortions below 0.01%. It is also possible, due to the very dynamic progress in the technology of DDS integrated circuits as well as fast and accurate DAC converters, to generate the voltage in the range of hundreds MHz.

Digital sources of the sine-wave voltage taken to reproduce the complex voltage ratio should be characterized by high accuracy and resolution of the settings of the amplitude, phase and frequencies. Moreover high time stability of the quantities listed above is necessary. The generated voltage signal should have a low level of higher harmonics and noise.

1.2.2.2. Accuracy of digital sources of the sinusoidal voltage

Modeling the process of creating, from N equal intervals (samples) by period, the stepwise waveform approximating sinusoidal function (Fig. 1.8(a)), it is possible to use the circuit consisting of the impulse generator and zero-order extrapolator (Fig. 1.8(b)).

The sinusoidal signal, continuous in time,

$$x(t) = X_m \sin(\omega_0 t), \tag{1.12}$$

of the frequency f_0 is sampled, in the impulse generator circuit, with the sampling frequency f_S. A impulse sampled signal $x_D(t)$, which represents the samples $x(nT_S)$ of the signal $x(t)$, is yielded as a result of the sampling. The zero-order extrapolator turns the impulse signal $x_D(t)$ into the stepwise signal $x_{ST}(t)$.

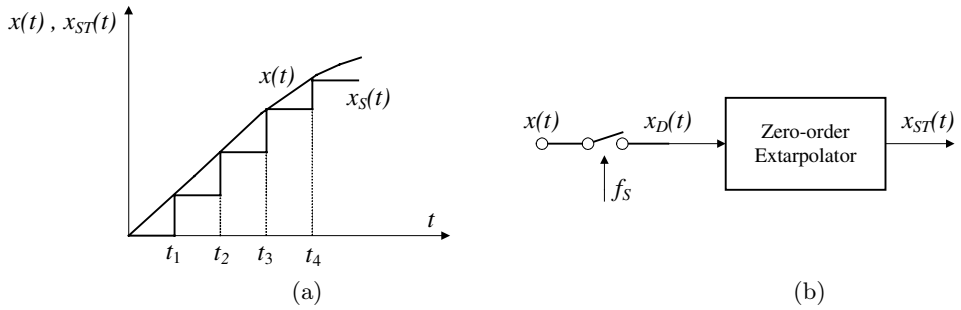


Fig. 1.8. Model of stepwise signal generation (a) timing waveform, (b) schematic block diagram

The spectrum of the output signal comprises the fundamental frequency f_0 and the higher harmonic frequencies $f = kf_C \pm f_0$, where $k = 1, 2, \ldots$, whose amplitudes are

determined as follows:

$$X\left(f_0\right) = X_m \mathrm{sinc}\left(\pi \frac{f_0}{f_S}\right), \tag{1.13}$$

$$X\left(kf_S \pm f_0\right) = X_m \mathrm{sinc}\left[\pi\left(k \pm \frac{f_0}{f_S}\right)\right]. \tag{1.14}$$

It is possible to show the amplitudes of spectrum components in the form making it possible to easily estimate the influence of the number of samples N of the sinusoidal signal (falling on one period of the reproducing sine-wave) on their values:

$$X\left(f_0\right) = X_m \mathrm{sinc}\left(\frac{\pi}{N}\right), \tag{1.15}$$

$$X\left(kf_S \pm f_0\right) = X_m \mathrm{sinc}\left[\pi\left(k \pm \frac{1}{N}\right)\right]. \tag{1.16}$$

The amplitude spectrum of the stepwise waveform is shown in Fig. 1.9.

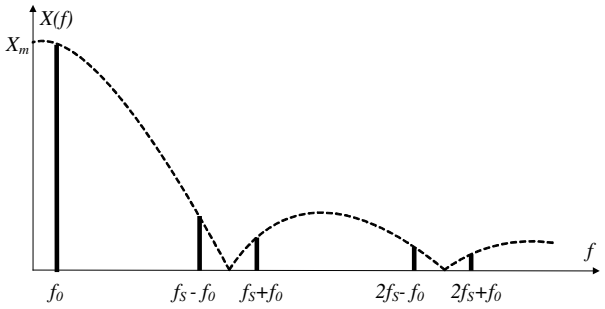


Fig. 1.9. Amplitude spectrum of the stepwise waveform

The relation which was determined by means of the equation (1.15) between the amplitude error of the fundamental harmonic and the number of samples in the period is shown in the Fig. 1.10.

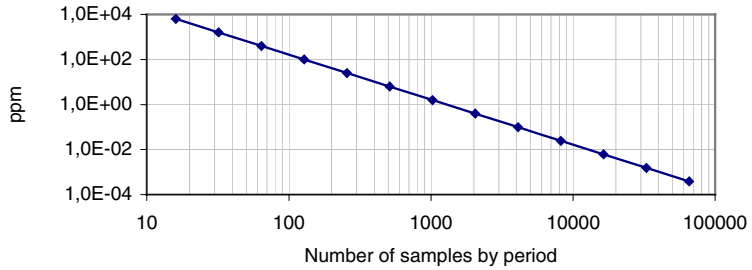


Fig. 1.10. Amplitude error of the fundamental harmonic from the number of samples in the period of the generated signal

The influence of the quantization error is described in detail in the literature from a statistical perspective. It is assumed that the quantization error e accompanying

the quantization process makes up the kind of noise whose value fits in the range $\pm(1/2)q$, where q is the range of the quantization. Establishing uniform distribution for the density function $p(e)$ of the probability of the quantization error, the variance is equal to

$$\sigma^2 = \int_{-q/2}^{q/2} e^2 p(e)\,\mathrm{d}e = \frac{q^2}{12}. \tag{1.17}$$

In the circuit of digital synthesis with a K bits D/A converter, for the reproduction of the sine-wave, the ratio of signal to quantization noise equals

$$\frac{S}{N} = (6.02K + 1.76)\ dB. \tag{1.18}$$

The process quantization introduces additional components in the spectrum of the digitally generated waveform, whose frequencies are an integral multiple of the fundamental harmonic frequency. The influence of these harmonics is not always essential. For example, in impedance measurement circuits with a selective detector, the spectrum frequency range around the fundamental harmonic is most interesting. The SNR relation as a function of the K number of bits is shown graphically in Fig 1.11, and quantization influence on the amplitude error of the fundamental harmonic determined by a computer simulation is shown in Fig. 1.12.

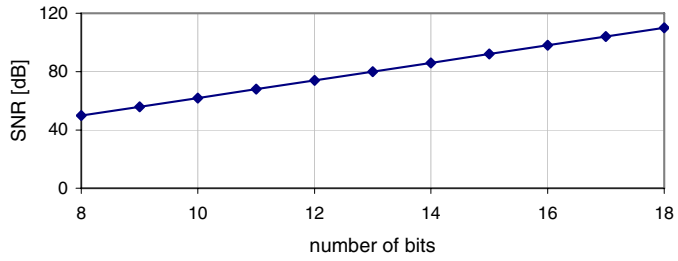


Fig. 1.11. Signal to noise ratio caused by the quantization process

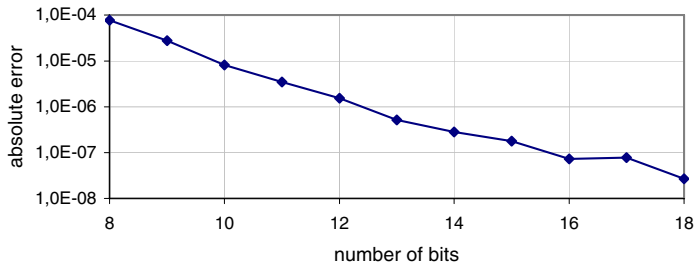


Fig. 1.12. Amplitude error of the fundamental harmonic caused by quantization, $N = 65536$

Apart from the above analysis, the number of samples in the period and a quantization error, the following parameters, among others, also have an influence on the quality of the digitally generated stepwise waveform: spike impulses, oscillations from

dynamic properties of the D/A converter, appearing on the output, jitter and static parameters of the D/A converter. A detailed analysis of their influence exceeds the scope of the presented study; however, it is possible to find in (Rybski, 2004) appropriate information on the subject.

1.3. Complex voltage measurement using the discrete Fourier transform

The determination of the voltage ratio of two AC current voltages with great accuracy is among, other things, indispensable in accurate measurements of impedances, power and ratios of precise voltage dividers and measuring transformers. Now, for measuring the AC voltage ratio, sampling methods with DSP algorithms are used more often and successfully.

The endeavor to increase measurement accuracy requires minimizing the error related to the sampling and quantization of signals. Very promising results were obtained in measuring systems in which sampled signals were produced with the application of the direct digital synthesis method, at the same time ensuring full synchronization of digital sources of signals and sampling schemes (Kürten Ihlenfeld *et al.*, 2003; Ramm *et al.*, 1999; Ramm and Moser, 2001; 2003). The results of work in this area indicate the possibility of measuring complex voltage ratios in the range of a low frequency (from about 1 kHz), with uncertainty at the level of 1×10^{-6}.

For many years at the Institute of Electrical Metrology works related to the application of the sampling method and also direct digital synthesis in systems for impedance measurements have been conducted. Among others, systems based on commercial data acquisition cards and permitting the comparison of impedance components with the uncertainty of 1×10^{-5} (Rybski and Kaczmarek, 2000; 2001; 2002) were developed. The increase in accuracy in these systems requires the application of high-resolution A/D converters simultaneously omitting sample-and-hold circuits that introduce additional errors. The application of commercial sampling voltmeters in the measurement system (e.g. HP3458A type) with the so-called integrative sampling mode created new capabilities (Kampik *et al.*, 2000; Muciek J. and Muciek A., 1999; Pogliano, 2002; 2006). The values of samples appointed in the integrative sampling mode are equal to the averages of the sampled signal in time equal to the integration time. The programming capability of the integration time allows exerting an effect on the accuracy of the analog-to-digital conversion process.

The concept of the measurement of a complex voltage ratio with the employment of integrative sampling and the discrete Fourier transform is also presented. Main sources of the errors of measurements were analyzed, particularly taking into consideration the consequences of non-ideal synchronization of sampled and sampling signals.

1.3.1. Sampling method for the measurement of a complex voltage ratio

As a result of using integral sampling in order to acquire the periodic signal $v(t)$ with the period T and the limited frequency band at f_g (Fig. 1.13), the following series of

samples is obtained:

$$v\left(kT_S\right) = v\left(k\right) = \frac{1}{T_I} \int\limits_{kT_S}^{kT_S+T_I} v\left(t\right) \mathrm{d}t, \quad k = 0, 1, \ldots, N-1, \tag{1.19}$$

where k is the number of the sample, N is the number of samples in the period T, T_S is the sampling period, T_I is the integration time.

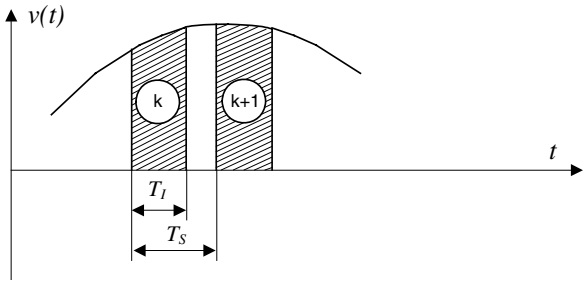


Fig. 1.13. Illustration of integrative sampling

It is assumed that sampling is synchronous, which means that

$$N \cdot T_S = T = \frac{1}{f}. \tag{1.20}$$

The discrete Fourier transform of a signal given in the form of a series of samples $v(k)$ is equal to

$$DFT\left(m\,f\right) = DFT(m) = \sum_{k=0}^{N-1} v\left(k\right) e^{-j\frac{2\pi}{N}k \cdot m}, \tag{1.21}$$

where $m = 0, 1, \ldots, N-1$ means the number of the harmonic of the frequency spectrum of the analyzed signal.

In order to determine the spectrum of a signal $v(t)$ from the relation (1.21), it is necessary to introduce a correction coefficient $K(m)$ (Pogliano, 1997) that results from a finite sampling time. Hence, the relation describing the spectrum can be expressed as follows:

$$V\left(m\right) = \frac{1}{K\left(m\right)} DFT\left(m\right), \tag{1.22}$$

where $K(m)$ is defined as

$$K(m) = \frac{\sin\left(\pi T_I f\, m\right)}{\pi T_I f\, m} e^{j\pi T_I f\, m}, \tag{1.23}$$

and for given m takes on a constant value which can be determined provided that the integration time T_I and the frequency f of the analyzed signal are known.

Taking into consideration the equations (1.21)–(1.23), the frequency spectrum of a signal $v(t)$ can be written as

$$V\left(m\right) = \frac{\pi T_I f\, m}{\sin\left(\pi T_I f\, m\right)} e^{-j\pi T_I f\, m} \;\; e^{-j\frac{2\pi}{N}k\, m}. \tag{1.24}$$

In accurate measurements carried out in alternating current circuits (i.e. impedance measurements, ratio measurements of inductive voltage dividers and measuring transformers), the complex ratio is determined only for fundamental harmonics. Hence, restricting further deliberations to the fundamental harmonic of the signal $v(t)$, from relation (1.24) for $m = 1$ we obtain:

$$V(1) = \frac{\pi T_I f}{\sin(\pi T_I f)} e^{-j\pi T_I f} \sum_{k=0}^{N-1} v(k)\, e^{-j\frac{2\pi}{N}k}. \tag{1.25}$$

The fundamental harmonic calculated from the relation (1.25) can be written as follows:

$$V(1) = \mathrm{Re}[V(1)] + j\mathrm{Im}[V(1)]. \tag{1.26}$$

Further, it is assumed that signals determined from their voltage ratio are defined as

$$v_1(t) = V_{1m} \sin(2\pi f t), \tag{1.27}$$

$$v_2(t) = V_{2m} \sin(2\pi f t + \varphi). \tag{1.28}$$

Taking the procedure outlined above into consideration, the following relations are determined, which makes it possible to calculate the real and imaginary parts of the complex voltage ratio of the signals $v_1(t)$ and $v_2(t)$:

$$\underline{K}_V = \frac{\mathrm{Re}[V_1(1)] + j\mathrm{Im}[V_1(1)]}{\mathrm{Re}[V_2(1)] + j\mathrm{Im}[V_2(1)]} = A + jB, \tag{1.29}$$

also, based on integrative samples and the DFT algorithm, the magnitude and argument are given respectively as

$$K_V = \sqrt{(A)^2 + (B)^2}, \tag{1.30}$$

$$\varphi = \arctan\left(\frac{B}{A}\right). \tag{1.31}$$

1.3.2. Error sources of complex voltage ratio measurement by the sampling method

The main error sources include the synchronization error, the influence of higher harmonics, the influence of voltmeter uncertainty. The analysis of the synchronization error is shown below. In (Rybski *et al.*, 2004), the remained sources of errors were investigated in detail.

One of the conditions of proper determination of frequency spectrum components with the employment of the discrete Fourier transform is synchronous sampling, boiling down to the implementation of a condition expressed with the relation (1.20). For accurate measurements of voltage ratios, ensuring a negligible synchronization error defined as the difference between the real frequency of signal sampling and its value resulting from the relation (1.20) constitutes a necessary condition to obtain a result

of the measurement with acceptable uncertainty. The most advantageous solution in this situation would be clocked with the same clock signal of the sampling voltmeter (precisely – the A/C converter employed in the voltmeter) and also the source (or sources) of sinusoidal signals used in the measurement system. Such possibility exists when digitally synthesized sinusoidal voltage sources are employed, and the sampling voltmeter has appropriate outputs with synchronizing signals. In that case, the influence of the synchronization error on the accuracy of the measurement of a complex voltage ratio is very small (below 1×10^{-6} (Kürten Ihlenfeld *et al.*, 2003)). However, it remains difficult to remove the uncertainty evoked by fluctuations (jitter) of the system clock generator. If there are commercial instruments employed in the system, then the assurance of hardware synchronization is not always possible. Then, a sinusoidal voltage generator with high resolution and accuracy of frequency setting should be used. An additional influence of synchronization error occurs if both sinusoidal signals are not sampled simultaneously. This situation takes place if one sampling voltmeter is used to measure the voltage ratio, and the sampled signals are connected to its input in order (sequentially) with the assistance of the controlled switch (Fig. 1.14(b)). In the application of sequential sampling, an account should be taken of an additional component of the systematic error of voltage ratio measurement. This is evoked by the synchronization error and its value can be many times greater in comparison with the error that will stand out in the system with simultaneous sampling (Fig. 1.14(a)).

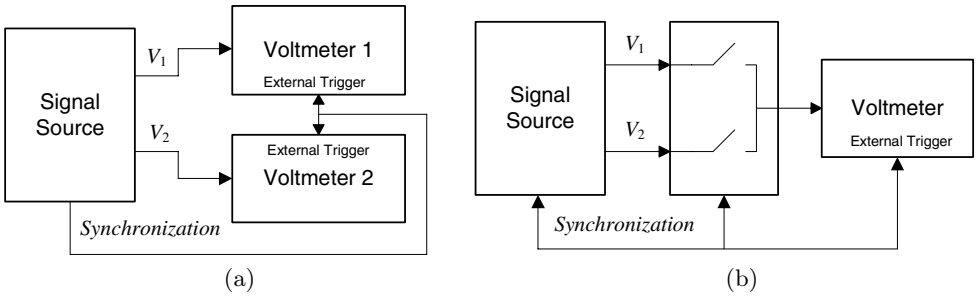
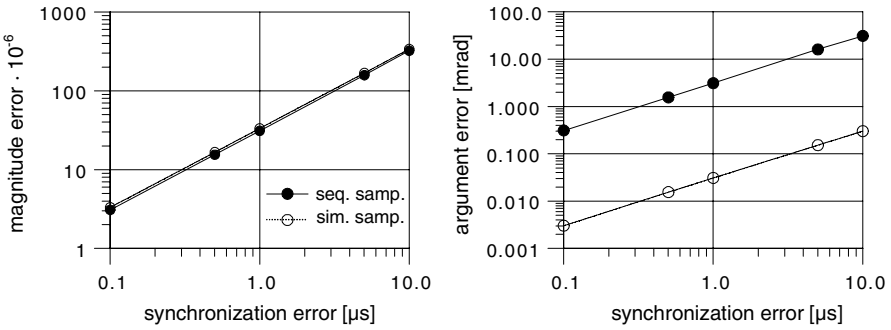


Fig. 1.14. Complex voltage ratio measurement using the sampling method: (a) simultaneous sampling, (b) sequential sampling

In order to research the influence of the synchronization error on the accuracy of magnitude and argument evaluation from the complex voltage ratio, computer simulations were carried out. A range of experiments was carried out during which, among other things, the integration time T_I, the number of samples in the signal period N (frequency of sampling), the number of sampled periods, equally for simultaneous sampling as well as for sequential sampling, were selected. When establishing simulation conditions, the parameters of the sampling voltmeter HP3458A were also taken into consideration. Examples of the research results are presented in Figs. 1.15(a) and 1.15(b).

The results of the research presented in Fig. 1.15 indicate that the argument error determined with the sequential sampling method might assume very large values. To ensure magnitude and argument errors on the level of 1×10^{-6}, it is necessary that



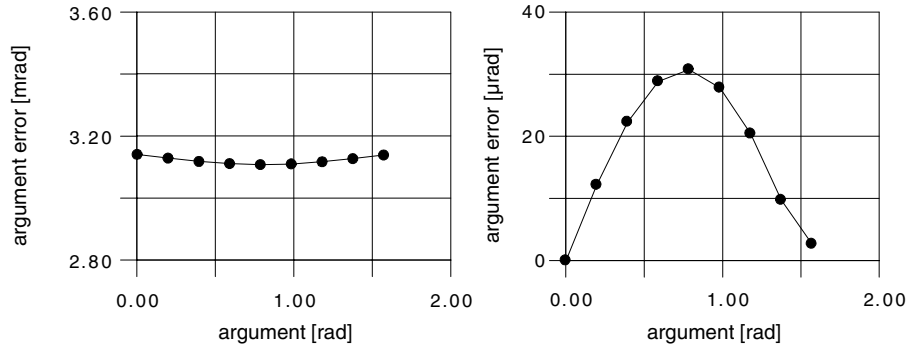
(a) (b)

Fig. 1.15. Influence of the synchronization error on the determination of the accuracy of the complex voltage ratio: (a) magnitude error, (b) argument error

the absolute synchronization error be smaller than 0.1 μs (in this case it is related to a relative error of 6×10^{-6}).

The dependence of absolute magnitude and the argument error of the complex voltage ratio on the absolute error of synchronization is shown in Fig. 1.15. The results presented in this figure were calculated for the following conditions: signal period $T = 16$ ms, $T_S = 1$ ms, $N = 16$, time of collecting samples $= 8 \times T$, $T_I = 0.9$ ms, $K_{Vn} = 1$, $\varphi_n = \pi/4$ (where K_{Vn} and φ_n are the nominal values of the magnitude and argument of the complex voltage ratio).

For the above-described conditions and for a constant value of the absolute synchronization error $\Delta T = 1$ μs, the dependence of the argument error of the complex voltage ratio as a function of the set up value of argument was also determined (Fig. 1.16).



(a) (b)

Fig. 1.16. Influence of the synchronization error on argument determination: (a) sequential sampling, (b) simultaneous sampling

The simulation results for sequential sampling were experimentally verified in a measuring system whose simplified diagram is presented in Fig. 1.17. The measuring

system consists of a sampling voltmeter, a signal switch and two commercial voltage generators which, based on the direct digital synthesis method, work in a synchronous mode.

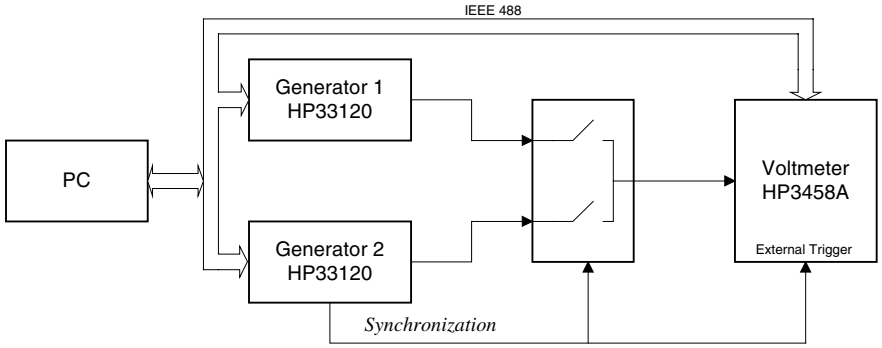


Fig. 1.17. Simplified diagram of the measurement system with a sampling voltmeter

The synchronization signal from a reference generator (Generator 2) is used to control the signal switch and also to trigger the measurement cycle of the sampling voltmeter. In this case, the voltmeter runs in DCV mode, without a sample-hold circuit at its input. Properly integrated software, prepared in the LabWindows/CVI environment, assures attendance of the measuring system.

Example research results that have shown the influence of synchronization errors on the accuracy of argument measurement of a complex voltage ratio are presented in Fig. 1.18. The measurement results were presented together with results obtained from computer simulation tests. Measurements and simulations were carried out for identical conditions as in the case of the results shown in Figs. 1.15 and 1.16.

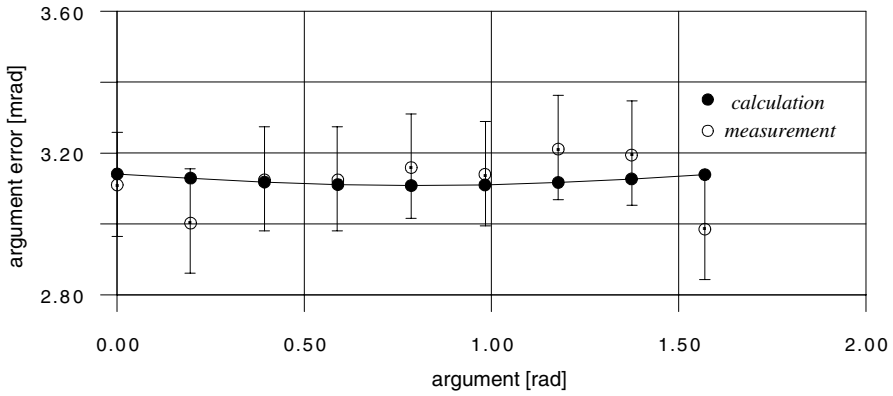


Fig. 1.18. Influence of the synchronization error on the accuracy of argument determination in the case of sequential sampling – the comparison of the calculation and measurement results

The uncertainty limitations of the measurements of the argument error are also marked in Fig. 1.18. The resolution of the phase meter employed in the measurement system (omitted on the diagram) has a decisive influence on the estimated value of the uncertainty. The phase meter was applied to ensure a constant difference of phase angles between output signals of the generators during changes of frequency settings. This is necessary because the signals from the generators HP33120 (used in the system) have an unknown value of the difference of phase angels after power on or after changes of frequency settings.

The influence of the above-mentioned uncertainty sources might be in a significant way decreased through the ensuring of the synchronization error with an appropriately low value and the application of voltage sources with high spectrum purity. The influence of uncertainty components of the voltmeter (derived from its metrological specifications) on the accuracy of integrative sample measurement and full measurement uncertainty of a complex voltage ratio requires greater attention.

1.4. Application examples of circuits for the measurement and reproduction of the complex voltage ratio

1.4.1. Impedance bridge with two voltage sources

Classic high-precision impedance measurement circuits are based on the use of measurement transformers, Inductive Voltage Dividers (IVD) and current comparators. For many years research has been conducted on high-precision impedance measurement circuits, in which methods of digital signal processing are used (Bohaček, 2004; Callegaro *et al.*, 2001; Corney, 2003; Helbach *et al.*, 1983; Muciek, 1997; Rybski and Kaczmarek, 2001). This tendency follows from, among other things, the availability of top class sampling voltmeters on the market, data acquisition cards, programmable sine-wave generators that are used for the reproduction and measurement of the complex voltage ratio – the operation underlying the functioning of impedance measurement devices and circuits.

Good metrological properties, as well as the simplicity of software-based adjustment of the parameters of the voltage generated by the frequency synthesis method, have contributed to the development of impedance circuits with digital sine-wave sources. The accuracy of impedance measurement in bridge circuits with digital sine-wave sources depends on the accuracy of the determination of the complex voltage ratio reproduced by these sources. In the solutions presented in the literature, a tendency prevails consisting in the use of precise voltage sources that ensure a suitable accuracy and stability of the reproduction of the complex voltage ratio in relation to both the amplitude and phase. Such a solution imposes high demands on the voltage sources applied, which are mostly generators purpose-built for this kind of applications.

An impedance comparison circuit, in which commercial generators based on digital frequency synthesis are applied, is proposed below. Thanks to the application of a suitable circuit of complex voltage ratio measurement, high accuracy is achieved.

The impedance comparison is best performed in a balanced AC bridge circuit (Fig. 1.19).

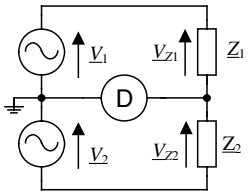


Fig. 1.19. Bridge with two sine wave generators

The balance condition is given by

$$\frac{\underline{V}_1}{\underline{V}_2} = \frac{\underline{V}_{Z1}}{\underline{V}_{Z2}} = \frac{\underline{Z}_1}{\underline{Z}_2}. \tag{1.32}$$

The accuracy of the comparison of the impedances $\underline{Z}_1$ and $\underline{Z}_2$ depends on the accuracy of the determination of the $\underline{V}_1$ and $\underline{V}_2$ ratio, further referred to as the complex voltage ratio $\underline{K}_V$.

It is possible to use digital sine-wave generators in a way that differs from the ways explored so far. Varying the parameters of the output voltages of the generators, the circuit is balanced, whereas the complex voltage ratio is not determined directly on the basis of appropriate settings of the generators in a state of balance, but by measurement. This approach permits the application of commercial generators with digital frequency synthesis, whose accuracy of amplitude settings or amplitude and phase temporal stability are insufficient (Rybski, 2000).

It is assumed that the generators' output voltage ratio is measured, and the result of the measurement is written as

$$\underline{K}_V = A + jB. \tag{1.33}$$

The accuracy of impedance comparison depends on the uncertainty of the measurement of the in-phase A and quadrature B, components of the complex voltage ratio.

The schema of the bridge realizing the principle of impedance comparison by measuring the complex voltage ratio is shown in Fig. 1.20. In the circuit, inductive voltage dividers are used in order to ensure high resolution of the generators' output voltages. The circuit is balanced by adjusting the IVD ratio $\underline{k}_1$, $\underline{k}_2$ of the dividers IVD1, IVD2, as well as by adjusting the phase shift angle φ between the output voltages of the generators G1, G2. When the bridge is in balance, the complex voltage ratio $\underline{V}_1/\underline{V}_2$ is determined by means of the sampling voltmeters V_1, V_2.

Taking the relationships (1.32), (1.33) into account, the bridge balance condition will assume the form

$$\frac{\underline{Z}_1}{\underline{Z}_2} = \frac{\underline{k}_1}{\underline{k}_2}\left(A + jB\right). \tag{1.34}$$

The presented measurement method has some advantages. Voltage measurement directly at the source terminals makes the best use of the voltmeter range, regardless of the ratio of the impedances compared. It has considerable significance, particularly when an A/D converter or a data acquisition card is used directly in the measurement. Moreover, the voltmeters' input impedance does not affect the bridge balance, and

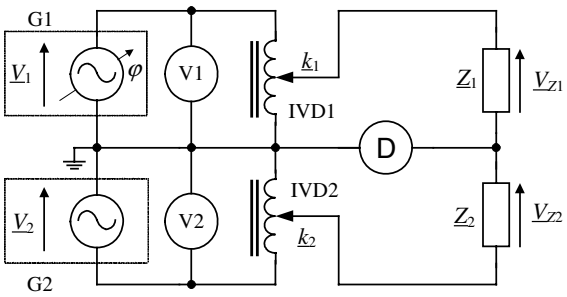


Fig. 1.20. Bridge with two generators for complex voltage ratio measurement

it is possible to measure voltages during the balancing. This accelerates reaching the balance. The simultaneous measurement of both voltages reduces the effect of the sources' short-term stability on measurement accuracy. The presented idea of impedance comparison has been realized in a circuit whose simplified block diagram is given in Fig. 1.21.

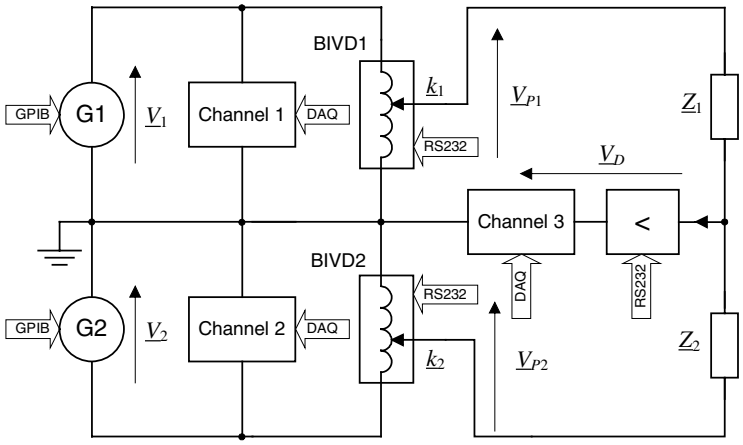


Fig. 1.21. Block diagram of a circuit for impedance comparison using a data acquisition card

The circuit consists of two commercial generators working in a synchronous mode, a four-channel data acquisition card, two binary inductive voltage dividers, BIVD1 and BIVD2, as well as a selective amplifier. The generators G1 and G2 constitute a two-phase sinusoidal voltage source with 0.001° resolution phase adjustment. Controlling the generators in the system is carried out by means of the interface GPIB. For fine amplitude adjustment, RS 232-interface controlled, 20-bit binary IVDs have been employed. The measurement of the generators' output voltages and the unbalance voltage are realized by means of a data acquisition card that works in an all-channel simultaneous sampling mode and features a 16-bit resolution. In the presented circuit, 3 channels are used (Channel 1, Channel 2, Channel 3). The system's software is realized in the LabWindows/CVI environment. The circuit has been designed to work in the frequency range from 100 Hz to 2 kHz.

An example of the comparison of two resistors is presented below. It is assumed that the compared elements are described by the parameters of an equivalent series circuit, and their impedances are determined by the relationships

$$\underline{Z}_{R1} = R_1 (1 + j\omega\tau_1), \quad \underline{Z}_{R2} = R_2 (1 + j\omega\tau_2), \tag{1.35}$$

where $\underline{Z}_{R1}$, $\underline{Z}_{R2}$ represent the impedance of the resistors, and R, τ stand for the AC resistance and the time constant of the resistor.

It is also assumed that the IVD ratios of the inductive dividers used for balancing are described by the relationships

$$\underline{k}_1 = k_{1n} (1 + \alpha_1 + j\beta_1), \quad \underline{k}_2 = k_{2n} (1 + \alpha_2 + j\beta_2), \tag{1.36}$$

in which k_{1n}, k_{2n} signify nominal values of the IVD ratios, whereas α_1, α_2 and β_1, β_2 represent in-phase and quadrature components, respectively, of the IVD ratio complex error of both inductive dividers.

Taking into account the relations (1.34)–(1.36), the impedance comparison condition is obtained:

$$\frac{R_1}{R_2} = \frac{k_{1n}}{k_{2n}} A \left\{ \left[1 + \alpha_1 - \alpha_2 - \frac{B}{A} (\beta_1 - \beta_2) \right] + \omega\tau_2 \left[\frac{B}{A} (1 + \alpha_1 - \alpha_2) + \beta_1 - \beta_2 \right] \right\}. \tag{1.37}$$

Proceeding in a similar way, it is possible to determine the impedance comparison conditions for the impedance of any type. The presented circuit was used for comparing the impedances R–R and R–C with the uncertainty approximately equal to $6 \cdot 10^{-6}$. A detailed analysis of metrological properties of the circuit and experimental results were described in (Rybski, 2004).

1.4.2. Virtual bridge

The most precise impedance measurements are still achievable by applying AC bridges with inductive voltage dividers (see Part 1.4.1). However, in many applications a lower precision, of the order of (10–100) ppm, is sufficient. Circuits presented in the literature that can guarantee the above precision increasingly often apply digital signal processing methods. An example of such a solution is the system using digital direct synthesis to generate sine waves. Among these systems, bridges applying two digital voltage sources and traditional balance algorithms can be mentioned (Helbach *et al.*, 1983; Waltrip and Oldham, 1995). Moreover, bridges balanced using the LMS algorithm (Awad *et al.*, 1994; Dutta *et al.*, 1987; 2001) and circuits applying compensating (successive approximation) methods (Tarach and Trenkler, 1993) are frequently used. Nowadays, the increasing applicability of DSP techniques, e.g. Fast Fourier Transform (FFT) (Ramm *et al.*, 1999) or the parameter estimation method (Angrisani *et al.*, 1996), is also observed in impedance measurements.

Here, the virtual bridge idea shown in Fig. 1.22 was used to determine the resistance and capacitance components of unknown impedance. Instead of a relatively complex non-linear optimisation algorithm employed in (Angrisani *et al.*, 1996), which produces estimates off-line, a recursive Outer-Bounding Ellipsoid (OBE) algorithm is proposed which, apart from its simplicity, offers a possibility to measure the unknown quantities on-line and then to quantify the uncertainty of the estimates.

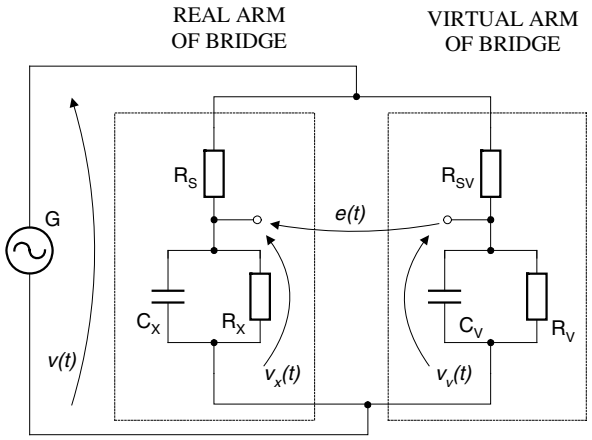


Fig. 1.22. Virtual bridge

In the steady state (i.e. when the transient response is small enough to be neglected), we have

$$v(k; A, B) = Au_1(k) + Bu_2(k), \tag{1.38}$$

where

$$u_1(k) = \sqrt{2}U \sin(\omega k \tau_c), \quad u_2(k) = \sqrt{2}U \cos(\omega k \tau_c), \tag{1.39}$$

$$A = \frac{R_V(R_V + R_S)}{(1 + \omega^2 C_V^2 R_S^2)R_V^2 + 2R_S R_V + R_S^2}, \tag{1.40}$$

$$B = -\frac{\omega C_V R_V^2 R_S}{(1 + \omega^2 C_V^2 R_S^2)R_V^2 + 2R_S R_V + R_S^2}, \tag{1.41}$$

assuming that $R_S = R_{SV}$.

In what follows, $u_1(k)$ and $u_2(k)$ are treated as known inputs. Let us note that if the parameters A and B were known (balanced state of the virtual bridge), we would be able to recover the original unknown resistance $R_X = R_V$ and capacitance $C_X = C_V$. In fact, it follows that

$$R_X = \frac{R_S(A^2 + B^2)}{A^2 + B^2 - A}, \quad C_X = \frac{B}{\omega R_S(A^2 + B^2)}. \tag{1.42}$$

Thus, we have managed to replace the original problem of directly finding R and C (which is highly non-linear) by the equivalent problem of calculating A and B, which is much simpler, as the model structure (1.38) is linear in its parameters and some effective on-line approaches can then be exploited.

As a matter of fact, we are faced with a classical parameter-estimation problem. Indeed, we take the measurements of the voltage $v(k)$ in (1.38) and hence obtain the sequence $v_x(k)$. The difference between the measured and model voltages,

$$e(k; A, B) = v_x(k) - v(k; A, B), \tag{1.43}$$

is called the output error. We wish this error to be as close to zero as possible, which can be carried out by a proper choice of A and B. Note that a precise reduction of $e(k)$ to zero is impossible in practice, since the measured voltage is always corrupted by some measurement errors which cannot be neglected.

The usual statistical framework adopted in estimation procedures assumes that these errors are modelled as realisations of independent random variables, with a known or parameterised distribution. In the case considered, however, we are forced to give up this usual approach, because the only information regarding the measurement errors is in the form of bounds (this is due to the fact that the data are collected through an A/D converter and the resulted quantisation errors seem to dominate other types of errors). The developed algorithm of virtual bridge operation was discussed in detail in (Kaczmarek *et al.*, 1998).

The virtual bridge algorithm was implemented on the measurement system that is shown in Fig. 1.23. The excitation signal for a real arm of the "virtual bridge" is provided by the HP33120A universal signal generator controlled by an IEEE488 interface. The plug-in National Instruments AT2150C measurement card is used for processing the voltages $v(t)$ and $v_x(t)$. Both voltages are simultaneously sampled and converted by two 16-bit sigma-delta A/D converters with a sampling frequency up to 51.2 kHz.

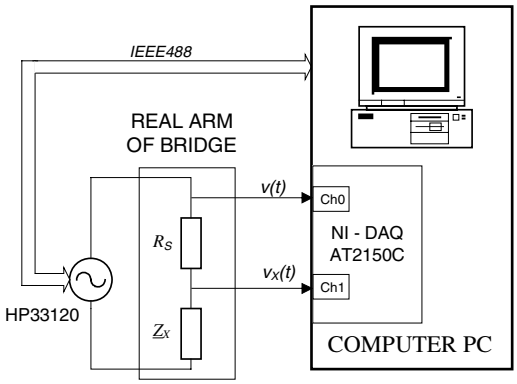


Fig. 1.23. Block diagram of the measuring system

Some experimental results that confirm the effectiveness of the proposed procedure are described in (Kaczmarek *et al.*, 1998). As an unknown impedance, there was used a capacitive impedance (R-C parallel model), but it is also possible to measure inductive impedances (R-L models). The results obtained from the virtual bridge were compared with those obtained from the HP4284 RLC meter with the basic accuracy 0.05%. As the standard resistance R_s, a decade resistor P4834 with the basic accuracy 0.02% and frequency range 100—10 kHz was chosen.

A recursive outer-bounding ellipsoid algorithm employed in the "virtual impedance bridge", which offers a possibility to measure on-line the unknown impedance, is proposed. This algorithm requires no sampling of the measured signals over the entire period. The hardware implementation based on a PC computer and a plug-in card has been applied to verify the possibilities of the OBE algorithm during the balanc-

ing of the virtual bridge. In this case, the computation time is critical for an on-line system. To increase the measurement speed of the bridge, a faster microprocessor is necessary. Then, after small changes of the measuring procedure, the system will have an ability of tracking time-varying parameters of the measured impedance.

1.4.3. AC power calibrator

Together with the growth of the demand for power energy quality, there are growing requirements concerning measuring devices used for measurements of power-energy parameters: watt meters, electric energy meters, power energy analyzers, etc. Among others, the devices of this type must have the ability to work in distorted environments – designed to measure the parameters of polyharmonic signals. Therefore, the standard sources (e.g. the electric power and energy calibrators) taken for periodic checking of the numerous groups of measuring devices of AC current parameters should stand out with the possibility for the generation of polyharmonic signals. In the case of AC power calibrators, on account of requirements concerning metrological and functional properties (wide spectrum of ranges, precision and number of controlled parameters), as well as circuit complexity, meeting this condition is particularly difficult (Arseneau *et al.*, 1995; Carullo *et al.*, 1998).

The structure of the AC power calibrator with the possibility of polyharmonic signal generation is presented in Fig. 1.24 (Kaczmarek and Kulesza, 2003). In the voltage and current paths of the calibrator, the generation of polyharmonic signals bases on the two-channel DDS integrated circuit. The first channel is used to generate the fundamental frequency, and the second one – higher harmonic frequencies. Multiplying D/A converters (MDAC) are taken for the realization of the amplitude setting of output signals. Further, the signals of fundamental frequencies and n-th harmonic are summed up. The resultant signal is passed to the input of the power amplifier. In order to obtain the appropriate precision of the generated output signals, a digital loop of the feedback built from the standardization circuit, ADC converter, DSP processor and multiplying DAC converter was used in each channel of the calibrator. Standardized measuring signals proportional to output calibrator signals are sampled (ADC processing). Subsequently, on the basis of the collected samples, the DSP processor determines the new values settings, which are transferred to the appropriate adjustment circuits. The entire process is repeated until the difference between the nominal (value of the setting of the output signal) and measured values is smaller than the assumed level.

The solution applied provides high stability and settings resolution of the frequency and phase angle of the generated signals. It permits precise measurement of calibrator output signal parameters (complex quantities) and applying advanced digital correction methods (in a digital way) of errors carried out by the hardware of the signal generation path and the output load.

1.5. Summary

The problems presented in the chapter, concerning the measurement and reproduction of the complex voltage ratio with the application of DSP algorithms, are one of the

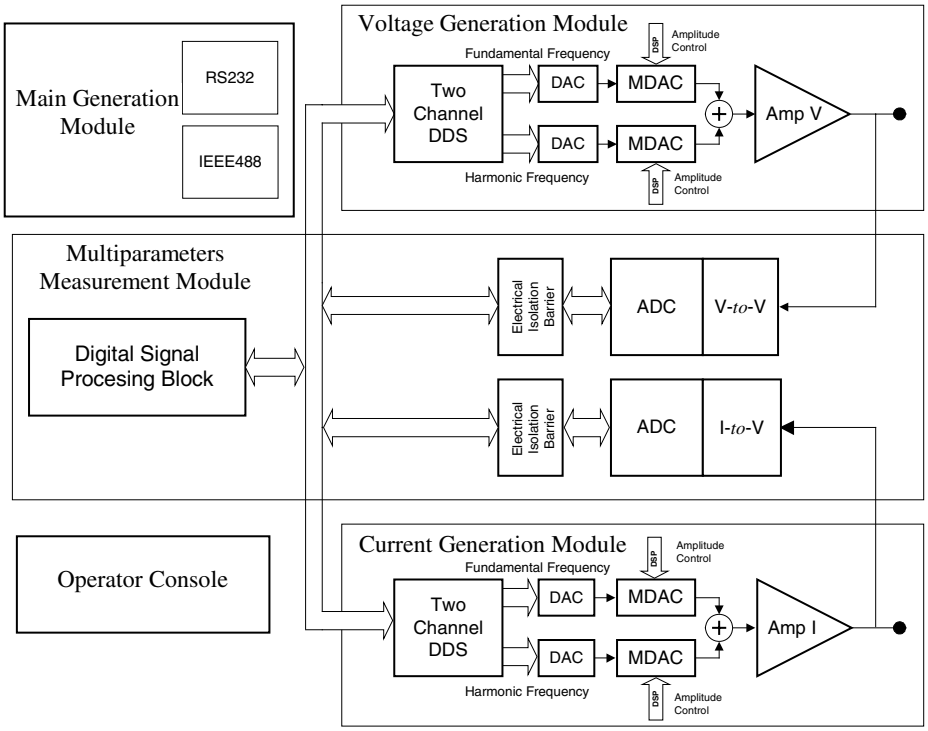


Fig. 1.24. Block diagram of the structure of the AC power calibrator

most important and dynamically developing subject matters of research in the area of accurate measurements of AC current quantities. Present tendencies and possibilities connected with applying DSP algorithms in the areas mentioned above were presented here using examples chosen from amongst works, conducted in last few years by authors in the range of accurate measurements of the impedance and AC power calibrators.

Digital methods of the generation of sinusoidal voltages and currents are fundamental to the construction of digital standards of the ratio of two sinusoidal signals. A method of direct digital synthesis, currently applied most frequently, was presented taking into consideration its most important advantages and limitations. In the area of accurate measurements of the complex voltage ratio, the method based on synchronous integration sampling and DFT is shown. From among many well-known methods—based on signal sampling—of determining the parameters of sinusoidal signals, it seems that the above method permits complex voltage ratio measurement with the smallest, achieved at this moment, uncertainty in the low frequencies range. The current tendencies and possibilities of using digital methods in area of the measurement and generation of sinusoidal voltages are illustrated well by two described examples. Applying such elements as inductive voltage dividers and measuring transformers is still necessary in the highest accuracy measurements (see Part 1.4.1). On the other hand—where possible—hardware solutions are reduced and replaced with the software carrying out more and more advanced algorithms (see Part 1.4.2).

The above digital techniques taken from the domain of digital signal processing also found applications in the structures of AC power standard sources (AC power calibrators). Digital methods are applied in the process of signal generation (e.g. direct digital frequency synthesis) as well as in the process of the stabilization of the main parameters of calibrator output signals (e.g. the FFT algorithm). The application of the digital technique in this field permits simplifying the structure of such complex measuring instruments, to which power calibrators belong, and increasing their functional possibilities, e.g. enabling the generation of polyharmonic signals (see Part 1.4.3) and improving dynamic properties (Gubisch *et al.*, 1997).

References

Angrisani L, Baccigalupi A. and Petrosanto A. (1996): *A digital signal-processing instrument for impedance measurement.* — IEEE Trans. Instrum. Meas., Vol. 45, No. 6, pp. 930–934.

Arseneau R., Filipski P.S. and Zelle J.J. (1995): *Portable and stable source of AC voltage, current, and power.* — IEEE Trans. Instrum. Meas., Vol. 44, No. 2, pp. 433–435.

Awad S.S., Narasimhamurthi N. and Ward W.H. (1994): *Analysis, design, and implementation of an AC bridge for impedance measurements.* — IEEE Trans. Instrum. Meas., Vol. 43, No. 6, pp. 894–899.

Bell B.A. (1990): *Standards for waveform metrology based on digital techniques.* — J. Research National Institute of Standards and Technology, Vol. 95, No. 4, pp. 377–405.

Bohaček J. (2004): *A QHE-based system for calibrating impedance standards.* — IEEE Trans. Instrum. Meas., Vol. 55, No. 4, pp. 977–980.

Callegaro L. and D'Elia V. (2001): *Automated system for inductance realization traceable to AC resistance with a three-voltmeter method.* — IEEE Trans. Instrum. Meas., Vol. 50, No. 6, pp. 1630–1633.

Callegaro L., Galzerani G. and Svelto C. (2001): *A multiphase direct-digital-synthesis sinewave generator for high-accuracy impedance comparison.* — IEEE Trans. Instrum. Meas., Vol. 50, No. 4, pp. 926–929.

Carullo A., Ferraris F., Parvis M. and Vallan A. (1998): *Phantom power generator for the calibration of wattmeters in distorted environments.* — Proc. IMEKO TC-4 Symp. *Development in Digital Measuring Instrumentation*, Italy, Naples, pp. 67–71.

Ciglaric S., Fefer D. and Jeglic A. (2002): *Evaluation of an alternatively designed digital phase angle standard.* — IEEE Trans. Instrum. Meas., Vol. 51, No. 4, pp. 845–848.

Corney A.C. (2003): *Digital generator assisted impedance bridge.* — IEEE Trans. Instrum. Meas., Vol. 52, No. 2, pp. 388–391.

Crescini D., Flammini A., Mariolli D. and Taroni A. (1998): *Application of FFT-based algorithm to signal processing of LVDT position sensors.* — IEEE Trans. Instrum. Meas., Vol. 47, No. 5, pp. 1119–1123.

Dutta M, Bhattacharyya S.N. and Choudhury J.K. (1987): *An application of an LMS adaptive algorithm for a digital AC bridge.* — IEEE Trans. Instrum. Meas., Vol. IM-36, No. 4, pp. 894–897

Dutta M., Rakshit A. and Bhattacharyya S.N. (2001): *Development and study of an automatic AC bridge for impedance measurement.* — IEEE Trans. Instrum. Meas., Vol. IM-50, No. 5, pp. 1048–1052.

Gubisch A., Lualdi P. L., Miljanic P. N. and West J. L. (1997): *Power calibrator using sampled feedback for current and voltage.* — IEEE Trans. Instrum. Meas., Vol. 46, No. 2, pp. 403–407.

Helbach W., Marczinowski P. and Trenkler G. (1983): *High-precision automatic digital AC bridge.* — IEEE Trans. Instrum. Meas., Vol. IM-32, No. 1, pp. 159–162.

Ilic D. and Butorac J. (2001): *Use of Precise digital voltmeters for phase measurements.* — IEEE Trans. Instrum. Meas., Vol. 50, No. 2, pp. 449–452.

Kaczmarek J. and Kulesza W. (2003): *Three-phase AC power calibrator INMEL 8033.* — Proc. Conf. *Metrologia Wspomagana Komputerowo, MWK*, Waplewo, Poland, Vol. III, pp. 45–50, (in Polish).

Kaczmarek J. and Rybski R. (1995): *A direct digital synthesis-method based phase sensitive detector for automatic bridge application.* — Proc. 7-th Int. Symp. *Modern electrical and magnetic measurement, IMEKO TC-4*, Prague, Czech Republic, Part 1, pp. 76–80.

Kaczmarek J., Rybski R. and Uciński D. (1998): *A recursive DSP approach to impedance measurement.* — Proc. IMEKO TC-4 Symp. *Development in Digital Measuring Instrumentation*, Naples, Italy, Vol. II, pp. 690–693.

Kampik M, Laiz H. and Klonz M. (2000): *Comparison of three accurate methods to measure AC voltage at low frequencies.* — IEEE Trans. Instrum. Meas., Vol. 49, No. 2, pp. 429–433.

Kürten Ihlenfeld W.G., Ramm G., Bachmair H. and Moser H. (2003): *Evaluation of the synchronus generation and sampling technique.* — IEEE Trans. Instrum. Meas., Vol. 52, No. 2, pp. 371–374.

Lapuh R. and Svetik Z. (1997): *Evaluation of a voltage source with tree calculable RMS output.* — IEEE Trans. Instrum. Meas., Vol. 46, No. 4, pp. 784–788.

Locci N. and Muscas C. (2001): *Comparative analysis between active and passive current transducers in sinusoidal and distorted conditions.* — IEEE Trans. Instrum. Meas., Vol. 50, No. 1, pp. 123–128.

Muciek A. (1997): *Digital impedance bridge based on a two-phase generator.* — IEEE Trans. Instrum. Meas., Vol. 46, No. 2, pp. 467–470.

Muciek J. and Muciek A. (1999): *The method based on integrative sampling for the precise measurement of the RMS value at low frequencies.* — Proc. Conf. *Metrologia Wspomagana Komputerowo*, Warsaw, Poland, pp. 99–104, (in Polish).

Pogliano U. (1997): *Precision measurement of AC voltage below 20 Hz at IEN.* — IEEE Trans. Instrum. Meas., Vol. 46, No. 2, pp. 369–372.

Pogliano U. (2001): *Use of integrative anlog-to-digital converters for high-precision measurement of electrical power.* — IEEE Trans. Instrum. Meas., Vol. 50, No. 5, pp. 1315–1318.

Pogliano U.(2006): *Evaluation of the uncertainties in the measurement of distorted power by means of the IEN sampling system.* — IEEE Trans. Instrum. Meas., Vol. 55, No. 2, pp. 620–624.

Ramm G. and Moser H. (2001): *From the calculable AC resistor to capacitor dissipation factor determination on the basis of time constants.* — IEEE Trans. Instrum. Meas., Vol. 50, No. 2, pp. 286–289.

Ramm G. and Moser H. (2003): *Calibration on electronic capacitance and dissipation factor bridges.* — IEEE Trans. Instrum. Meas., Vol. 52, No. 2, pp. 396–399.

Ramm G., Moser H. and Braun A. (1999): *A new scheme for generating and measuring active, reactive, and apparent power at power frequencies with uncertainties of 2.5×10^6.* — IEEE Trans. Instrum. Meas., Vol. 48, No. 2, pp. 422–426.

Rybski R. (2000): *Digital Sinewave Sources in Impedance Comparators.* — Wydawnictwo Politechniki Zielonogórskiej, No. 100, (in Polish).

Rybski R. (2004): *Impedance comparison in a circuit with two digital sinewave generators.* — Metrology and Measurement Systems, Vol. XI, No. 2, pp. 131–145.

Rybski R. and Kaczmarek J. (1997): *The precise unbalanced AC bridge for capacitance measurements.* — Proc. XIV *IMEKO World Congess*, Tampere, Finland, Vol. IVB, pp. 54–59.

Rybski R. and Kaczmarek J. (2000): *Calibration of a system for the measurement of complex voltage ratios.* — Proc. XVI *IMEKO World Congress*, Vienna, Austria, Vol. X, pp. 287–290.

Rybski R. and Kaczmarek J. (2001): *Comparison of R-C components in a circuit with two digital sinewave generators.* — Proc. 2-nd Polish Conf. *Kongres Metrologii*, Warsaw, Poland, pp. 365–368, (in Polish).

Rybski R. and Kaczmarek J. (2002): *Calibration of a sampling system for the of the complex ratio measurement of the voltage signals.* — Pomiary, Automatyka, Kontrola, No. 7/8, pp. 93–96, (in Polish).

Rybski R. and Krajewski (2003): *Measurement of displacement using Linear variable Differential Transformer (LVDT) and Discrete Fourier Transform.* — Pomiary, Automatyka, Kontrola, No. 6, pp. 15–18, (in Polish).

Rybski R, Kaczmarek J. and Krajewski M. (2004): *Accurate measurement of complex voltage ratio with a sampling voltmeter.* — Metrology and Measurement Systems, Vol. XI, No. 2, pp. 148–158.

Sasdelli R., Menchetti A. and Peretto L. (1998): *A digital instrument for the calibration of current-to-voltage transducers.* — IEEE Trans. Instrum. Meas., Vol. 47, No. 2, pp. 189–193.

Skubis T. (1995): *Calibration methods of electrical instruments reproducing the standard ratio of two values.* — Zeszyty Naukowe Politechniki Śląskiej, Elektryka, No. 143, Gliwice, (in Polish).

Tarach D. and Trenkler G.(1993): *High-accuracy N-port impedance measurement by means of modular digital AC compensators.* — IEEE Trans. Instrum. Meas., Vol. 42, No. 2, pp. 622–626.

Waltrip B.C. and Oldham N.M. (1995): *Digital impedance bridge.* — IEEE Trans. Instrum. Meas., Vol. 44, No. 2, pp. 436–439.

Chapter 2

ESTIMATION OF CORRELATION FUNCTIONS ON THE BASIS OF DIGITAL SIGNAL REPRESENTATION

Jadwiga LAL-JADZIAK*

2.1. Introduction

In most contemporary devices and measurement systems, the investigated signals are the subject of digitization in the time domain (sampling) and in the value domain (quantization). The distortion accompanying quantization may influence measurement accuracy. The aim of this paper is an analysis of the influence of Analog-to-Digital (A/D) conversion and A/D conversion with dither on the accuracy of the determination of correlation functions.

The crosscorrelation function of the ergodic processes $\{x(t)\}$ and $\{y(t)\}$ can be expressed by the relation (Bendat and Piersol, 1986; 1993):

$$R_{xy}(\tau) = \lim_{T\to\infty} \frac{1}{T} \int_0^T x(t)y(t+\tau)\,\mathrm{d}t, \tag{2.1}$$

where $x(t)$ and $y(t)$ are realizations of the processes, τ is the delay, and T is the observation time. For simplicity, but without affecting the generality of the deliberations, it has been assumed that $x(t)$ and $y(t)$ have zero mean values. Substituting $y(t) = x(t)$ into (2.1), we obtain a relation for the autocorrelation function $R_x(\tau)$.

Two methods of function estimation are currently being used. One is a direct method based on a definition, the other – an indirect one – consists in determining the spectral power density before subjecting it to the inverse Fourier transform (Bendat and Piersol, 1993). The direct method is simpler to write software for and represents a more logical way of approaching the definition. The advantage of the other – its great

* Institute of Electrical Metrology
e-mail: j.jadziak@ime.uz.zgora.pl

computational efficiency achieved by using FFT algorithms – is losing its significance owing to the constantly rising processing speed of the currently offered IT tools.

From among the direct digital estimators of the function (2.1), let us analyze two, in the form of

$$\tilde{R}^{q}_{xy}(k,M) = \frac{1}{M}\sum_{i=0}^{M-1} x_{q1}(i\Delta t)y_{q2}(i\Delta t + k\Delta t), \tag{2.2}$$

$$\tilde{R}^{d}_{xy}(k,M) = \frac{1}{M}\sum_{i=0}^{M-1} x_{1q1}(i\Delta t)y_{1q2}(i\Delta t + k\Delta t). \tag{2.3}$$

The estimator (2.2) is a digital estimator obtained on the basis of A/D converted signals, i.e. after simultaneous sampling with a constant step Δt (where $M = T/\Delta t$ is the number of samples taken, T is the averaging time, and $k\Delta t$ is the delay) and quantization with the steps q_1 and q_2, respectively. However, the estimator (2.3) was obtained on the basis of A/D converted signals with dither signals.

Below are presented the issues touched upon in this chapter. In Section 2.2, Widrow's quantizing theorems and quantizing reconstruction conditions for the estimation of auto- and crosscorrelation functions are presented. In Sections 2.3 and 2.4, the influence of A/D conversion without and with dither on the accuracy of auto- and crosscorrelation function determination is considered. Analytic expressions for bias errors of direct digital estimators are derived and discussed. For negligible bias errors, the conditions which signals and dithers should satisfy are formulated. The realizability of these conditions is evaluated.

It is shown that the application of dither signals leads to an improvement in quantizing reconstruction and makes it possible, after taking Sheppard corrections into account, to obtain unbiased estimators of correlation functions. In this way, the bias can be eliminated, in practice – reduced, without knowing its mathematical model. However, it may lead to an increase in the estimator variance. An increase in the variance manifests itself in an increase in the scatter of the measurement results (increase in a type A uncertainty level). The analysis of a variance component coming from quantization with a dither signal is presented in Section 2.5.

Analytical models of estimation errors of correlation functions are highly complex, therefore the evaluation of accuracy is difficult and in many cases unachievable. For this reason a virtual correlator model is proposed as an alternative to analytical modeling. The model allows determining the uncertainty of digital measurements. In Section 2.6, some preliminary research results are presented and discussed. The comparison of bias of the mean square value estimator modeled in the *Mathcad* program with that obtained by means of a virtual correlator model is carried out.

2.2. Statistical theory of quantization for moments of signals

Widrow is considered to be the creator of signal quantization theory. In his publications from the end of the 1950s, Widrow showed that quantization can be thought of as sampling the probability density function of the signal being converted and he formulated several important theorems (Widrow, 1956). Widrow's theory is valid for

uniform quantization in a quantizer with an unlimited range and a signal being a continuous random variable (a random variable is the value of an ergodic stochastic process). Owing to the assumptions of the ergodicity of processes, the conclusions following from the theory can be transferred onto particular process realizations, thus stochastic signals that are subject to investigation in practice.

The interest in A/D conversion expressed in prestigious journals induced Widrow to formulate the theorems once again and to take a stance on certain opinions (Widrow *et al.*, 1996). The demonstration of Widrow's theory calls for the introduction of the concept of the characteristic function.

The Fourier transform of the probability density function $p(x)$ is known as the Characteristic Function (CF) (Wojnar, 1988). The CF of the signal x is

$$\Phi_x(v) = \int_{-\infty}^{\infty} p(x)\, e^{jvx}\, \mathrm{d}x = E\left[e^{jvx}\right]. \tag{2.4}$$

The joint CF of the signals x and y is

$$\Phi_{xy}(v_1, v_2) = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} p(x,y)\, e^{j(v_1 x + v_2 y)}\, \mathrm{d}x\, \mathrm{d}y = E\left[e^{j(v_1 x + v_2 y)}\right]. \tag{2.5}$$

If the quantization characteristic is of the roundoff type (Fig. 2.1), then the characteristic functions $\Phi_{xq}(v)$ and $\Phi_{xqyq}(v_1, v_2)$, corresponding to the signals which have been quantized, can be determined from the formulae (2.6) and (2.7) (Widrow *et al.*, 1996).

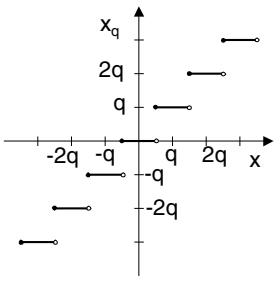


Fig. 2.1. Characteristic of a uniform quantizer (roundoff type)

At the same time the signal x_q resulting from quantization (with a step q) has the characteristic function in the form

$$\Phi_{xq}(v) = \sum_{i=-\infty}^{\infty} \Phi_x\left(v - \frac{2\pi}{q} i\right) \operatorname{sinc}\left[\frac{q}{2}\left(v - \frac{2\pi}{q} i\right)\right], \tag{2.6}$$

whereas the joint characteristic function of the quantized (with the steps q_1 and q_2, respectively) signals x_q and y_q can be expressed as follows:

$$\Phi_{xqyq}(v_1, v_2) = \sum_{i=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \Phi_{xy}\left(v_1 - \frac{2\pi}{q_1} i, v_2 - \frac{2\pi}{q_2} l\right) \times \operatorname{sinc}\left[\frac{q_1}{2}\left(v_1 - \frac{2\pi}{q_1} i\right)\right] \operatorname{sinc}\left[\frac{q_2}{2}\left(v_2 - \frac{2\pi}{q_2} l\right)\right]. \tag{2.7}$$

In Fig. 2.2 there are shown the characteristic functions of the signals x_q and y_q.

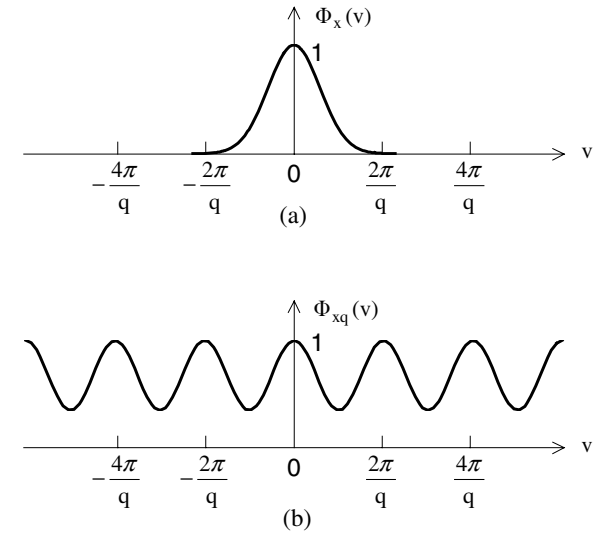


Fig. 2.2. Characteristic function: (a) of the signal x, (b) of the signal x_q

Moments of signals, such as the mean, the mean square, auto- and crosscorrelation, etc., can be determined by taking derivatives of the CF (Widrow and Kollar, 2006). The m-th moment of the signal x is

$$E\left[x^m\right] = j^{-m} \left.\frac{\mathrm{d}^m \Phi_x(v)}{\mathrm{d}v^m}\right|_{v=0}. \tag{2.8}$$

The joint $(m+n)$-th moment of the signals x and y is

$$E\left[x^m y^n\right] = j^{-(m+n)} \left.\frac{\partial^{m+n} \Phi_{xy}(v_1, v_2)}{\partial^m v_1 \partial^n v_2}\right|_{(v_1, v_2)=(0,0)}, \tag{2.9}$$

where $j = \sqrt{-1}$.

The theorems of Widrow's quantization theory concerning the reconstruction of moments are the foolowing two:

Quantizing Theorem II (QT II). If the CF of x is band-limited so that

$$\Phi_x(v) = 0 \quad \text{when} \quad |v| > \frac{2\pi}{q} - \varepsilon, \tag{2.10}$$

with ε positive and arbitrarily small, then the moments of x can be calculated from the moments of x_q.

Multidimensional Quantizing Theorem II (QT II). If the CF of $x_1, \ldots, x_N$ is band-limited in N-dimensions, so that[†]

$$\Phi_{x_1,\ldots,x_N}(v_1, \ldots, v_N) = 0 \ \text{ for } \ |v_k| > \frac{2\pi}{q} - \varepsilon, \ \text{ for any } \ k \in [1, N], \tag{2.11}$$

with ε positive and arbitrarily small, then the moments of $x_1, \ldots, x_N$ can be calculated from the moments of $x_{q1}, \ldots, x_{qN}$.

When QT II is satisfied, the quantization noise is uniformly distributed in multidimensions, white, und uncorrelated with x. It has a mean square of $q^2/12$. In practice, input CFs are not exactly band-limited, and the quantizing theorems apply only approximately.

The exact condition of whiteness was given (Sripad and Snyder, 1977) in terms of the joint CF of two input signals as

$$\Phi_{x1,x2}\left(\frac{2\pi i}{q}, \frac{2\pi l}{q}\right) = 0, \tag{2.12}$$

for every integer value of i and l, except $(i, l) = (0, 0)$. This condition is quite difficult to apply in practice (Widrow *et al.*, 1996).

According to the theory of quantization, when the relations (2.10) and (2.11) are satisfied, then the following relations are valid:

$$E[x_q] = E[x], \tag{2.13}$$

$$E\left[x_q^2\right] = E\left[x^2\right] + \frac{q^2}{12}, \tag{2.14}$$

$$E[x_q y_q] = E[xy] \, (x \neq y). \tag{2.15}$$

Assuming $x = x(t_1)$ and $y = x(t_1 + \tau)$ in the formulae (2.14) as well as (2.15) and bearing in mind that $E[x_q^2(t_1)] = R_{xq}(t_1, t_1)$, the following expression can be obtained:

$$R_{xq}(t_1, t_1{+}\tau) = \begin{cases} R_x(t_1, t_1{+}\tau){+}\dfrac{q^2}{12} & (\tau{=}\,0), \\ R_x(t_1, t_1 + \tau) & (\tau \neq 0), \end{cases} \tag{2.16}$$

whereas by substituting $x = x(t_1)$ and $y = y(t_1 + \tau)$ into the formula (2.15) we obtain the relation

$$R_{xqyq}(t_1, t_1 + \tau) = R_{xy}(t_1, t_1 + \tau) \quad (x \neq y). \tag{2.17}$$

[†] Not every signal needs to be quantized with the same step q. For example, the reproduction of the second-order moment can be done on the basis of two signals quantized with the steps q_1 and q_2, respectively (Korn, 1966).

The formulae (2.16) and (2.17) define the connections between correlation functions and their estimators obtained on the basis of quantized signals. If the processes are stationary (ergodic processes are stationary), then these expressions are independent of the time t_1. They are valid in reproducibility conditions. Certain real signals satisfy Widrow's theorem with such a good approximation that the obtained results are of high accuracy (Kawecka and Lal-Jadziak, 2004).

2.3. Estimation errors due to A/D conversion of signals

An often applied direct crosscorrelation function estimator is the estimator (2.2) created from signals digitized in both the time and value domains. To simplify the notation, let us present it as the relation

$$\tilde{R}^q_{xy}(k,M) = \frac{1}{M}\sum_{i=0}^{M-1} x_{q1}(i)y_{q2}(i+k), \tag{2.18}$$

where Δt occurs in an implicit form.

Figure 2.3 shows the basic structure of a circuit working according to such an algorithm.

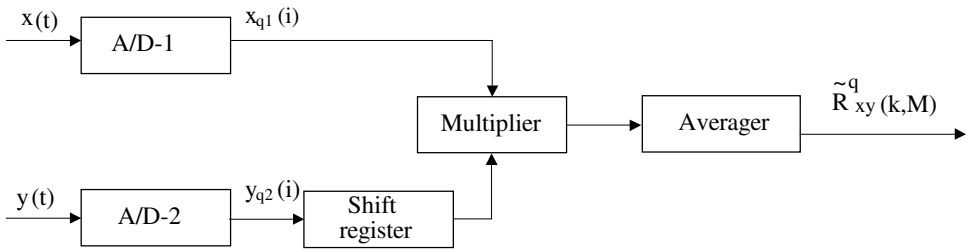


Fig. 2.3. Digital analyzer of the crosscorrelation function

The estimation accuracy can be determined by means of the mean square error defined by the formula (Bendat and Piersol, 1986):

$$e^2_{\tilde{R}^q_{xy}}(k,M) = E\left[\left(\tilde{R}^q_{xy}(k,M) - R_{xy}(k)\right)^2\right]. \tag{2.19}$$

This error is a sum of the variance and the square of the estimator bias, i.e.

$$e^2_{\tilde{R}^q_{xy}}(k,M) = Var\left[\tilde{R}^q_{xy}(k,M)\right] + b^2\left[\tilde{R}^q_{xy}(k,M)\right], \tag{2.20}$$

where

$$Var\left[\tilde{R}^q_{xy}(k,M)\right] = E\left[\tilde{R}^q_{xy}(k,M)\right]^2 - E^2\left[\tilde{R}^q_{xy}(k,M)\right], \tag{2.21}$$

$$b\left[\tilde{R}^q_{xy}(k,M)\right] = E\left[\tilde{R}^q_{xy}(k,M)\right] - R_{xy}(k). \tag{2.22}$$

The variance and bias describe, respectively, the random and systematic components of the error.

Employing simple transformations, we can show that the mean square error of the analyzed estimator assumes the form

$$e^2_{\tilde{R}^q_{xy}}(k,M) = \frac{1}{M^2}\sum_{i=0}^{M-1}\sum_{l=0}^{M-1} E\left[x_{q1}(i)\,y_{q2}(i+k)\,x_{q1}(l)\,y_{q2}(l+k)\right] - R^2_{xy}(k), \quad (2.23)$$

which means that it is determined by the fourth-order moments of A/D converted signals.

The calculation of the error (2.23) in general is not possible because its level depends not only on the number of samples used for estimation, and A/D conversion resolution, but also on the class of the signals being converted and the delay between them.

If the assumption that the processes are stationary holds, then the expected value of the estimator $\tilde{R}^q_{xy}(k,M)$ is equal to

$$\begin{aligned} E\left[\tilde{R}^q_{xy}(k,M)\right] &= E\left[\frac{1}{M}\sum_{i=0}^{M-1} x_{q1}(i)y_{q2}(i+k)\right] \\ &= \frac{1}{M}\sum_{i=0}^{M-1} E\left[x_{q1}(i)y_{q2}(i+k)\right] = E\left[x_{q1}y_{q2}\right], \end{aligned} \quad (2.24)$$

where $x_{q1} = x_{q1}(0)$, $y_{q2} = y_{q2}(k)$. In other words, the expected value of the digital correlation function estimator, i.e. the estimator obtained on the basis of the quantized samples, is equal to the joint second-order moment of the quantized signals x_{q1} and y_{q2}, and can be expressed by the formula (Lal-Jadziak, 2001a; 2001b; 2001c):

$$\begin{aligned} E\left[\tilde{R}^q_{xy}(k,M)\right] = R_{xy}(k) &+ \frac{q_1}{2\pi}\sum_{\substack{i=-\infty\\ i\neq 0}}^{\infty}\frac{(-1)^i}{i}\frac{\partial\Phi_{xy}(v_1-2\pi i/q_1,v_2)}{\partial v_2}\bigg|_{(v_1,v_2)=(0,0)} \\ &+ \frac{q_2}{2\pi}\sum_{\substack{l=-\infty\\ l\neq 0}}^{\infty}\frac{(-1)^l}{l}\frac{\partial\Phi_{xy}(v_1,v_2-2\pi l/q_2)}{\partial v_1}\bigg|_{(v_1,v_2)=(0,0)} \\ &- \frac{q_1q_2}{4\pi^2}\sum_{\substack{i=-\infty\\ i\neq 0}}^{\infty}\sum_{\substack{l=-\infty\\ l\neq 0}}^{\infty}\frac{(-1)^{i+l}}{il}\Phi_{xy}\left(-\frac{2\pi}{q_1}i,-\frac{2\pi}{q_2}l\right), \end{aligned} \quad (2.25)$$

where $\Phi_{xy}(v_1, v_2)$ is the joint characteristic function of the unquantized signals x and y. The bias of the estimator (2.2) is of the form

$$b\left[\tilde{R}^q_{xy}(k,M)\right] = \frac{q_1}{2\pi}\sum_{\substack{i=-\infty\\ i\neq 0}}^{\infty}\frac{(-1)^i}{i}\frac{\partial\Phi_{xy}(v_1-2\pi i/q_1,v_2)}{\partial v_2}\bigg|_{(v_1,v_2)=(0,0)} + \frac{q_2}{2\pi}\sum_{\substack{l=-\infty\\ l\neq 0}}^{\infty}\frac{(-1)^l}{l}\frac{\partial\Phi_{xy}(v_1,v_2-2\pi l/q_2)}{\partial v_1}\bigg|_{(v_1,v_2)=(0,0)} - \frac{q_1q_2}{4\pi^2}\sum_{\substack{i=-\infty\\ i\neq 0}}^{\infty}\sum_{\substack{l=-\infty\\ l\neq 0}}^{\infty}\frac{(-1)^{i+l}}{il}\Phi_{xy}\left(-\frac{2\pi}{q_1}i,-\frac{2\pi}{q_2}l\right). \tag{2.26}$$

From the formula (2.26), it follows that the fulfillment by the signals x and y joint characteristic function, as well as its derivatives, of the conditions

$$\Phi_{xy}\left(\frac{2\pi}{q_1}i,\frac{2\pi}{q_2}l\right)=0, \tag{2.27a}$$

$$\frac{\partial\Phi_{xy}(v_1,v_2-2\pi l/q_2)}{\partial v_1}\bigg|_{(v_1,v_2)=(0,0)}=0,\quad \frac{\partial\Phi_{xy}(v_1-2\pi i/q_1,v_2)}{\partial v_2}\bigg|_{(v_1,v_2)=(0,0)}=0, \tag{2.27b}$$

for $\forall i\neq 0$ and $\forall l\neq 0$ leads to a lack of bias.

Of course, the fulfillment of the Widrow reconstruction condition, given by the formula

$$\Phi_{xy}(v_1,v_2)=0 \text{ for } |v_1|>\frac{2\pi}{q_1}-\varepsilon \text{ and } |v_2|>\frac{2\pi}{q_2}-\varepsilon, \tag{2.28}$$

where ε is an infinitesimally small positive number, leads to ensuring (2.27a), (2.27b). In practice, signals completely fulfilling the conditions (2.27a), (2.27b) or (2.28) do not exist, and the estimator (2.2) is always biased.

Assuming that $x = x(0)$, $y = x(k)$, $x_{q1} = x_{q1}(0)$, $y_{q2} = x_{q2}(k)$ in the formulae (2.23)–(2.28), we can obtain an expression corresponding to digital estimation of an autocorrelation function in a two-channel circuit (this method is known as autocorrelation analysis via a crosscorrelation function). However, assuming additionally that $q_1 = q_2 = q$, we obtain a relation for autocorrelation function estimation for the argument $k\Delta t \neq 0$, realized in the circuit presented in Fig. 2.4 (Lal-Jadziak, 2001a; 2001b).

It remains to evaluate the autocorrelation function estimator for $k\Delta t = 0$, i.e. the signal mean square value. If the assumption that the process $\{x(t)\}$ is stationary holds, then the expected value of the estimator $\tilde{R}^q_x(0,M)$ is equal to

$$E\left[\tilde{R}^q_x(0,M)\right]=E\left[\frac{1}{M}\sum_{i=0}^{M-1}x_q^2(i)\right]=\frac{1}{M}\sum_{i=0}^{M-1}E\left[x_q^2(i)\right]=E\left[x_q^2\right], \tag{2.29}$$

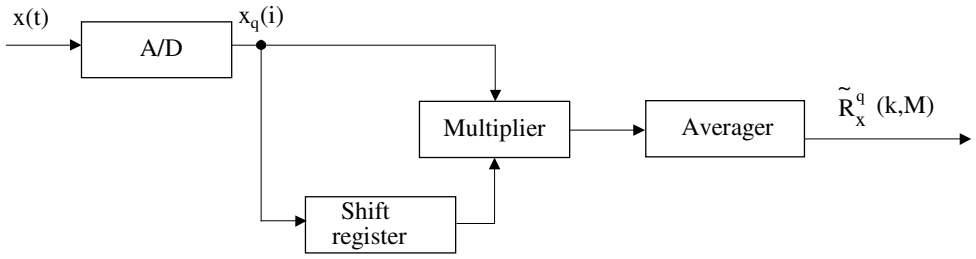


Fig. 2.4. Digital analyzer of the autocorrelation function

where $x_q = x_q(0)$. In other words, the expected value of the digital mean square value estimator, i.e. the estimator obtained on the basis of quantized samples, is equal to the second-order moment of the quantized signal x_q and can be expressed by the formula (Domańska, 1995):

$$E\left[x_q^2\right] = E\left[x^2\right] + \frac{q^2}{12} + \frac{q}{\pi}\sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \dot{\Phi}_x\left(\frac{2\pi}{q}i\right)\frac{(-1)^{i+1}}{i} + \frac{q^2}{2\pi^2}\sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \Phi_x\left(\frac{2\pi}{q}i\right)\frac{(-1)^{i}}{i^2}. \tag{2.30}$$

After taking account of the Sheppard correction in the result, the bias level can be determined by the relation

$$b\left[E\left[x_q^2\right]\right] = \frac{q}{\pi}\sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \dot{\Phi}_x\left(\frac{2\pi}{q}i\right)\frac{(-1)^{i+1}}{i} + \frac{q^2}{2\pi^2}\sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \Phi_x\left(\frac{2\pi}{q}i\right)\frac{(-1)^{i}}{i^2}. \tag{2.31}$$

The fulfillment of the conditions

$$\Phi_x\left(\frac{2\pi}{q}i\right) = 0, \quad \forall i \neq 0 \tag{2.32a}$$

and

$$\dot{\Phi}_x\left(\frac{2\pi}{q}i\right) = 0, \quad \forall i \neq 0 \tag{2.32b}$$

makes the estimator bias assume the value 0.

Naturally, satisfying the Widrow theorem (2.10) leads to ensuring the conditions (2.32a) and (2.32b).

2.4. Estimation errors caused by the application of A/D conversion with dither

A/D conversion with a dither signal has become a promising direction in the development of A/D converters (Domańska, 1995; Wagdy, 1989; Widrow and Kollar, 2006).

During A/D conversion with dither, two extra signals $d_1(t)$ and $d_2(t)$, called dither signals, are added to the signals $x(t)$ and $y(t)$:

$$x_1(t) = x(t) + d_1(t), \tag{2.33a}$$

$$y_1(t) = y(t) + d_2(t). \tag{2.33b}$$

The obtained signals $x_1(t)$ and $y_1(t)$ are converted to the digital form $x_{1q1}(i\Delta t)$ and $y_{1q2}(i\Delta t)$. Next, they are delayed with respect to each other by k samples, multiplied, and the result of the multiplication is averaged. The estimator thus obtained assumes the form (2.3) or – after simplifying the notation – can be expressed by the formula

$$\tilde{R}^d_{xy}(k, M) = \frac{1}{M} \sum_{i=0}^{M-1} x_{1q1}(i) y_{1q2}(i + k). \tag{2.34}$$

Figure 2.5 shows the basic structure of a circuit working according to such an algorithm.

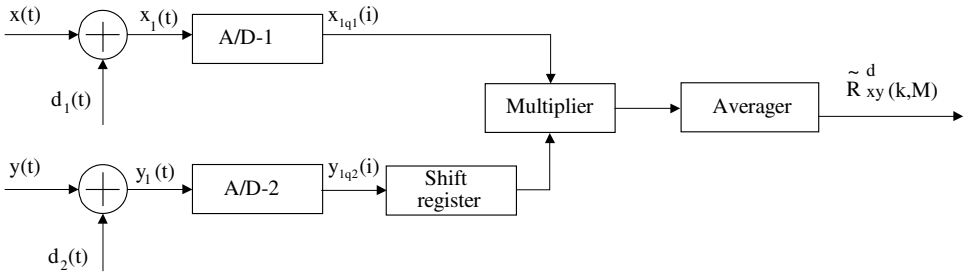


Fig. 2.5. Digital analyzer of the crosscorrelation function with dither signals

Under the stationary state conditions, the following relation holds true:

$$E\left[\tilde{R}^d_{xy}(k, M)\right] = \frac{1}{M} \sum_{i=0}^{M-1} E\left[x_{1q1}(i) y_{1q2}(i + k)\right] = E\left[x_{1q1} y_{1q2}\right], \tag{2.35}$$

where $x_{1q1} = x_{1q1}(0)$, $y_{1q2} = y_{1q2}(k)$. In other words, the expected value of the estimator with dither is equal to the joint second-order moment of the quantized signals x_{1q1} and y_{1q2}, and can be expressed by the joint characteristic function $\Phi_{x1qy1q}(v_1, v_2)$ (Lal-Jadziak, 2001b; 2001c):

$$E\left[\tilde{R}^d_{xy}(k, M)\right] = -\left.\frac{\partial^2 \Phi_{x1q1y1q2}(v_1, v_2)}{\partial v_1 \partial v_2}\right|_{(v_1, v_2)=(0,0)}. \tag{2.36}$$

For the roundoff quantization characteristic, the relationship between the characteristic function $\Phi_{x1qy1q}(v_1, v_2)$ of quantized signals and the characteristic function

$\Phi_{x1y1}(v_1, v_2)$ of unquantized signals can be derived from the formula

$$\Phi_{x1q1y1q2}(v_1, v_2) = \sum_{i=-\infty}^{\infty} \sum_{l=-\infty}^{\infty} \Phi_{x1y1}\left(v_1 - \frac{2\pi}{q_1} i, v_2 - \frac{2\pi}{q_2} l\right) \times \operatorname{sinc}\left[\frac{q_1}{2}\left(v_1 - \frac{2\pi}{q_1} i\right)\right] \operatorname{sinc}\left[\frac{q_2}{2}\left(v_2 - \frac{2\pi}{q_2} l\right)\right]. \quad (2.37)$$

If $d_1(t)$ and $d_2(t)$ are independent of $x(t)$ and $y(t)$ and of each other, then the following relation holds true:

$$\Phi_{x1y1}(v_1, v_2) = \Phi_{xy}(v_1, v_2)\ \Phi_{d1}(v_1)\ \Phi_{d2}(v_2), \quad (2.38)$$

where $\Phi_{x1y1}(v_1, v_2)$ is the joint characteristic function of the signals x_1 and y_1, and $\Phi_{d1}(v_1)$, $\Phi_{d2}(v_2)$ are the characteristic functions of the signals d_1, d_2. Substituting (2.38) into (2.37) and differentiating (2.36) with respect to v_1 and v_2, we obtain an expression for the expected value of the correlator's output signal (Chang and Moore, 1970; Lal-Jadziak, 2001b; 2001c):

$$E\left[\tilde{R}^d_{xy}(k, M)\right] = R_{xy}(k) + \frac{q_1}{2\pi} \sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \frac{(-1)^i}{i} \Phi_{d1}\left(-\frac{2\pi}{q_1} i\right) \left.\frac{\partial \Phi_{xy}(v_1 - 2\pi i/q_1, v_2)}{\partial v_2}\right|_{(v_1, v_2)=(0,0)}$$

$$+ \frac{q_2}{2\pi} \sum_{\substack{l=-\infty \\ l\neq 0}}^{\infty} \frac{(-1)^l}{l} \Phi_{d2}\left(-\frac{2\pi}{q_2} l\right) \left.\frac{\partial \Phi_{xy}(v_1, v_2 - 2\pi l/q_2)}{\partial v_1}\right|_{(v_1, v_2)=(0,0)}$$

$$- \frac{q_1 q_2}{4\pi^2} \sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \sum_{\substack{l=-\infty \\ l\neq 0}}^{\infty} \frac{(-1)^{i+l}}{il} \Phi_{d1}\left(-\frac{2\pi}{q_1} i\right) \Phi_{d2}\left(-\frac{2\pi}{q_2} l\right)$$

$$\times \Phi_{xy}\left(-\frac{2\pi}{q_1} i, -\frac{2\pi}{q_2} l\right). \quad (2.39)$$

From the formula (2.39), it follows that the expected value of the crosscorrelation function estimator determined on the basis of the signals $x_{1q1}(i\Delta t)$ and $y_{1q2}(i\Delta t)$ is equal to the real correlation function (of the signals $x(t)$ and $y(t)$) when the auxiliary signals $d_1(t)$ and $d_2(t)$ satisfy the conditions

$$\Phi_{d1}\left(\frac{2\pi}{q_1} i\right) = 0, \qquad \forall i \neq 0, \quad (2.40a)$$

$$\Phi_{d2}\left(\frac{2\pi}{q_2} l\right) = 0, \qquad \forall l \neq 0, \quad (2.40b)$$

i.e. they assume the value 0 for the arguments $2\pi i/q_1$ and $2\pi l/q_2$ for $\forall i \neq 0$ and $\forall l \neq 0$.

Satisfying the Widrow theorems expressed by the formulae

$$\Phi_{d1}(v_1) = 0, \qquad \text{for} \quad |v_1| > 2\pi/q_1 - \varepsilon, \tag{2.41a}$$

$$\Phi_{d2}(v_2) = 0, \qquad \text{for} \quad |v_2| > 2\pi/q_2 - \varepsilon, \tag{2.41b}$$

where ε is an infinitesimally small positive number, leads to ensuring the conditions (2.40a) and (2.40b). If the above conditions are not satisfied, then the bias of the estimator (2.3) can be estimated on the basis of the formula

$$\begin{aligned} b\left[\tilde{R}^d_{xy}(k,M)\right] = {} & \frac{q_1}{2\pi} \sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \frac{(-1)^i}{i} \Phi_{d1}\left(-\frac{2\pi}{q_1} i\right) \left.\frac{\partial \Phi_{xy}(v_1 - 2\pi i/q_1, v_2)}{\partial v_2}\right|_{(v_1,v_2)=(0,0)} \\ & + \frac{q_2}{2\pi} \sum_{\substack{l=-\infty \\ l\neq 0}}^{\infty} \frac{(-1)^l}{l} \Phi_{d2}\left(-\frac{2\pi}{q_2} l\right) \left.\frac{\partial \Phi_{xy}(v_1, v_2 - 2\pi l/q_2)}{\partial v_1}\right|_{(v_1,v_2)=(0,0)} \\ & - \frac{q_1 q_2}{4\pi^2} \sum_{\substack{i=-\infty \\ i\neq 0}}^{\infty} \sum_{\substack{l=-\infty \\ l\neq 0}}^{\infty} \frac{(-1)^{i+l}}{il} \Phi_{d1}\left(-\frac{2\pi}{q_1} i\right) \Phi_{d2}\left(-\frac{2\pi}{q_2} l\right) \\ & \times \Phi_{xy}\left(-\frac{2\pi}{q_1} i, -\frac{2\pi}{q_2} l\right). \end{aligned} \tag{2.42}$$

In Section 2.3, conditions are formulated which signals subjected to simultaneous quantization and sampling should satisfy (cf. (2.27a), (2.27b) and (2.28)) so that the estimator (2.2) of the correlation function produced on their basis will be unbiased. If these conditions are not satisfied, then the application of appropriate dither signals may lead to the elimination of (in practice – a decrease in) the bias.

The selection of dither signals and the evaluation of their influence on the estimation quality of correlation functions were dealt with by the author in (Lal-Jadziak, 1999). There, she analyzed, among other things, the bias of the mean square value estimator of a signal (the mean square value is the value of the autocorrelation function for the argument 0) following the application of Gaussian dither. The presence of this dither – despite satisfying neither the condition (2.40a) nor (2.41a) – enables a decrease in the estimator bias.

If dither signals satisfy the conditions expressed by the formulae (2.40a) or (2.41a), as well as (2.40b) or (2.41b) (therefore when the estimator (2.3) is unbiased), then the mean square error (equal to the variance) can be expressed by the relation (Chang and Moore, 1970):

$$\begin{aligned} e^2_{\tilde{R}^d_{xy}}(k,M) = {} & Var\left[\tilde{R}^d_{xy}(k,M)\right] \\ = {} & \frac{1}{M}\Big\{E\big[x^2_{1q1}(0)y^2_{1q2}(k)\big] - E\big[x^2(0)y^2(k)\big]\Big\} \\ & + \frac{1}{M^2} \sum_{i=0}^{M-1} \sum_{l=0}^{M-1} E\big[x(i)y(i+k)x(l)y(l+k)\big] - R^2_{xy}(k). \end{aligned} \tag{2.43}$$

The calculation of the error (2.43) in general is not possible, for its level depends not only on the number of samples used for estimation, the A/D conversion resolution, and the kind and level of dither signals, but also on the class of signals being converted and the delay between them. We can note, however, that its first term corresponding to the share of quantization with dither in the total mean square error is inversely proportional to the number M of samples taken. It means that increasing this number (e.g. via the application of oversampling) leads to an improvement in the accuracy of the estimator in question.

The results of the above deliberations can be extended to the issue of autocorrelation function estimation (by the so-called method via crosscorrelation, in other words – a two-channel method), assuming $x = x(0)$, $y = x(k)$, $x_{1q1} = x_{1q1}(0)$, $y_{1q2} = x_{1q2}(k)$ in the formulae (2.39), (2.42), (2.43).

Digital estimation of an autocorrelation function with dither was presented by the author in (Lal-Jadziak, 1999; 2000). It turns out that even under the reconstruction conditions, the autocorrelation function estimator is biased by a component coming from the dither applied, which follows from the formula

$$E\left[\tilde{R}_x^d(k,M)\right] = \begin{cases} R_x(k) + R_d(k) + \dfrac{q^2}{12} & (k=0), \\ R_x(k) + R_d(k) & (k \neq 0), \end{cases} \tag{2.44}$$

in which $R_d(k)$ is the dither autocorrelation function. In light of the above, such a realization of autocorrelation seems to be dubious, especially since, in practice, controlling dither parameters is rather difficult (Wagdy and Goff, 1994).

2.5. Analysis of variance component coming from quantization with dither

If dither signals satisfy the conditions expressed by the formulae (2.40a) or (2.41a), as well as (2.40b) or (2.41b) (therefore when the estimator (2.3) is unbiased), then the variance (equal to the mean square error) can be expressed by the relation (Lal-Jadziak, 2003):

$$Var\left[\tilde{R}_{xy}^d(k,M)\right] = Var_1\left[\tilde{R}_{xy}^d(k,M)\right] + Var_2\left[\tilde{R}_{xy}^d(k,M)\right], \tag{2.45}$$

where

$$Var_1\left[\tilde{R}_{xy}^d(k,M)\right] = \frac{1}{M^2}\sum_{i=0}^{M-1}\sum_{l=0}^{M-1} E\big[x(i)y(i+k)x(l)y(l+k)\big] - R_{xy}^2(k) \tag{2.46}$$

is the variance (equal to the mean square error) of the correlator, in which neither quantization nor dither signals occur, and

$$Var_2\left[\tilde{R}_{xy}^d(k,M)\right] = \frac{1}{M}\Big\{E\big[x_{1q}^2(0)y_{1q}^2(k)\big] - E\big[x^2(0)y^2(k)\big]\Big\} \tag{2.47}$$

is the variance component coming from quantization with a dither signal. From the formula (2.47) it follows that it is inversely proportional to the number M of samples used for the determination of the estimator.

Type A measurement uncertainty is determined by the standard deviation, which is the square root of the variance. Therefore, if in the expression (2.45) the component resulting from A/D conversion were dominant, then a k-fold increase in the number of samples would cause a $\sqrt{k}$-fold decrease in the uncertainty. The component $Var_1[\tilde{R}^d_{xy}(k,M)]$ is dealt with by the author in (Lal-Jadziak, 2001b).

Let us submit for analysis the component $Var_2[\tilde{R}^d_{xy}(k,M)]$ resulting from quantization with a dither signal. Since the joint $(m+n)$-th order moment of the signals x and y can be determined by differentiating the characteristic function according to (2.9), then the relation (2.47) can be expressed in the form

$$Var_2\left[\tilde{R}^d_{xy}(k,M)\right] = \frac{1}{M}\left\{\frac{\partial^4 \Phi_{x1q1y1q2}(v_1,v_2)}{\partial v_1^2 \partial v_2^2} - \frac{\partial^4 \Phi_{xy}(v_1,v_2)}{\partial v_1^2 \partial v_2^2}\right\}\Bigg|_{(v_1,v_2)=(0,0)}. \tag{2.48}$$

Taking into account the relations (2.37) as well as (2.38), the function $\Phi_{x1q1y1q2}(v_1,v_2)$ can be expressed as

$$\Phi_{x1q1y1q2}(v_1,v_2) = \sum_{i=-\infty}^{\infty}\sum_{l=-\infty}^{\infty} \Phi_{xy}\left(v_1-\frac{2\pi}{q_1}i, v_2-\frac{2\pi}{q_2}l\right)\Phi_{d1}\left(v_1-\frac{2\pi}{q_1}i\right)\Phi_{d2}\left(v_2-\frac{2\pi}{q_2}l\right)$$
$$\times \operatorname{sinc}\left[\frac{q_1}{2}\left(v_1-\frac{2\pi}{q_1}i\right)\right]\operatorname{sinc}\left[\frac{q_2}{2}\left(v_2-\frac{2\pi}{q_2}l\right)\right]. \tag{2.49}$$

Substituting the relation (2.49) into the formula (2.48) and taking into account that

i) the conditions (2.40a) or (2.41a) as well as (2.40b) or (2.41b) are satisfied,

$$\text{ii)}\quad \Phi_{d1}\left(\frac{2\pi}{q_1}i\right) = 1 \quad \text{for} \quad i=0 \quad \text{as well as} \quad \Phi_{d2}\left(\frac{2\pi}{q_2}l\right) = 1 \quad \text{for} \quad l=0, \tag{2.50}$$

$$\text{iii)}\qquad \left.\frac{\mathrm{d}\Phi_{d1}\left(v_1-\frac{2\pi}{q_1}i\right)}{\mathrm{d}v_1}\right|_{v_1=0} \neq \infty, \qquad \left.\frac{\mathrm{d}^2\Phi_{d1}\left(v_1-\frac{2\pi}{q_1}i\right)}{\mathrm{d}v_1^2}\right|_{v_1=0} \neq \infty, \tag{2.51}$$

$$\text{iv)}\qquad \left.\frac{\mathrm{d}\Phi_{d2}\left(v_2-\frac{2\pi}{q_2}l\right)}{\mathrm{d}v_2}\right|_{v_2=0} \neq \infty, \qquad \left.\frac{\mathrm{d}^2\Phi_{d2}\left(v_2-\frac{2\pi}{q_2}l\right)}{\mathrm{d}v_2^2}\right|_{v_2=0} \neq \infty, \tag{2.52}$$

an expression for the component $Var_2[\tilde{R}^d_{xy}(k,M)]$ can be obtained in the form (Lal-Jadziak, 2003):

$$Var_2\left[\tilde{R}^d_{xy}(k,M)\right]$$
$$= \frac{1}{M}\left\{\Phi_{xy}(v_1,v_2)\left[\frac{\mathrm{d}^2}{\mathrm{d}v_1^2}\Phi_{d1}(v_1)\frac{\mathrm{d}^2}{\mathrm{d}v_2^2}\Phi_{d2}(v_2) - \frac{q_2^2}{3}\frac{\mathrm{d}^2}{\mathrm{d}v_1^2}\Phi_{d1}(v_1) - \frac{q_1^2}{3}\frac{\mathrm{d}^2}{\mathrm{d}v_2^2}\Phi_{d2}(v_2) + \frac{q_1^2 q_2^2}{9}\right]\right.$$

$$+\frac{\partial\Phi_{xy}(v_1,v_2)}{\partial v_1}\left[2\frac{\mathrm{d}^2}{\mathrm{d}v_2^2}\Phi_{d2}(v_2)\frac{\mathrm{d}}{\mathrm{d}v_1}\Phi_{d1}(v_1)-2\frac{q_2^2}{3}\frac{\mathrm{d}}{\mathrm{d}v_1}\Phi_{d1}(v_1)\right]$$

$$+\frac{\partial\Phi_{xy}(v_1,v_2)}{\partial v_2}\left[2\frac{\mathrm{d}^2}{\mathrm{d}v_1^2}\Phi_{d1}(v_1)\frac{\mathrm{d}}{\mathrm{d}v_2}\Phi_{d2}(v_2)-2\frac{q_1^2}{3}\frac{\mathrm{d}}{\mathrm{d}v_2}\Phi_{d2}(v_2)\right]$$

$$+\frac{\partial^2\Phi_{xy}(v_1,v_2)}{\partial v_1^2}\left[\frac{\mathrm{d}^2}{\mathrm{d}v_2^2}\Phi_{d2}(v_2)-\frac{q_2^2}{3}\right]+\frac{\partial^2\Phi_{xy}(v_1,v_2)}{\partial v_2^2}\left[\frac{\mathrm{d}^2}{\mathrm{d}v_1^2}\Phi_{d1}(v_1)-\frac{q_1^2}{3}\right]$$

$$+\frac{\partial^2\Phi_{xy}(v_1,v_2)}{\partial v_1\partial v_2}\left[4\frac{\mathrm{d}}{\mathrm{d}v_1}\Phi_{d1}(v_1)\frac{\mathrm{d}}{\mathrm{d}v_2}\Phi_{d2}(v_2)\right]$$

$$+\frac{\partial^3\Phi_{xy}(v_1,v_2)}{\partial v_1\partial v_2^2}\left[2\frac{\mathrm{d}}{\mathrm{d}v_1}\Phi_{d1}(v_1)\right]+\frac{\partial^3\Phi_{xy}(v_1,v_2)}{\partial v_1^2\partial v_2}\left[2\frac{\mathrm{d}}{\mathrm{d}v_2}\Phi_{d2}(v_2)\right]$$

$$+\frac{q_1q_2}{\pi^2}\sum_{\substack{i=-\infty\\ i\neq 0}}^{\infty}\sum_{\substack{l=-\infty\\ l\neq 0}}^{\infty}\frac{(-1)^{i+l}}{il}\Phi_{xy}\left(v_1-\frac{2\pi}{q_1}i,v_2-\frac{2\pi}{q_2}l\right)$$

$$\times\left[\frac{\mathrm{d}}{\mathrm{d}v_1}\Phi_{d1}\left(v_1-\frac{2\pi}{q_1}i\right)\frac{\mathrm{d}}{\mathrm{d}v_2}\Phi_{d2}\left(v_2-\frac{2\pi}{q_2}l\right)\right]\Bigg\}\Bigg|_{(v_1,v_2)=(0,0)}. \tag{2.53}$$

The relation (2.53) is complicated, and the analyses carried out on its basis so far have made it possible to formulate fairly scant conclusions (Lal-Jadziak, 2003).

In this situation, further research in the field of the estimation of correlation functions on the basis of digital signal representation is conducted in two ways: using analytical models of characteristic functions (Sienkowski, 2006) or applying a virtual correlator model (Kawecka, 2006; Lal-Jadziak and Kawecka, 2006).

2.6. Experimental research results and their assessment

Taking into account the complexity of analytical models of bias and variance, a piece of software, called a virtual correlator model, was designed, which then can be used to assess the quality of estimation (Kawecka, 2006; Lal-Jadziak and Kawecka, 2006).

To realize the model of the correlator, the environment of *LabWindows*®, version 7.0, by *National Instruments*, was applied. In the program, original procedures to define correlation functions were used, because the available *LabWindows*® functions lead to unreliable results.

An important stage of the experiment was the comparison of the results obtained by means of the correlator with those calculated on the basis of analytical models (Lal-Jadziak and Kawecka, 2006). The research was conducted for a signal with Gaussian dither, and the calculations were done by means of the *Mathcad* program.

In Fig. 2.6 there are shown example research results obtained from the experiment $\delta(\sigma/q)$, as well as the mathematical analyses $\delta_M(\sigma/q)$.

As follows from the diagrams, the relative bias of estimator decreases with an increase in the ratio of the standard deviation of a dither signal to the quantization

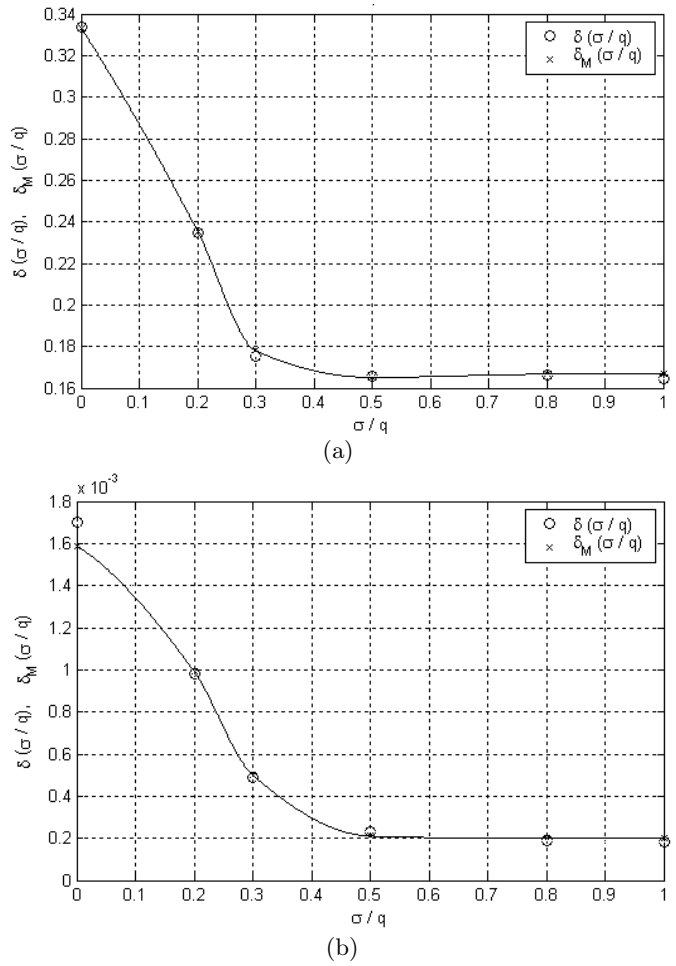


Fig. 2.6. Relative bias of the mean square value estimator of the harmonic signal with Gaussian dither as a function of σ/q: (a) $A/q = 1$, (b) $A/q = 29$

step of the converter, and the results obtained in the experiment are convergent with those calculated with the *Mathcad* program.

The values of the estimator relative bias for $A/q = 1$ and $\sigma/q = 0,5$[‡] are, respectively,

$$\delta\,(0,5) = 1,65 \cdot 10^{-1}, \quad \delta_M\,(0,5) = 1,65 \cdot 10^{-1},$$

whereas for $A/q = 29$:

$$\delta\,(0,5) = 2,31 \cdot 10^{-4}, \quad \delta_M\,(0,5) = 2,11 \cdot 10^{-4}.$$

[‡] Gaussian dither is optimally selected if the ratio of σ/q equals 0.5 (Domańska, 2005; Koeck, 2001).

It is planned to apply the correlator model to research on the influence of dither signals on the variance of correlation function estimators responsible for type A uncertainty.

Unfortunately, for multibit A/D converters the results obtained by means of the virtual correlator model and the *Mathcad* program differ. The source of the differences are the inaccuracies of the numeric calculations.

2.7. Conclusions

The level of estimation errors in digital correlation measurements depends not only on A/D resolution, the kind and level of dither signals, but also on the probabilistic characteristics of the investigated signals.

The fulfillment of the conditions (2.27a) and (2.27b) by the joint characteristic function of the signals x and y and its derivatives leads to a lack of bias of the digital crosscorrelation estimator. The Widrow reconstruction condition (2.28) leads to ensuring (2.27a), (2.27b).

The fulfillment of the conditions (2.32a) and (2.32b) by the signal x characteristic function and its derivatives leads – after taking the correction $q^2/12$ into account – to the elimination of the bias of the digital autocorrelation function estimator for $k\Delta t = 0$, i.e. of the mean square value of the signal. The Widrow reconstruction condition (2.10) leads to ensuring (2.32a), (2.32b).

It can be stated that the condition of the non-occurrence of the bias (and therefore of the error systematic component) of the digital crosscorrelation estimator, or of the digital autocorrelation estimator determined via crosscorrelation, is the fulfillment by the dither signals of the following conditions: the zero mean assumption, their statistical independence of the measured signals and of each other, and the fulfillment by their characteristic functions of the condition (2.40a) or (2.41a), as well as (2.40b) or (2.41b). If dither signals satisfy the above assumptions, then the mean square error component due to quantization with dither is inversely proportional to the number of samples taken. It means that increasing this number (e.g. through the application of oversampling) leads to an improvement in estimation accuracy. Signals completely satisfying the conditions mentioned do not exist, and this is why digital conversion of signals always causes a bias of correlation function estimators. The bias levels can be evaluated using the formulae (2.26), (2.31) or (2.42).

Finally, it is worth emphasizing that we can affect correlation measurement accuracy not only by exerting influence on the properties of A/D converters (e.g. selecting an appropriate quantization step), but also by adding appropriate dither signals. The application of dither signals leads to an improvement in quantizing reconstruction and can – after taking Sheppard corrections into account – allow obtaining unbiased estimators of correlation functions. It is a way of eliminating, in practice – decreasing, the bias without knowing its mathematical model.

The application of a suitable dither signal may cause an increase in the estimator variance, which will result in an increase in the scatter of the measurement results. The variance component resulting from A/D conversion with dither is inversely proportional to the number M of samples used for estimation. Type A measurement uncertainty is determined by the standard deviation, which is the square root of the

variance. Therefore, if in the expression (2.45) the component resulting from A/D conversion were dominant, then a k-fold increase in the number of samples would cause a $\sqrt{k}$ fold decrease in the uncertainty.

Analytical models of estimation errors of digitally determined correlation functions are of limited practical use because of their great complexity. In this situation, for research reasons, in the environment of *LabWindows*® by *National Instruments*, a virtual correlator model was designed. For A/q values not exceeding 30, the results obtained with it are convergent with those obtained on the basis of analytical models.

References

Bendat J.S. and Piersol A.G. (1986): *Random Data: Analysis and Measurement Procedures*. — New York: John Wiley.

Bendat J.S. and Piersol A.G. (1993): *Engineering Applications of Correlation and Spectral Analysis*. — New York: John Wiley.

Chang K.Y. and Moore A.D. (1970): *Modified digital correlator and its estimation errors*. — IEEE Trans. Information Theory, pp. 699–706.

Domańska A. (1995): *Influencing the reliability in measurement systems by the application of A-D conversion with dither signal*. — Monographs, No. 308, Wydawnictwo Politechniki Poznańskiej, (in Polish).

Domańska A. (2005): *The impact of the randomization of the quantization error on the accuracy of measuring systems applying a digital measuring algorithm*. — Metrology and Measurement Systems, Vol. XII, No. 2, pp. 175–182.

Kawecka E. (2006): *The use of virtual correlator model for the evaluation of the uncertainty of digital correlation estimators*. — Pomiary, Automatyka, Kontrola, No. 6, Special issue, pp. 80–82, (in Polish).

Kawecka E. and Lal-Jadziak J. (2004): *The influence of quantizing on the accuracy of Gaussian signals moments estimation*. — Pomiary, Automatyka, Robotyka, Nos. 7–8, pp. 154–158, (in Polish).

Koeck P. (2001): *Quantization errors in averaged digitized data*. — Signal Processing, Vol. 81, pp. 345–356.

Korn G.A. (1966): *Random Process Simulation and Measurements*. — New York: McGraw-Hill Book Comp.

Lal-Jadziak J. (1999): *Accuracy of correlation measurements by A-D conversion with dither*. — Metrologia i Systemy Pomiarowe, Vol. VI, No. 1–2, pp. 27–46, (in Polish).

Lal-Jadziak J. (2000): *Dither in digital correlation measurements*. — Proc. XVI *IMEKO World Congress*, Vienna, Austria, Vol. IX, pp. 105–110.

Lal-Jadziak J. (2001a): *Accuracy in determination of correlation functions by digital methods*. — Metrology and Measurement Systems, Vol. VIII, No. 2, pp. 153–163.

Lal-Jadziak J. (2001b): *Influencing Accuracy in Correlation Measurements*. — Monograph, No. 101, Wydawnictwo Politechniki Zielonogórskiej, (in Polish).

Lal-Jadziak J. (2001c): *The influence of quantizing on the accuracy of correlation functions estimation*. — Metrology and Measurement Systems, Vol. VIII, No. 1, pp. 25–40, (in Polish).

Lal-Jadziak J. (2003): *Bias and variance of crosscorrelation function estimator determined on the basis of signals resulting from A-D conversion with dither.* — Metrology and Measurement Systems, Vol. 10, No. 4, pp. 341–351.

Lal-Jadziak J. and Kawecka E. (2006): *Evaluation of estimation accuracy of correlation functions with use of virtual correlator model.* — Pomiary, Automatyka, Kontrola, No. 6, pp. 16–18, (in Polish).

Sienkowski S. (2006): *Modelling characteristic functions of determined and random signals in LabWindows.* — Proc. 1-st Int. Conf. *Young Reserchers in Computer Science, Control, Electrical Engineering and Telecommunications, ICYR*, University of Zielona Góra, Poland, Abstracts, pp. 56–57.

Sripad B. and Snyder D. (1977): *A necessary and sufficient condition for quantization errors to be uniform and white.* — IEEE Trans. Acoust. Speech, Signal Process., Vol. ASSP-25, No. 5, pp. 442–448.

Wagdy M.F. (1989): *Effect of various dither form on quantization errors of ideal A/D converters.* — IEEE Trans. Instrum. Meas., Vol. 38, No. 4, pp. 850–855.

Wagdy M.F. and Goff M. (1994): *Linearizing average transfer characteristics of ideal ADC's via analog and digital dither.* — IEEE Trans. Instrum. Meas., Vol. 43, No. 2, pp. 146–150.

Widrow B. (1956): *A study of rough amplitude quantization by means of Nyquist sampling theory.* — IRE Trans. Circuit Theory, Vol. 3, No. 4, pp. 266–276.

Widrow B. and Kollar I. (2006): *Quantization Noise – A Book on Uniform and Floating-Point Quantization.* — `http://www.mit.bme.hu/books/quantization/`.

Widrow B., Kollar I. and Liu M.-C. (1996): *Statistical theory of quantization.* — IEEE Trans. Instrum. Meas., Vol. 45, No. 2, pp. 353–361.

Wojnar A. (1988): *Signal Theory.* — Warsaw: Wydawnictwa Naukowo-Techniczne, (in Polish).

Chapter 3

COMPENSATION OF CONDITIONING SYSTEM IMPERFECTIONS IN MEASURING SYSTEMS

Leszek FURMANKIEWICZ*, Mirosław KOZIOŁ*, Radosław KŁOSIŃSKI*

3.1. Introduction

Nowadays, most measuring devices use information in the form of digital samples of a signal to execute their tasks. In spite of the approach, analog parts still exist in these devices. Their usage is necessary to:

- acquire information (sensors),
- change one form of energy to another (transducers),
- change the value of the voltage to the level accepted by the analog-to-digital (A/D) converter,
- remove some components from the signal spectrum according to the sampling theory (antialiasing filter).

In most cases, these parts do not have desirable static and/or dynamic characteristics, which adversely influence measurement accuracy. Therefore, in common sense, they can be called distorting systems.

Before the samples are used in a measuring process, a correction process should be carried out. The compensation of the distorting system influence on the signal carrying the information is its main purpose. Its result in the time domain should be the reconstruction of the signal shape. It should appear as signal spectrum reconstruction in the frequency domain as well.

The most comfortable way is to carry out the correction process in the discrete rather than analog domain because of a microprocessor unit in the structure of measuring devices which executes some measuring tasks. Therefore, the implementation of the correction can be realized without constructional changes of this device.

* Institute of Electrical Metrology
e-mails: {L.Furmankiewicz, M.Koziol, R.Klosinski}@ime.uz.zgora.pl

The distortion type depends on the distorting system. Amplitude and delay distortions appear if the distorting system is linear. Additional distortions can appear if the system is non-linear, (a new harmonic generation, signal spectrum alteration, shifting in the frequency domain).

In many cases, a distorting system can be modeled as a linear time-invariant (LTI) system, where the input signal is processed by the system using the convolution operation. To extract the input signal from the output signal, the deconvolution operation in the time domain has to be carried out. This approach is used widely in the reconstruction of spectrometric (Miekinia *et al.*, 1997) and biomedical data (Merino *et al.*, 2005), and image restoration (Dabóczi and Bakó, 2001) by iterative and/or regularized methods. The disadvantage of iterative methods is the necessity for signal processing, which has to be of a finite length. Another disadvantage is long computational time. Sometimes the convergence of an algorithm can be improved (Szczecinski and Barwicz, 1997). However, the first drawback still makes it impossible to use this kind of algorithms in real-time reconstruction of signals.

In some circumstances, a non-linear system can be modeled as a linear time-invariant system. Such an approach is presented in the first part of this chapter, where a current transformer is modeled by the linear time-invariant system. The LTI model is used to determine, on the basis of measurements, the magnitude and phase error characteristics. The reconstruction of the primary current spectrum is achieved on the basis of these characteristics. This approach is an example of the correction process in the frequency domain.

In practice, a correction system has to perform the correction process and also meet some requirements. One of them is stability, which depends on the zero locations of the distorting system and sometimes on the design process of the correction system as well. The middle part of the chapter is devoted to this problem.

The above-mentioned approach, where a non-linear distorting system is modeled as a linear system, cannot be applied if it is strongly non-linear. In general, the description of non-linear systems is complex but, if signals are periodic, a non-linear dynamical system can be represented by a set of linear time-varying approximations, which is presented in the last part of this chapter.

3.2. Frequency error correction in power measurements

3.2.1. Frequency linear model of input circuits

In many situations, the assumption of a linear model of input circuits is sufficient to correct signal distortions. In the case of periodic signals, it is convenient to use the spectrum domain, which is frequently used in the processing of measurement signals. Such correction can be realized on the basis of the knowledge of frequency errors of input circuits.

Assuming, that the input circuits of measuring devices are linear transducers, for the purpose of the description of their dynamic properties (Sydenham, 1983), we can apply spectral transmittance, defined as

$$K(j\omega) = \frac{Y(j\omega)}{X(j\omega)}, \tag{3.1}$$

where $Y(j\omega)$ and $X(j\omega)$ are Fourier transforms of the transducer output and input signals, respectively.

When assuming that the input signal of the transducer is a sinusoidal wave, we can obtain the output signal amplitude Y_m based on transducer transmittance:

$$Y_m = |K(j\omega) X(j\omega)|. \tag{3.2}$$

The module of spectral transmittance equals the relation of the output signal amplitude Y_m and the input signal amplitude X_m:

$$|K(j\omega)| = \frac{Y_m}{X_m}. \tag{3.3}$$

The module dependence on frequency is called the amplitude characteristic. The argument of the spectral transmittance

$$\varphi(\omega) = \arg[K(j\omega)] \tag{3.4}$$

defines the phase shift between the output and input signals of the transducer. The phase shift dependence on frequency is called the phase characteristic.

It results from the above equations that both the relationship of the output signal amplitude Y_m and the input signal amplitude X_m and the phase shift depend on the frequency of the input signal and transducer properties described by spectral transmittance.

The input circuits of power measurement instruments should have the constant amplitude characteristic in the transmitted frequency band and either the zero or linear phase characteristic. The characteristics of real input circuits differ from these specifications. From that, amplitude dynamic errors and phase dynamic errors are defined. Assuming that the frequency range of the instrument is determined within the frequency range from $\omega = 0$ to $\omega = \omega_k$, which means that a constant value of the signal is processed, in the case of an ideal input circuit, the spectral transmittance is of the form

$$K = K(j\omega)\,|_{\omega=0}\,. \tag{3.5}$$

The relationship (3.5) means that the amplitudes of the signals ratio is a constant value. The dynamic error $\Delta(j\omega)$ is defined as a difference between the spectral response of a real input circuit and an ideal input circuit:

$$\Delta(j\omega) = Y(j\omega) - KX(j\omega). \tag{3.6}$$

The magnitude of the dynamic error Δ_m is called the magnitude error or the amplitude error:

$$\Delta_m = |\Delta(j\omega)|. \tag{3.7}$$

The argument of the dynamic error is called the phase error:

$$\gamma = \arg(\Delta(j\omega)). \tag{3.8}$$

In practice, it is convenient to use the relative dynamic error δ, which may be written as

$$\delta = \frac{\Delta_m}{KX(j\omega)} = \left|\frac{\Delta(j\omega)}{KX(j\omega)}\right| = \left|\frac{Y(j\omega) - KX(j\omega)}{KX(j\omega)}\right| = \left|\frac{K(j\omega) - K}{K}\right|. \tag{3.9}$$

The dynamic errors of the input circuit used in power systems transducers follow from the fact that their equivalent circuit diagrams, except for resistors, contain elements whose reactance is frequency-depended. The elements are capacitors representing parasitic capacitance or inter-winding and winding capacitance in transformers as well as coils representing, among other things, parasitic inductance.

The correcting method is based on the knowledge of frequency magnitude and phase errors. The frequency characteristics of magnitude errors can be obtained by measuring the output and input signals of the input circuit and calculating them according to (3.9). The frequency characteristics of phase errors can be obtained by measuring the phase shift between the output and input signals of the input circuit.

3.2.2. Active power measurement errors

The active power of periodic non-sinusoidal signals P equals the sum of active harmonic powers contained in the processed signals, including the zero harmonic

$$P = U_0 I_0 + \sum_{k=1}^{N} U_k I_k \cos(\varphi_k), \tag{3.10}$$

where $U_0,\ I_0$ are the constant components of the voltage and the current, $U_k,\ I_k$ are root-mean-square values of the voltage and current harmonics, φ_k is the phase shift angle between the k-th harmonic of the voltage and the current.

The active power P_δ after processing through the input voltage and current circuits is loaded with errors of such circuits and can be expressed by means of a component including a constant component P_{δ_0} and components deriving from the higher harmonics

$$P_\delta = P_{\delta 0} + \sum_{k=1}^{N} \frac{1}{K_{Un}} (1 + \delta_{Uk}) \frac{1}{K_{In}} (1 + \delta_{Ik}) U_k I_k \cos(\varphi_k + \gamma_{Uk} - \gamma_{Ik}), \tag{3.11}$$

where

$$P_{\delta 0} = \frac{1}{K_{Un}} (1 + \delta_{U0}) \frac{1}{K_{In}} (1 + \delta_{I0}) U_0 I_0. \tag{3.12}$$

The following denotations of input circuit parameters were assumed in (3.11) and (3.12): K_{Un} and K_{In} represent nominal processing factor, δ_{Uk} and δ_{Ik} are module errors, γ_{Uk} and γ_{Ik} are phase errors of the input voltage and current circuits.

Assuming that P denotes the power value determined for the circuits of the zero module and angle errors, the power processing error δ_P caused by the input circuits can be determined as follows:

$$\delta_P = \frac{P_\delta - P}{P}. \tag{3.13}$$

It was shown in (Furmankiewicz, 1999) that the error δ_P, in the case of transformer input circuits and distorted signals, can reach values between ten and twenty percent. Thus, it is purposeful to use the correction of errors caused by input circuits in the measurement of active power.

3.2.3. Error correction in power measurements

During the development of the correction method, it was assumed that the module errors δ_{Uk} and δ_{Ik} as well as the phase errors γ_{Uk} and γ_{Ik} of the input circuits are determined only once or periodically during the calibration of a measuring instrument. Furthermore, it is assumed that error value changes in time are omitted and, to a large extent, do not depend on external conditions, such as temperature, humidity, etc..

The correction method is intended for application to measuring instruments with digital signal processing. The input signals of voltage $u_Y(t)$ and current $i_Y(t)$ circuits are sampled. Then, the Discrete Fourier Transformation (DFT) algorithm is realized in each sequence N of the voltage u_{Yn} and current i_{Yn} samples taken at equal time intervals. The algorithm provides N equidistant spectrum samples $\underline{A}_k$ of a complex input sequence taken in the points of $\omega_k = 2\pi k/N$ pulsations:

$$A_k = \frac{1}{N}\sum_{n=0}^{N-1} u_{Yn} W_N^{kn}, \tag{3.14}$$

where $W_N = \exp(-j2\pi/N)$, and $k = 0, 1, 2, \ldots, N-1$.

Subsequently, root-mean-square values of the voltage signal harmonics U_{Yk} are calculated according to the formula

$$U_{Yk} = \sqrt{\frac{1}{2}\left(\mathrm{Re}^2(A_k) + \mathrm{Im}^2(A_k)\right)}, \tag{3.15}$$

where $k = 0, 1, 2, \ldots, N/2$, and the phases φ_{UYk} of the signal harmonics

$$\varphi_{UYk} = \arg(A_k) = \mathrm{arctg}\,\frac{\mathrm{Im}\,(A_k)}{\mathrm{Re}\,(A_k)}, \tag{3.16}$$

and $k = 1, 2, \ldots, N/2$. Root-square-mean values of I_{Yk} and the phase φ_{IYk} as well as the current signal harmonics are determined analogically.

The spectrum amplitude characteristic of the voltage signal occurring in the input of the input circuit, taking into account the amplitude errors, and the spectrum phase characteristic, taking into account the angle errors, are determined at the next stage:

$$U_{Xk} = \frac{1}{K_{Un}}\frac{U_{Yk}}{1+\delta_{Uk}}, \tag{3.17}$$

$$\varphi_{UXk} = \varphi_{UYk} - \gamma_{Uk}. \tag{3.18}$$

The spectrum amplitude characteristic of the current signal I_{Xk} and the spectrum phase characteristic φ_{IXk} are determined analogically.

Taking into account the above-described algorithm, the active power is calculated in the following manner:

$$P = \frac{1}{K_{Un}}\frac{1}{K_{In}}\left(\frac{U_0}{1+\delta_{U0}}\frac{I_0}{1+\delta_{I0}} + \sum_{k=1}^{N/2}\frac{U_{Yk}}{1+\delta_{Uk}}\frac{I_{Yl}}{1+\delta_{Ik}}\cos\left(\varphi_{UY} - \varphi_{IY} - \gamma_U + \gamma_I\right)\right). \tag{3.19}$$

3.2.4. Transformer error correction of input circuits

The above method was applied to the correction of transformer errors of input circuits used in power measurement transducers in the measurements of the active power of distorted signals. Transformer input circuits cause current distortions resulting from the non-linear characteristic of substitute resistance, loss resistance and magnetic inductance, the so-called non-linear distortions and linear distortions causing the processing of subsequent harmonics with different errors of module and phase shift. The linearity of input circuits in the work with input the signals, which do not cause iron core saturation, was assumed. The above assumption is possible despite the non-linear character of magnetizing component impedances causing the distortion of the magnetizing current. The distortion of the magnetizing current is not considerable and the share of higher harmonics does not exceed several percents (Koszmider *et al.*, 1985).

The sampling power transducer model was built and experimental tests were carried out in order to verify the rightness of the assumptions and the efficiency of the correction method. Transformer input circuits used in factory power transducers were used in the power transducer model. The National Instruments' data acquisition card AT-A2150 was used to process analog signals from the input circuits. The processing and correction algorithm was realized by a PC machine. The processing results of the power transducer realizing a software error correction was compared with the indications of the reference instrument. The tests were carried out for sinusoidal and non-sinusoidal signals. The non-sinusoidal test signals were generated digitally and contained 40 signal harmonics occurring in the thyristor voltage regulator. The test results proved the high efficiency of the proposed correction method. The Fig. 3.1 shows example test results of the correction efficiency for sinusoidal signals. Error ranges caused by power measurement uncertainty of the model instrument were marked on the characteristic showing the errors after the correction.

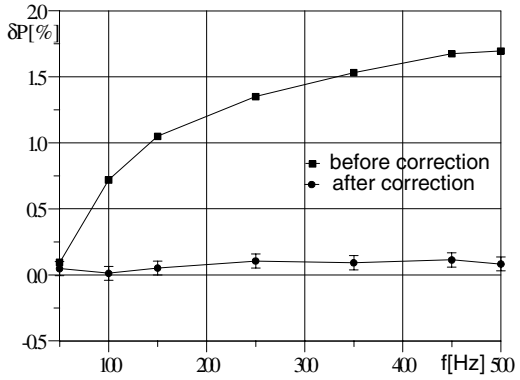


Fig. 3.1. Characteristics of the power measurements error before and after correction obtained in the results of an experiment on the sinusoidal signal $\varphi = 20^{\circ}$

In the case of the measurements of non-sinusoidal signals, the usage of correction caused error reduction from the level of between ten and twenty percent to the level

of $\pm 0,2$–0,3%. The detailed data of the performed experiments were shown in the paper (Furmankiewicz, 1998).

3.2.5. Error correction in the industrial transducer

The developed error correction method for input circuits can be used in a sampling measuring instrument of the classical structure, in which analog signal processing circuits (input circuits, measuring amplifiers, multiplexers and analog parts of the sample-and-hold circuit and analog-to-digital converter), the circuit of the analog-to-digital converter and the digital processing circuit are used. The general error model of the sampling transducer (Jakubiec, 2002) takes into account the following error sources: errors introduced by the circuits of analog signal processing, errors introduced by analog circuits of the analog-to-digital converter, and processing algorithm errors.

The software processing algorithm is the last element of the signal processing chain in the sampling measuring instrument, and it transfers errors introduced by the analog processing circuit and the analog-to-digital converter from the input to the output, and introduces own errors. The presented correction method requires the determination of the amplitude and phase spectrum for the voltage and current signals. This can be done by means of the discrete Fourier transform algorithm. Because of DFT algorithm calculation complexity, the Fast Fourier Transform (FFT) algorithm, which reduces the number of performed arithmetic operations thus shortening the calculation time, is used in practice. Signal microprocessors equipped with hardware solutions supporting the realization of the FFT algorithm are used in the construction of measuring instruments.

Signal microprocessors are still not commonly used in industrial power transducers. Circuit solutions realized on the basis of 8-, 16- or 32-bit microcontrollers, which do not possess specialized arithmetic modules, are dominant in the group of instruments. The realization times of several tens of points of FFT on floating point numbers (required processing band – to the 40th harmonic), including the determination of the amplitude spectrum by such microcontrollers, exceed values acceptable by users in the range of several seconds. In such a situation, the application of the FFT algorithm realized on integral numbers is a possible solution. Such an algorithm is used in the family of industrial electric power quality analyzers AJE from the *Metrol* Research and Development Centre in Zielona Góra, Poland. (Furmankiewicz and Rybski, 2003), to analyze the amplitude spectrum of the voltage and current signals. The algorithm realizes 1024-point FFT. The coefficients W_N^{kn} (3.14) from the $\pm 1,0$ range are represented by integral numbers from the range of ± 32767, while the coefficient values are written in the microprocessor memory, in the 1024-element array.

Statical own errors of the algorithm determining root-square-mean values of harmonics and harmonic phases were determined by a simulation test method in order to assess the efficiency of the software correction method on input circuits in industrial power transducers. The relative error value in the determination of the root-mean-square value of the harmonics δ_X contained in the measured signals was determined as a difference between the result yielded as a result of algorithm realization and a real value – consistent with the root-mean-square value definition, related to the nominal value. Furthermore, the absolute value of the phase measurement error $\Delta\varphi$ was also determined.

In the simulation tests, measurement conditions were shaped so that they corresponded to conditions occurring in industrial analyzers. Signals with the period equal to the measurement window width were used as an input function, which corresponds to the synchronous sampling method. The assumed sampling frequency makes it possible to take 128 samples in the basic harmonic period, also, transient digitization through the 12-bit analog-to-digital converter was assumed. The simulation tests were carried out for both mono-harmonic and poly-harmonic signals. In the case of the poly-harmonic signals, signals occurring in thyristor voltage regulators were assumed for the tests. In order to avoid the aliasing phenomenon, signals containing odd harmonics from 1 to 39 were used for the tests. The cut-off angle α was the parameter during tests. The detailed test results were presented in (Furmankiewicz, 2005). Figure 3.2 shows selected results of the simulation tests. Figure 3.2(a) shows the error characteristics of root-mean-square harmonic value measurements contained in the measured transient for two angles α, and Fig. 3.2(b) shows the error characteristics of phase measurements for two harmonics.

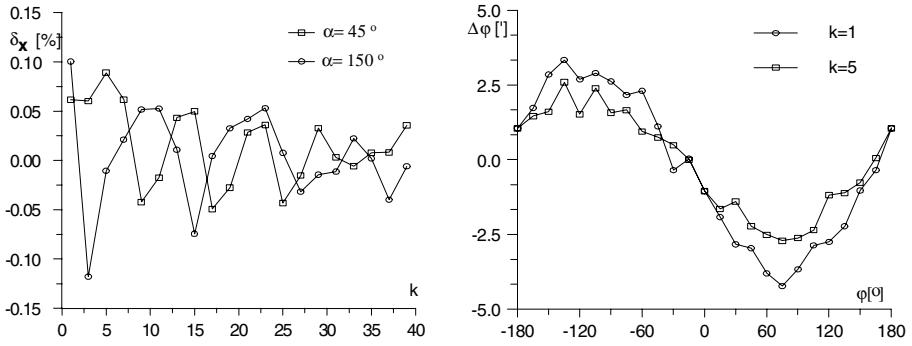


Fig. 3.2. Processing error of (a) magnitude spectrum, (b) phase angle

It can be stated on the basis of the yielded results that the FFT algorithm being analyzed introduces determination errors of root-mean-square harmonic values approaching the values of the ±0,15% range and determination errors of harmonic phases approaching the value ±5'. It was found that the main source of errors introduced by the FFT algorithm are digitalization errors of the algorithm coefficients occurring in the formula (3.14).

Assuming that the only source of power measurement errors are the amplitude δ_X and phase Δ_φ errors introduced by the processing algorithm FFT, the active power containing these errors can be expressed:

$$P_\delta = P_{\delta 0} + \sum_{k=1}^{N} \frac{1}{K_{Un}} (1 + \delta_{Xk}) \frac{1}{K_{In}} (1 + \delta_{Xk}) U_k I_k \cos(\varphi_k + \Delta\varphi_k), \tag{3.20}$$

where

$$P_{\delta 0} = \frac{1}{K_{Un}} (1 + \delta_{X0}) \frac{1}{K_{In}} (1 + \delta_{X0}) U_0 I_0. \tag{3.21}$$

The measurement error caused by the influence of the algorithm errors can be determined from the following formula:

$$\delta P_A = \frac{P_\delta - P}{P}, \tag{3.22}$$

where P is the real active power of the processed transients (the formula 3.10), and P_δ is the power yielded on the algorithm output (the formula 3.20). Substitution to the formula (3.22) and application of the justified reduction yields the constant component measurement error

$$\delta P_0 = 2\delta_X, \tag{3.23}$$

and measurement errors of the k-th power harmonic

$$\delta P_{Ak} = 2\delta_{Xk} + \Delta\varphi_k \tan\varphi_k. \tag{3.24}$$

It follows from (3.24) and the yielded simulation test results shown in Fig. 3.2 that already for low values of the phase shifts φ_k, the error δP_A can reach values from the range $\pm 0{,}5\%$, which considerably influence the efficiency of frequency error correction in active power measurements.

3.3. Quasi-inverse correction filters

Assuming that the model of a distorting system is known as the transfer function of a linear, causal and time-invariant system whose coefficients are real, and the digital correction filter is connected in series with the distorting system (Fig. 3.3), distortions in the signal $x[n]$ can be corrected by using a compensating system whose transfer function $G(z)$ should have the following form:

$$G(z) = \frac{1}{H(z)}. \tag{3.25}$$

It is necessary to mention that the problem of finding the transfer function $H(z)$ of the analog distorting system was not considered, because it is a broad separate problem. Processing an input signal $s[n]$ by series interconnection of these two systems, which in this case is equivalent to the identity system (i.e. system of the unit gain and zero phase response), it is possible to achieve perfect correction, i.e. $s[n] = y[n]$.

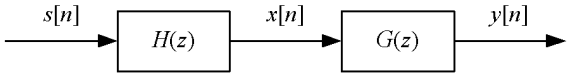


Fig. 3.3. Illustration of distortion compensation by an inverse system

In practice, a correction filter has to meet further requirements. One of them is stability, e.g. in the Bounded Input/Bounded Output (BIBO) sense. The decision which design method to apply to assure stability of the correction filter can be made by the analysis of the zeros location of the distorting systems. Three cases can be distinguished:

- all zeros of the transfer function $H(z)$ lie inside the unit circle (the distorting system is the minimum-phase system),

- at least one zero lies outside the unite circle (the distorting system is the non-minimum-phase system),
- at least one zero lies on the unit circle.

In the first case, the stable correction filter can be designed without any problems. Appling the equation (3.25) and the association of the region of convergence,

$$|z| > \max_k |q_k| \,, \tag{3.26}$$

to the obtained transfer function, where q_k is k-th zero of the transfer function $H(z)$, always yields a stable and causal filter (Oppenheim *et al.*, 1999).

Obtaining the stable correction filter when the transfer function $H(z)$ describes the non-minimum-phase system constitutes a bigger problem. In this case, determining directly the transfer function by (3.25) and assuming that it describes a causal system does not lead to the stable correction filter. Splitting the transfer function of the distorting system into two parts is a common approach applied in this case. Thus,

$$H(z) = H_{\min}(z) H_{ap}(z), \tag{3.27}$$

where $H_{\min}(z)$ describes the minimum-phase system; however, $H_{ap}(z)$ describes the all-pass system (Oppenheim *et al.*, 1999). The transfer function $G(z)$ of the correction filter is obtained according to (3.25), although on the basis of the transfer function $H_{\min}(z)$. It makes the correction filter causal and stable. As a result of the approach, the magnitude response is exactly compensated for, while the phase response is modified to the phase response of the all-pass system.

For the case when the distorting system is a non-minimum phase system, there are also approaches that use the blind deconvolution technique. The method presented in (Fiori and Maiolini, 2000) allows online deconvolution of signals distorted by non-minimum phase systems with neither knowledge of this system's impulse response nor distorted signal statistics, except for its moments up to the fourth order.

There are domains, e.g. medicine, where the compensation of signal phase distortions is a very important problem, because the useful information is not on the magnitude and phase of its Fourier transform but on its shape. Therefore, another possibility is to suppose that exact inversion of a distorting system describes the non-causal system. Processing the signal $x[n]$ adequately leads to full correction of distortions (Siwczyński and Kozioł, 2002).

None of them assures the stability of the compensating filter, when the transfer function $H(z)$ of a distorting system has the zeros on the unit circle. There are solutions proposed for the case, when the kernel has no spectral inverse, but they deconvolve a signal in either the frequency domain (Zazula and Gyergyek, 1993) or base on finite-length sequence (Tuncer, 1999), therefore online filtering of an infinite-length signal is unfeasible. The proposed solution allows obtaining stable but non-causal compensating filters. They can fully compensate the phase response of a distorting system and compensate for the magnitude response on an assumed level.

The solution of the above-mentioned problem can be carried out by a compromise, using a compensating filter that is stable and in series connection with the distorting system, which would form a system by all means similar to the identity system. It has been decided that searching for the solution will be treated as an optimization problem which will consider the following two criteria:

- stability of the compensating filter,
- approximation of the identity system by series interconnection of the distorting system and the compensating filter.

The minimization of the indices of these two criteria makes it possible to determine the optimal solution.

The mathematical form of this problem has been solved algebraically assuming that each real-valued signal is represented by the column vector in the multidimensional space, i.e. any signal $a(n)$ has an equivalent vector,

$$\boldsymbol{a} = \left[\ldots a_{-2}, a_{-1}, a_0, a_1, a_2 \ldots\right]^{\mathrm{T}}, \tag{3.28}$$

where a_n corresponds to the n-th sample of the signal $a(n)$. It was assumed that the indices of the vector coefficients can be negative because, in general, these vectors can represent non-causal signals. Therefore, the vectors must always have odd numbers of the coefficients to unambiguously determine the sample with the index 0. It can be always achieved by adding the zeros to a vector to position the sample with the index 0 into the middle of a vector.

The indices of the aforementioned criteria for the optimization procedure have been described mathematically by the inner product as below:

- the approximation index

$$f(\boldsymbol{g}) = \langle \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta}, \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta} \rangle = \sum_n \left|(h(n) * g(n)) - \delta(n)\right|^2, \tag{3.29}$$

  where $*$ represents the convolution, while $h(n)$, $g(n)$ and $\delta(n)$ represent the n-th sample of the impulse response of the distorting system, the correction filter and the identity system, respectively. $\boldsymbol{H}$ is the Toeplitz matrix, which is built out of the impulse response $h(n)$ of the distorting system (the columns of this matrix are successively delayed replicas of $h(n)$);

- the stability index

$$c(\boldsymbol{g}) = \langle \boldsymbol{g}, \boldsymbol{g} \rangle = \sum_n \left|g(n)\right|^2, \tag{3.30}$$

  where $g(n)$ represents n-th sample of the impulse response of the correction filter.

The first of the above-mentioned indices describes the level of the approximation of the identity system by series interconnection of the distorting system and the searched for compensation filter. Its value can vary between 0 and 1. The lower the value of the index, the better the approximation of the identity system. The second indicator describes the energy of the impulse response of the correction filter. If its value is finite, then the obtained filter will certainly be asymptotically stable. Additionally, the lower the value of the indicator, the shorter the impulse response of the correction filter and its poles lie further from the unit circle.

3.3.1. Optimization problems leading to quasi-inverse filters

In a situation when the discrete transfer function of a distorting system has the zeros on the unit circle, the search for stable inversion can be carried out by the minimization of the first of the presented indices with the assumed constant value of the other one. Therefore, the proposed procedure can be realized in the following two ways:

- A compensating filter which, in connection with the distorting system, forms the best approximation of the identity system is searched for, i.e.

$$f_1(\boldsymbol{g}) = \langle \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta}, \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta} \rangle \rightarrow \min . \tag{3.31}$$

  The second index forms the constraint

$$c_1(\boldsymbol{g}) = \langle \boldsymbol{g}, \boldsymbol{g} \rangle = q_1, \tag{3.32}$$

  which means that the stability index has to be equal to q_1.

- A compensating filter with the minimum value of the stability index is searched for, i.e.

$$c_2(\boldsymbol{g}) = \langle \boldsymbol{g}, \boldsymbol{g} \rangle \rightarrow \min . \tag{3.33}$$

  Additionally, the following constraint is formed:

$$f_2(\boldsymbol{g}) = \langle \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta}, \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta} \rangle = q_2, \tag{3.34}$$

  which means that the approximation index has to be equal to q_2.

The presented assumptions will be called the first and the second optimization problem, respectively. It seems that assuming the approximation index is more intuitive than in the case of the stability index. Nevertheless, both optimization problems have been solved to show that they lead to equivalent solutions.

3.3.2. Solutions of optimization problems

In order to group the requirements (3.31) and (3.32) in the first optimization problem, the Lagrange functional is determined:

$$L_1(\boldsymbol{g}, \lambda) = f_1(\boldsymbol{g}) + \lambda \left(c_1(\boldsymbol{g}) - q_1 \right) = \langle \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta}, \boldsymbol{H}\boldsymbol{g} - \boldsymbol{\delta} \rangle + \lambda \left(\langle \boldsymbol{g}, \boldsymbol{g} \rangle - q_1 \right), \tag{3.35}$$

where $\lambda \in \mathbb{R}^+$ is the Lagrange multiplier. For the established value of λ, the functional (3.35) can attain the minimum $\boldsymbol{g}_\lambda$ when for any variation $\boldsymbol{\Delta}_g$ of the vector $\boldsymbol{g}_\lambda$ the following inequality is always true:

$$L_1(\boldsymbol{g}_\lambda + \boldsymbol{\Delta}_g, \lambda) - L_1(\boldsymbol{g}_\lambda, \lambda) > 0. \tag{3.36}$$

It leads to the following solution (Kozioł, 2006):

$$\boldsymbol{g}_\lambda = \frac{\boldsymbol{H}^*}{\lambda \boldsymbol{1} + \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta}, \tag{3.37}$$

where $\boldsymbol{H}^*$ denotes generally the conjugate transpose of the matrix $\boldsymbol{H}$. This operation for linear time-invariant systems with real-valued samples of the impulse response is simply the transposition of the matrix $\boldsymbol{H}$. The dependence (3.37) defines the set of filters called the λ-family of quasi-inverse filters.

Identically, the following relationship has been obtained:

$$\boldsymbol{g}_\lambda = \frac{\lambda \boldsymbol{H}^*}{\mathbf{1} + \lambda \boldsymbol{H}^* \boldsymbol{H}}, \tag{3.38}$$

which represents the solution of the second optimization problem.

Taking (3.37), (3.38) and the definitions of the indices of the approximation and stability into consideration for each of the problems, the approximation $A(\lambda)$ and stability $S(\lambda)$ functions have been determined:

$$A_1(\lambda) = \left\langle \frac{\boldsymbol{H}^*}{\lambda \mathbf{1} + \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta} - \boldsymbol{\delta}, \frac{\boldsymbol{H}^*}{\lambda \mathbf{1} + \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta} - \boldsymbol{\delta} \right\rangle, \tag{3.39}$$

$$S_1(\lambda) = \left\langle \frac{\boldsymbol{H}^*}{\lambda \mathbf{1} + \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta}, \frac{\boldsymbol{H}^*}{\lambda \mathbf{1} + \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta} \right\rangle, \tag{3.40}$$

$$A_2(\lambda) = \left\langle \frac{\lambda \boldsymbol{H}^*}{\mathbf{1} + \lambda \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta} - \boldsymbol{\delta}, \frac{\lambda \boldsymbol{H}^*}{\mathbf{1} + \lambda \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta} - \boldsymbol{\delta} \right\rangle, \tag{3.41}$$

$$S_2(\lambda) = \left\langle \frac{\lambda \boldsymbol{H}^*}{\mathbf{1} + \lambda \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta}, \frac{\lambda \boldsymbol{H}^*}{\mathbf{1} + \lambda \boldsymbol{H}^* \boldsymbol{H}} \boldsymbol{\delta} \right\rangle. \tag{3.42}$$

According to the constraints (3.32) and (3.34), the stability index in the first optimization problem or the approximation index in the second optimization problem has to be equal to the assumed value. Therefore, in order to obtain the optimal solution for the established assumption, the value of λ has to be determined. In the first optimization problem, this is equivalent to solving the equation:

$$S_1(\lambda) = q_1. \tag{3.43}$$

In the second optimization problem, to determine the value of λ the following equation has to be solved:

$$A_2(\lambda) = q_2. \tag{3.44}$$

In both cases, the solution can be obtained using Newton's method (Siwczyński, 1995).

3.3.3. Transfer function of quasi-inverse filters

Using the properties of the Z transform, the solution (3.37) can be easily transformed to the transfer function $G_\lambda(z)$ of the quasi-inverse filter. For the first optimization problem, it has the following form:

$$G_\lambda(z) = \frac{H(z^{-1})}{\lambda + H(z^{-1})H(z)}. \tag{3.45}$$

In the same way, we can obtain the transfer function of the filter in the second optimization problem:

$$G_\lambda(z) = \frac{\lambda H(z^{-1})}{1 + \lambda H(z^{-1})H(z)} = \frac{H(z^{-1})}{\frac{1}{\lambda} + H(z^{-1})H(z)}. \tag{3.46}$$

The equations (3.45) and (3.46) are similar, thus all corollaries related to some variations of λ, which will be shown for the class of the filters described by (3.45), will be also true for the filters described by (3.46), although for the inverse variations of λ.

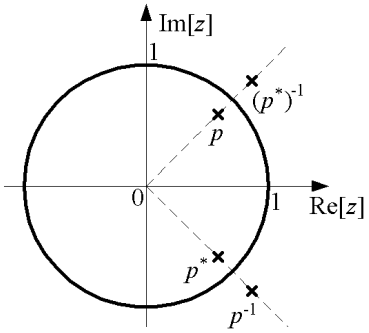


Fig. 3.4. Poles location pattern for quasi-inverse filters

It can be shown that the denominator of the transfer function $G_\lambda(z)$ is the symmetrical polynomial (Kozioł, 2006). It means that the complex poles appear by four, i.e. if p is the pole of the transfer function (3.45), then there exists the complex-conjugate pole p^*, and two poles p^{-1} and $(p^*)^{-1}$ conjugate reciprocally to the poles p^* and p, respectively. This is shown in Fig. 3.4. This pattern of pole locations for the quasi-inverse filters appears whether or not the transfer function has the rational or polynomial form.

Therefore, in order to obtain the BIBO-stable correction filter with a zero phase response, the quasi-inverse filter has to be deemed as a non-causal system which consists of two parts: causal and anticausal. The causal part $G^+\lambda(z)$ of the transfer function $G_\lambda(z)$ has to group all poles lying inside the unit circle, while the anticausal part $G_\lambda^-(z)$ – all poles lying outside the unit circle (Kozioł, 2006).

3.3.4. Frequency response of quasi-inverse filters

An advantage of the quasi-inverse filters is the independence of their phase response of the multiplier λ, and thus the established value of q as well. This can be shown by transforming (3.45), for example into the frequency domain, and determining the following equations for the magnitude and phase response:

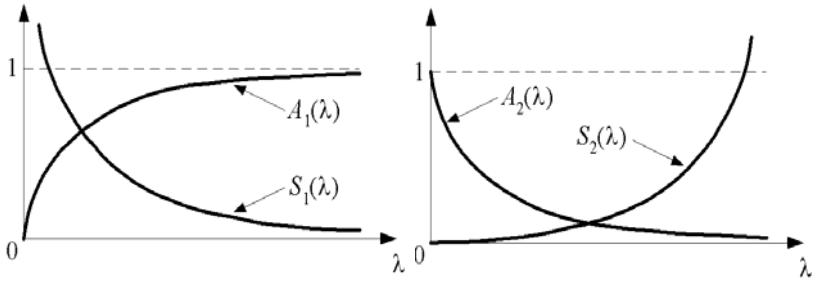
$$\left|G_\lambda\left(e^{j\omega}\right)\right| = \frac{\left|H\left(e^{j\omega}\right)\right|}{\lambda + \left|H\left(e^{j\omega}\right)\right|^2}, \tag{3.47}$$

$$\arg\left[G_\lambda\left(e^{j\omega}\right)\right] = -\arg\left[H\left(e^{j\omega}\right)\right]. \tag{3.48}$$

Additionally, it can be seen that the phase response is the exact contrary of the phase response of the distorting system, thus it provides exact correction of phase distortions. Of course, in practice, exact correction is achieved only when the transfer function $H(z)$ of the distorting system exactly describes the system.

3.3.5. Approximation and stability functions

Under the Parseval relation, the definitions (3.39)–(3.42) of the approximation and stability functions can be written in the Z transform domain. The knowledge of the transfer functions of the quasi-inverse filters allows us to analytically determine the plots of the approximation and stability functions for both optimization problems (Fig. 3.5) (Kozioł, 2006).



(a) First optimization problem (b) Second optimization problem

Fig. 3.5. Plots of the approximation $A(\lambda)$ and stability $S(\lambda)$ functions

It can be observed that for the specific optimization problem, the graphs of both functions have opposite trends, i.e. the better the approximation, the worse the stability, and vice versa. However, owing to the monotony of these functions, the equations (3.43) and (3.44) have always one and only one solution.

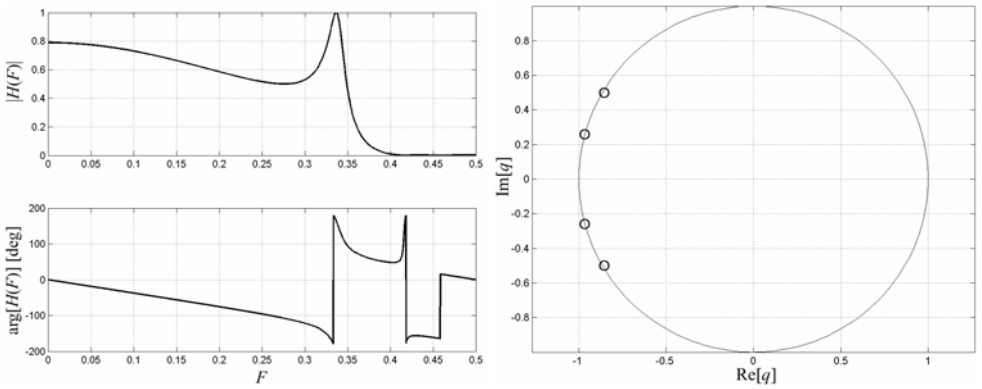
3.3.6. Signal processing by quasi-inverse filters

The assumed splitting of the quasi-inverse filter into the causal and anticasual parts requires proper processing of the signal by the anticasual part. According to its name, this part is the opposite of the causal system, so it processes the input signal in the other direction, i.e. in the decreasing direction of the sample indices. If the input signal has a finite number of samples, this limitation does not pose any problems. However, if the signal has an infinite number of samples, the realization of the process requires a proper attitude.

Block convolution can be used in order to realize real time filtering of the infinite input signal by the anticasual part. This procedure can be achieved by the overlap-add or overlap-save method. A very useful implementation of the overlap-add method in (Powell and Chau, 1991) is presented, where two LIFO stacks are used for time-reversed convolution.

3.3.7. Simulation example

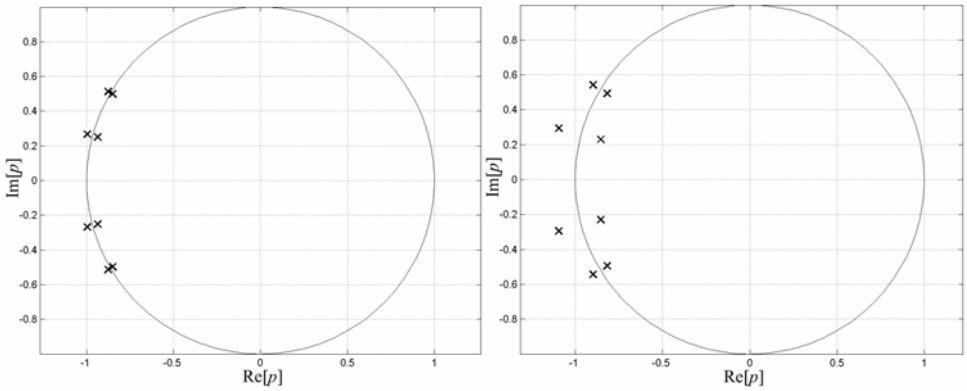
In the following example, the hypothetical distorting system has the polynomial transfer function. Its frequency response is shown in Fig. 3.6(a), where F is the normalized frequency.



(a) Frequency response (b) Zero plots

Fig. 3.6. Hypothetical distorting system

The zero locations (Fig. 3.6(b)) are matched so that a few of them lie on the unit circle or in its near proximity. It can be seen from Fig. 3.6(b) that the realization of the correction filter as the exact inversion of the distorting system does not lead to the BIBO-stable system. Therefore, in order to design the correction filter in this example, the solution for the second optimization problem was applied.

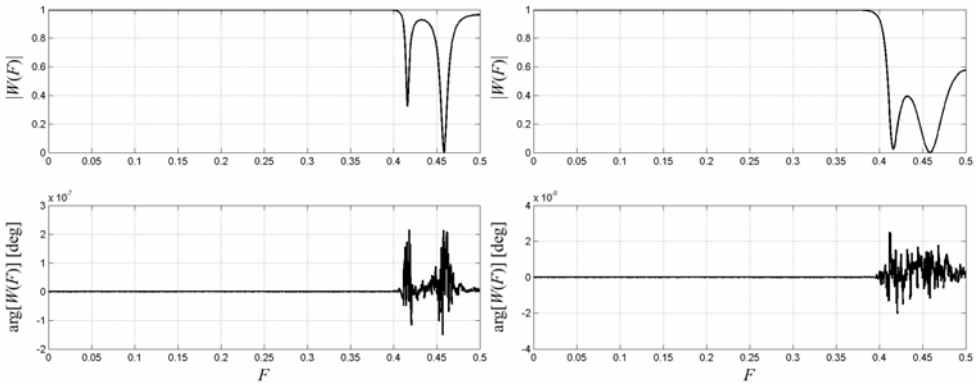


(a) $q_2 = 0.02$ (b) $q_2 = 0.1$

Fig. 3.7. Pole plots for the quasi-inverse filter for two values of q_2

The pole plots of the quasi-inverse filter for the two different values of q_2 in Fig. 3.7(a) and (b) are presented. It can be noticed that for a low value of q_2 some of the poles lie in low proximity to the unit circle. The increase the value of q_2 to 0.1

distinctly moves away the poles from the unit circle forming the stable correction filter. The frequency responses of the overall system obtained by applying the quasi-inverse filters, which have pole plots shown in Fig. 3.7(a) and (b), are presented in Fig. 3.8(a) and (b), respectively. As can be observed and has been mentioned before, the higher the value of the approximation index, the better the stability of the compensating system and the worse the approximation of the identity system.



(a) $q_2 = 0.02$ (b) $q_2 = 0.1$

Fig. 3.8. Frequency responses of the overall system for two values of q_2

3.4. Reconstruction of non-linear deformed periodic signals using the inverse circular parametric operators method

3.4.1. Non-linear system approximation by a sequence of linear time-varying systems

Some methods of the analysis of dynamic linear systems are well worked out, that is why there are some attempts at adapting them to analyze non-linear systems. A relatively new way is the approximation of a non-linear system by a sequence of Linear Time-Varying (LTV) systems. A non-linear system may be described in a state space notation:

$$\dot{x} = A(x)\, x + B(x)\, u, \tag{3.49}$$

where the matrices $A(x)$ and $B(x)$ are dependent on the state variable vector $\boldsymbol{x}$ and indirectly on time (Kaczorek *et al.*, 2005; Myszkowski, 2006; Tomas-Rodriques and Banks, 2003). The following sequence of linear time-varying approximations is introduced:

$$\dot{x}_k = A(x_{k-1})\, x_k + B(x_{k-1})\, u, \qquad k = 1, 2, \ldots \tag{3.50}$$

The initial element of the sequence (when $k = 0$) is a linear time invariant system:

$$\dot{x}_0 = A\left(x^0\right) x_0 + B\left(x^0\right) u, \tag{3.51}$$

with the initial condition $x_0(0) = x^0$. The sequence of solutions $x_k(t)$ of the linear time-varying systems (3.50) is convergent to the solution of the nonlinear equation (3.49). The proofs of local and global convergence are presented in (Tomas-Rodriques and Banks, 2003).

The solution of the non-linear state equation (3.49) may be obtained by analytically calculating some approximations $x_k(t)$, but the complexity of the solution is increasing with k (Kaczorek *et al.*, 2005; Myszkowski, 2006). It is better to use a discrete simulation.

The parameters of non-linear systems depend on the coercion signal. At the periodical steady state they change periodically and synchronously to the coercion. This similarity to LPTV systems produces the possibility of using a similar description. An LPTV system at a steady state may be described by using a Circular Parametric Operator (CPO). At the discrete time domain the CPO takes the form of a real coefficients constant matrix.

The simulation of the time invariant non-linear system using circular parametric operators requires the synchronization of the changes of parameters with the changes of signals. This synchronization for real non-linear systems is natural.

3.4.2. Description of an LPTV system using a circular parametric operator

The relationship between the input signal $x(t)$ and the output signal $y(t)$ of a time-varying system may be described with a differential equation with time-variable coefficients:

$$\sum_{i=0}^{q} a_i(t)\, y^{(i)}(t) = \sum_{i=0}^{q} b_i(t)\, x^{(i)}(t). \tag{3.52}$$

The equation (3.52) may be solved with the integral operator H:

$$y(t) = Hx(t) = \int_{-\infty}^{\infty} h(t,t')\, x(t')\, \mathrm{d}t'. \tag{3.53}$$

The operator kernel $h(t,\tau)$ is defined as a time-varying Dirac's pulse response. It is a function of two variables – it depends on the current time t and on the moment of pulse application τ.

Within the domain of discrete time, the relation of the input signal $x(n)$ to the output signal $y(n)$ for a time-varying system may be described with a difference equation of variable coefficients:

$$\sum_{i=0}^{q} A_i(n)\, y(n-i) = \sum_{i=0}^{q} B_i(n)\, x(n-i). \tag{3.54}$$

The equation (3.54) may be solved with the operator

$$y(n) = Hx(n) = \sum_{m=-\infty}^{\infty} h(n,m)\, x(m). \tag{3.55}$$

The operator kernel $h(n,k)$ is the pulse response. In the case of an LTV system, it is a function of two variables, i.e. it depends on the current time n, and on the moment of the pulse application k.

For the N-periodical input signal $x(n+N) = x(n)$ the response may be determined using the formula (Siwczyński, 1995):

$$y(n) = \tilde{H}x(n) = \sum_{m=0}^{N-1} \tilde{h}(n,m)\, x(m), \tag{3.56}$$

where

$$\tilde{h}(n,m) \equiv \sum_{p=-\infty}^{\infty} h(n, m-pN). \tag{3.57}$$

The N-periodic response of the N-periodically variable system to the N-periodic coercion may be determined with the use of the so-called circular parametric operator given in the form of the matrix (Siwczyński, 1995; 2003):

$$\begin{bmatrix} y_0 \\ y_1 \\ \dots \\ y_{N-1} \end{bmatrix} = \begin{bmatrix} \tilde{h}_{0,0} & \tilde{h}_{0,1} & \dots & \tilde{h}_{0,N-1} \\ \tilde{h}_{1,0} & \tilde{h}_{1,1} & \dots & \tilde{h}_{1,N-1} \\ \dots & \dots & & \dots \\ \tilde{h}_{N-1,0} & \tilde{h}_{N-1,1} & \dots & \tilde{h}_{N-1,N-1} \end{bmatrix} \begin{bmatrix} x_0 \\ x_1 \\ \dots \\ x_{N-1} \end{bmatrix}, \tag{3.58}$$

where $x_k = x(k)$, $y_k = y(k)$, $\tilde{h}_{n,m} = \tilde{h}(n,m)$. Or, shorter,

$$\boldsymbol{y} = \tilde{\boldsymbol{H}}\boldsymbol{x}, \tag{3.59}$$

where $\boldsymbol{x}, \boldsymbol{y}$ are the vectors of samples of one period of coercion and response signals, $\tilde{\boldsymbol{H}}$ is the circular parametric operator.

Circular parametric operators, designed for describing a periodic steady state, represent the phenomenon of mixing and generating the harmonics of input and output signals, typical for time-varying and non-linear systems.

3.4.3. Measurement-based determination of circular parametric operators for LPTV and non-linear systems

The operators describing real periodically time-varying systems may be determined on basis of the measured coercion and response signals. The base of identification is a matrix equation obtained from (3.59):

$$\tilde{\boldsymbol{H}}\boldsymbol{X} = \boldsymbol{Y}, \tag{3.60}$$

where $\tilde{\boldsymbol{H}}$ is $N \times N$ matrix – the sought circular parametric operator; $\boldsymbol{X}$, $\boldsymbol{Y}$ are $N \times K$ matrices of the K coercion and response signals. Each column is a vector of one signal samples.

In the case where $K = N$, i.e. the number of coercion and response signals K equals the size N of the square circular parametric matrix, the problem has an unequivocal solution:

$$\tilde{\boldsymbol{H}} = \boldsymbol{Y}\boldsymbol{X}^{-1}. \tag{3.61}$$

The solution depends on condition, which means linear independence of coercive signals:

$$\det \boldsymbol{X} \neq 0. \tag{3.62}$$

In the case where $K < N$, the matrix equation (3.60) has an infinite number of solutions. The optimal solution should be chosen. One can propose to seek an operator which describes a system of the smoothest changes of parameters. With the consideration of (3.60), the optimization problem may be originally defined as follows (Kłosiński, 2001):

$$(\Delta \boldsymbol{h}_n)^T \Delta \boldsymbol{h}_n \rightarrow \min, \tag{3.63}$$

$$\boldsymbol{X}^T \boldsymbol{h}_n = \boldsymbol{y}_n, \tag{3.64}$$

where $\boldsymbol{h}_n$ is the vector of elements of the n-th line of the matrix $\tilde{\boldsymbol{H}}$; $\boldsymbol{y}_n$ is the vector of n-th samples of all K response signals (elements of the n-th line of matrix $\boldsymbol{Y}$); $\Delta \boldsymbol{h}_n$ is the vector of increments of $\tilde{\boldsymbol{H}}$ elements for the n-th line, defined as follows ($(-)$ is the subtraction mark of modulo N):

$$\Delta h_{n,m} = h_{n,m} - h_{n(-)1,m(-)1}. \tag{3.65}$$

The criterion (3.63) means the minimization of increases in the coefficients of the matrix in the direction of the main diagonal. Such a criterion choice originates from the fact that in the case of Linear Time Invariant (LTI) system the relationship between signals of coercion and response, in a periodic steady state, is described by means of a cyclical matrix. Then, increases in elements defined by means of (3.65) equal zero. The choice criterion (3.63) means the research of the circular parametric operator describing the system of least variability of parameters, realizing (3.60). The equation (3.64) results from (3.60).

The optimization problem (3.63), (3.64) may be solved using the Lagrange method in the manner given by (Siwczyński, 2003). The solution is described in (Kłosiński, 2001; 2004; Siwczyński, 2003;). The result is the iterative solution

$$\boldsymbol{h}_n = \left(\boldsymbol{1} - \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{X}^T\right)\boldsymbol{P}\boldsymbol{h}_{n-1} + \boldsymbol{X}\left(\boldsymbol{X}^T\boldsymbol{X}\right)^{-1}\boldsymbol{y}_n, \tag{3.66}$$

where: $\boldsymbol{1} = \begin{bmatrix} 1 & 0 & \cdots & 0 \\ 0 & 1 & \cdots & 0 \\ \cdots & \cdots & & \cdots \\ 0 & 0 & \cdots & 1 \end{bmatrix}$ and $\boldsymbol{P} = \begin{bmatrix} 0 & \cdots & 0 & 1 \\ 1 & \cdots & 0 & 0 \\ \cdots & & \cdots & \cdots \\ 0 & \cdots & 1 & 0 \end{bmatrix}$ represent respectively a unit matrix and a circular delay matrix.

In order to effect the iterations (3.66), the invertibility of the matrix $(\boldsymbol{X}^T\boldsymbol{X})$ is necessary, thus the following condition must be fulfilled:

$$\det\left(\boldsymbol{X}^T\boldsymbol{X}\right) \neq 0, \tag{3.67}$$

which requires linear independence of the signals in the matrix $\boldsymbol{X}$.

The identification of the circular parametric operator consists in iteration determining consecutive lines of the matrix with reference to the previous lines, and with consideration of to the optimizing criterion and the periodicity of the identified system.

The obtained identification algorithm (3.66) has a standard form of a discrete state equation:

$$x\left(n+1\right)=\boldsymbol{A}x\left(n\right)+\boldsymbol{B}u\left(n\right), \tag{3.68}$$

where

$$\boldsymbol{A}=\left(\mathbf{1}-\boldsymbol{X}\left(\boldsymbol{X}^{T}\boldsymbol{X}\right)^{-1}\boldsymbol{X}^{T}\right)\boldsymbol{P}, \tag{3.69}$$

$$\boldsymbol{B}=\boldsymbol{X}\left(\boldsymbol{X}^{T}\boldsymbol{X}\right)^{-1}. \tag{3.70}$$

The convergence of the algorithm (3.66) depends on the eigenvalues of the matrix $\boldsymbol{A}$. The construction of $\boldsymbol{A}$ turns its eigenvalues to be

$$|\lambda_i|\leq 1, \qquad i=1,2,\ldots,N, \tag{3.71}$$

independently of the shape of signals included in the matrix $\boldsymbol{X}$. In the situation when $|\lambda_i|=1$, the solution of a homogeneous system is of a periodic signal form. That is why the required form of the initial vector is

$$x\left(0\right)=\left[0,0,\ldots,0\right]^{T}. \tag{3.72}$$

Depending on the input $u(n)$, the non-homogeneous system (3.68) may reach infinite amplitude. The eigenvalues of the matrix $\boldsymbol{A}$ (3.69), of the absolute value $|\lambda_i|=1$ of the form

$$\lambda_{k,N-k}=e^{\pm j\frac{2\pi}{N}k}, \tag{3.73}$$

will appear if any of the periodic signals included in the matrix $\boldsymbol{X}$ does not consist of the k-th harmonic. On the other hand, the vector

$$v\left(n\right)=\boldsymbol{B}u\left(n\right)=\boldsymbol{X}\left(\boldsymbol{X}^{T}\boldsymbol{X}\right)^{-1}u\left(n\right), \tag{3.74}$$

which is the addend of $x\left(n+1\right)$, is a linear combination of the signals included in $\boldsymbol{X}$ and does not include the k-th harmonic corresponding to the eigenvalues of the (3.73) form. This means that the identification algorithm (3.66) is not divergent and is convergent to an N-periodical solution.

The identification algorithm works properly. However, its usage may bring expected results only if the identification data, i.e. excitation and response signals, include the necessary portion of information about the performance of the system being identified. To obtain full identification, the set of N linear independent coercion signals (and N related response signals) is required.

As the behavior of a non-linear system depends on the coercion shape and amplitude, its description (in opposition to LPTV systems) requires an infinite number of linear operators. Different coercion causes different non-linear system behavior and implies different circular parametric operator. In practice, it is possible to simulate approximately a periodic steady state of a non-linear system using a finite set of circular parametric operators. The approach is similar in the case of curve approximation, where a finite set of straight line segments is used.

An LPTV system may be described by one circular parametric operator, which may be determined on the basis of a set of varied input and output signals. To describe a non-linear system, a different operator for different coercion is required.

The assumption that the low amplitude disturbance added to the dominant coercion signal (basic coercion) insignificantly changes the behavior of the non-linear system facilitates the determination of the approximate circular parametric operator assigned to this basic coercion.

3.4.4. Idea of the reconstruction of the non-linear deformed periodic signal method

A steady state of a time invariant non-linear system is taken into consideration. Input and output signals are given in the form of one period samples. It is assumed that the signals period is an integer multiple of the sampling period, and the sampling frequency is high enough to avoid the aliasing phenomenon.

The reconstruction consists of two stages. First, the distorting non-linear system has to be identified. It is necessary to specify a set of the basic coercion signals. One circular parametric operator is determined for each basic coercion signal. The identification algorithm (3.66) based on the measurements of coercion and response signals is used. To obtain an inverse operator, it is necessary to exchange the set of coercion signals for the set of response signals, and vice versa. The result of system identification is a set of circular parametric operators with the corresponding basic response signals. The basic response signal is the output signal obtained when the basic coercion signal is an input.

The second stage of the reconstruction consists in calculating the input signal on the basis of the measured output signal. First, the suitable CPO from the set must be chosen. The way of selection is the greatest similarity in some criterion of the output signal to the basic response signal. The reconstructed signal is determined by the operation of the selected CPO on the measured output signal; it is simply matrix multiplication.

3.4.5. Experiments

The presented method has been applied to the reconstruction of the primary current signal of a Current Transformer (CT). The diagram of the circuit is shown in Fig. 3.9. The exact parameters of the shunt resistors used were not known, so that the resistors were recognized as parts of the deformation system. The voltage u_x was the input signal and the voltage u_y was the output signal. The measurements were realized by the use of DaqBoard equipped with a 16 bits a/d converter. Signals were represented by vectors of 64 samples of one period.

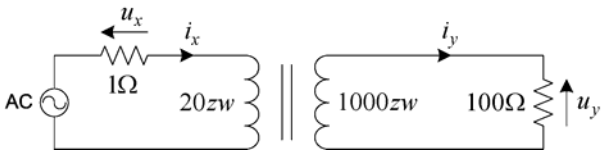


Fig. 3.9. Diagram of the tested circuit

Experiment 3.1. It was assumed that the basic harmonic of the reconstructed signal was dominant. The selection of the suitable circular parametric operator depended

on the Amplitude of Basic Harmonic (ABH). The determination of inverse circular parametric operators was based on the basic frequency sinusoidal signals of various amplitudes used as the basic coercion signals. The acceptable range of the value of the ABH was partitioned into seven ranges. The deforming system was represented by seven circular parametric operators determined for the input ABH: 1.0, 1.5, 2.0, 2.2, 2.4, 2.5 and 2.6 volts. For the used CT, the regular range of primary current is 1 ampere (1.41 V value of u_x ABH). Each circular parametric operator was determined by using the identification algorithm based on the set of 63 measured coercion signals and the set of 63 corresponding response signals. All coercion signals included a basic frequency sinusoidal signal of the exact ABH and one additional harmonic of a number from 2 to 32, each of phase 0 and 90 degrees, and one sinusoidal signal without any additional harmonic. An example of the obtained operator is presented in Fig. 3.10.

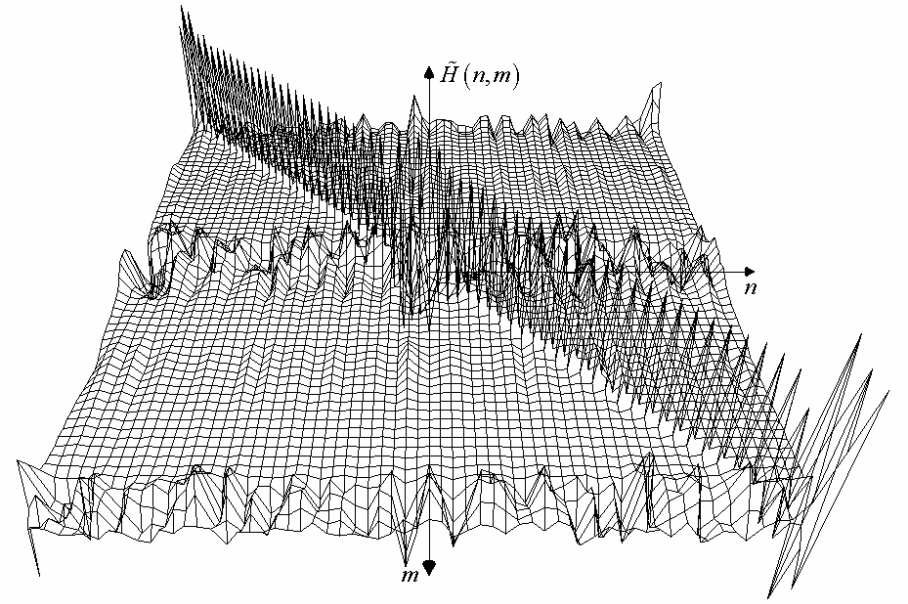


Fig. 3.10. Example of the obtained circular parametric operator

The synchronization of the changes of parameters with the changes of signals was realized by shifting in time the identification data signals and the corrected signals so that the phase of the base harmonic was nearest to the zero value. Some poly-harmonic periodic signals obtained from numerical synthesis were the test signals of reconstruction experiments. In the beginning of signal reconstruction, the amplitude and phase of the basic harmonic were determined. Then the suitable circular parametric operator was chosen and the reconstructed signal was determined. Each time the reconstructed signal and the measured output signal u_y scaled-down to the input level were compared with the original input signal u_x. The experiments results are presented in Figs. 3.11 and 3.12. For the lower amplitude of the input, situated in the range of linear operation of the CT, the reconstructed signals have a higher error than the scaled-down output signals. For the higher amplitude of the input, when the CT

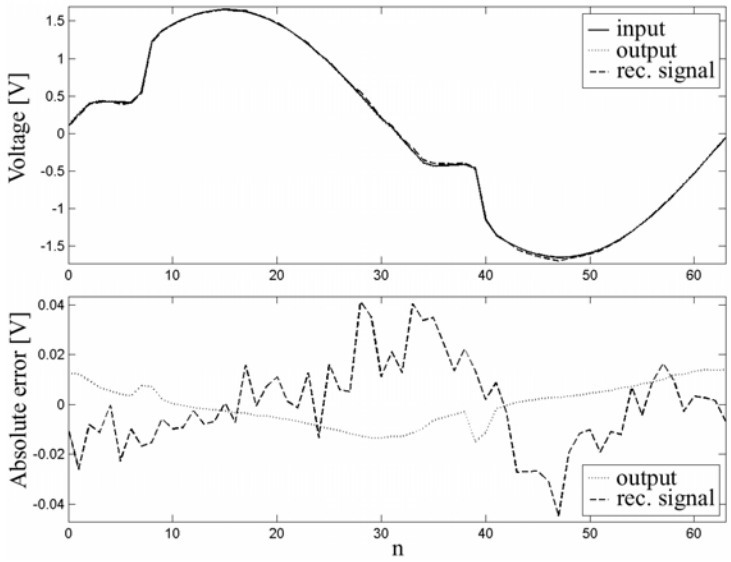


Fig. 3.11. Results of input signal reconstruction and absolute error characteristics obtained for a lower fundamental harmonic amplitude. Notation: input – input signal u_x, output – scaled-down output signal u_y, rec. signal – signal obtained from reconstruction

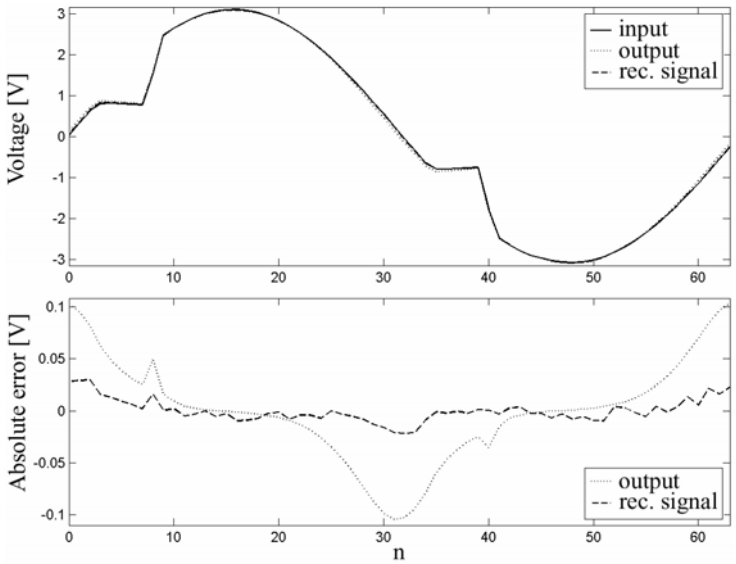


Fig. 3.12. Results of input signal reconstruction and absolute error characteristics obtained for a higher fundamental harmonic amplitude. Notation: input – input signal u_x, output – scaled-down output signal u_y, rec. signal – signal obtained from reconstruction

becomes non-linear, the reconstructed signals have a lower error than the scaled-down output signals.

Experiment 3.2. Signals of the shape similar to the reconstructed signals were the basis for identification this time. Pairs of coercion and response signals of the shape of the cut sinusoid of different amplitude and different cutting phase were grouped according to the maximum of the absolute value. Eight sets of twenty signals were prepared, thus the deforming system was represented by eight circular parametric operators. The criterion of selecting the CPO for reconstruction was the worth of the maximum of the absolute value of the measured output signal. The experiments results are presented in Fig. 3.13. In a situation when a non-sinusoidal signal is reconstructed using similar shape signals for identification brings better results.

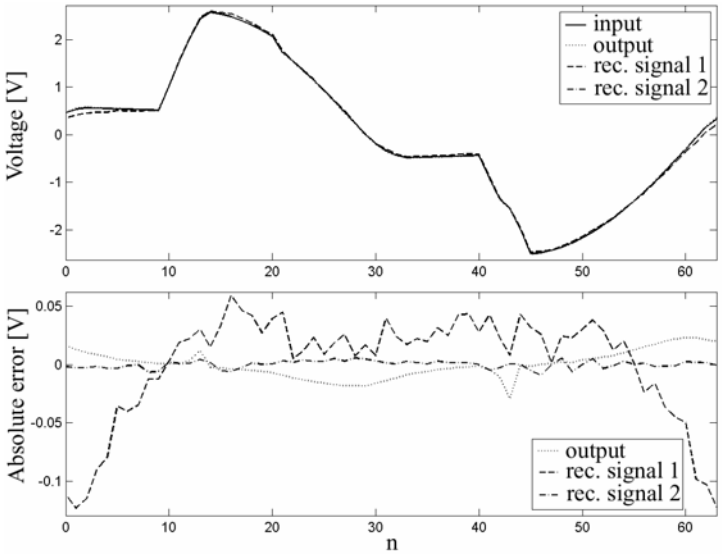


Fig. 3.13. Results of input signal reconstruction and absolute error characteristics obtained for a non-sinusoidal signal higher fundamental harmonic amplitude. Notation: input – input signal u_x, output – scaled-down output signal u_y, rec. signal 1 – signal obtained from reconstruction in the manner applied in Experiment 1, rec. signal 2 – signal obtained from reconstruction in the manner applied in Experiment 2

3.5. Conclusions

In this chapter selected compensation methods of conditioning system imperfections were presented. Linear and non-linear models were used for signal reconstruction. The linear model can be used in the case of low non-linearity of the compensated system. It allows one to obtain high correction accuracy. If the distorting system is strongly non-linear, the non-linear model is required. The description of nonlinear systems is very complex, but the proposed method, based on the simplified model, has given good results of signals reconstruction.

In order to design the correction system it is necessary to take into account not only correction accuracy but also the correction system stability. Providing stability is particularly difficult if the zeros of the discrete transfer function of the compensated system lie on the unit circle or its near proximity. The solution of this problem by the application of the so-called quasi-inverse filters has also been discussed here.

References

Dabóczi T. and Bakó T.B. (2001): *Inverse filtering of optical images.* — IEEE Trans. Instrumentation and Measurement, Vol. 50, No. 4, pp. 991–994.

Fiori S. and Maiolini G. (2000): *Weighted least-squares blind deconvolution of non-minimum phase systems.* — IEE Proc. Vis. Image Signal Process., Vol. 147, No. 3, pp. 557–563.

Furmankiewicz L. (1998): *Possibility of Software Correction of Errors Introduced by Transformer Type Input Circuit of Power Transducers in Distorted Signals Measurements.* — Ph.D. dissertation, Technical University of Zielona Góra Press, (in Polish).

Furmankiewicz L. (1999): *Using software as a method of correcting errors in power transducers.* — Electrical Power Quality and Utilization, Vol. 1, No. 1, pp. 31–35, (in Polish).

Furmankiewicz L. and Rybski R. (2003): *Multiparameters instruments for power systems measurement.* — Pomiary, Automatyka, Kontrola, PAK 2/3/2003, pp. 41–46, (in Polish).

Furmankiewicz L. (2005): *Algorithm errors in power quality analyser AJE1 on magnitude spectrum measurements.* — Zeszyty Naukowe Politechniki Śląskiej, No. 1670, Elektryka, No. 195, pp. 41–49, (in Polish).

Jakubiec J. (2002): *Application of Reductive Interval Arithmetic to Uncertainty Evaluation of Measurement Data Processing Algorithms.* — Gliwice: Politechnika Śląska Press.

Kaczorek T., Jordan A. and Myszkowski P. (2005): *The approximation of nonlinear systems by the use of linear time varying systems.* — Proc. XXVIII Int. Conf. *Fundamentals of Electrotechnics and Circuit Theory*, Gliwice-Ustroń, Poland, Vol. 2, pp. 321–323, (in Polish).

Myszkowski M. (2006): *The approximation of the non-linear model of electrical circuit with diode by the use of linear time-varying systems.* — Proc. XI Conf. *Computer Application in Electrical Engineering*, Poznań, Poland, pp. 93–94, (in Polish).

Kłosiński R. (2001): *The algorithm of cycloparametric operator identification and its application in optimal control of compensation circuits.* — Proc. IEEE Workshop *Signal Processing*, Poznań, Poland, pp. 115–120.

Kłosiński R. (2004): *Designing of digital parametric filters with the use of the identification algorithm for cycloparametric operator.* — Computer Applications in Electrical Engineering, Part 2, Poznań University of Technology, pp. 347–363.

Koszmider A., Olak J. and Piotrowski Z. (1985): *Current Transformers.* — Warszawa: Wydawnictwo Naukowo-Techniczne, WNT, (in Polish).

Koziol M. (2006): *Synthesis of digital quasi-inverse filters in the class of the noncausal systems.* — Ph.D. dissertation, Faculty of Electrical Engineering, Computer Science and Telecommunications, University of Zielona Góra Press, (in Polish).

Merino C., Luis-Garcia M.L., Hernandez S.E., Martin F.A., Casanova O., Gomez D., Castellano M.A. and Gonzales-Mora J.L. (2005): *Application of digital deconvolution technique to brain temperature measurement and its corellation with other physiological parameters.* — Proc. 18-th IEEE Symp. *Computer-Based Medical Systems*, Trinity College Dublin, Ireland, pp. 47–52.

Miekinia A., Morawski R.Z. and Barwicz A. (1997): *The use of deconvolution and iterative optimization for spectrogram interpretation.* — IEEE Trans. Instrumentation and Measurement, Vol. 46, No. 4, pp. 1049–1053.

Oppenheim A.V., Schafer R.W. and Buck J.R. (1999): *Discrete-Time Signal Processing.* — Prentice-Hall.

Powell S.R. and Chau P.M. (1991): *A technique for Realizing Linear Phase IIR Filter.* — IEEE Trans. Signal Processing, Vol. 39 No. 11, pp. 2425–2435.

Siwczyński M. (1995): *Optimization Methods in Power Theory of Electrical Circuits.* — Wydawnictwo Politechniki Krakowskiej, (in Polish).

Siwczyński M. and Kozioł M. (2002): *Synthesis of Optimised Digital Filters Used to Correction in Measurement Systems.* — Proc. IMEKO-TC7 Symp. *Measurement Science of the Information Era*, Cracow, Poland, pp. 150–156.

Siwczyński M. (2003): *Power Engineering Circuit Theory.* — Monograph, Wydawnictwo Instytutu Gospodarki Surowcami Mineralnymi i Energią Polskiej Akademii Nauk, IGSMiE PAN, Cracow, (in Polish)

Sydenham P. H. (Ed.)(1983): *Handbook of Measurement Science.* — Chichester: Wiley and Sons Ltd., Vol. I, II.

Szczecinski L. and Barwicz A. (1997): *Quickly converging iterative algorithm for measurand reconstruction.* — Measurement, Vol. 20, No. 3, pp. 211–217.

Tomas-Rodriquez M. and Banks S. P. (2003): *Linear approximations to nonlinear dynamical systems with applications to stability and spectral theory.* — IMAJ Math. Control Inform., Vol. 20, pp. 81–103.

Tuncer T.E. (1999): *A new method for D-dimensional exact deconvolution.* — IEEE Trans. Signal Processing, Vol. 47, No. 5, pp. 1324–1334.

Zazula D. and Gyergyek L. (1993): *Direct Frequency-Domain Deconvolution when the Signals Have No Spectral Inverse.* — IEEE Trans. Signal Processing, Vol. 41, No. 2, pp. 977–981.

Chapter 4

VOLTAGE AND CURRENT CALIBRATORS

Andrzej OLENCKI*, Jan SZMYTKIEWICZ*, Krzysztof URBAŃSKI*

4.1. Introduction

The first idea of the voltage calibrator was presented in the IEC document (IEC 443, 1974), where the calibrator was named a *stabilized supply apparatus for measurement*, whose scheme is presented in Fig. 4.1. This scheme explains the construction and

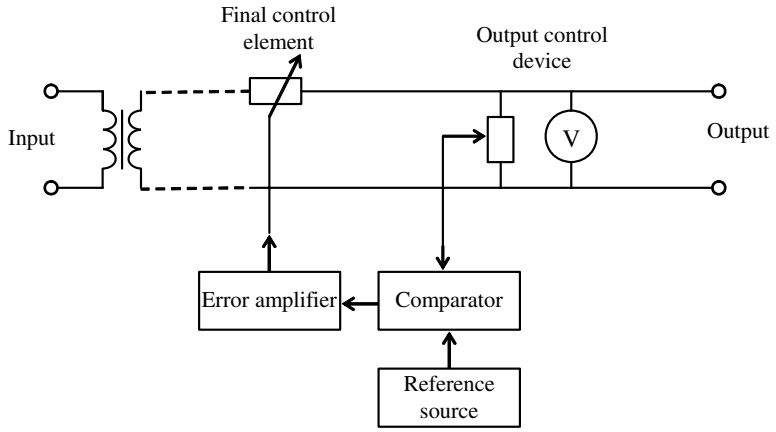


Fig. 4.1. Functional component diagram of a supply apparatus with a stabilized voltage by closed loop stabilization (scheme of the voltage calibrator)

"energetic" definition of the calibrator – the calibrator is an apparatus which takes electrical energy from a supply source and supplies the electrical energy, in a modified form, to one or more loads and in which one or more of the output quantities are stabilized. The input of the calibrator is connected to a power network.

* Institute of Computer Engineering and Electronics
e-mails: {A.Olencki, J.Szmytkiewicz, K.Urbanski}@iie.uz.zgora.pl

The calibrator consists of the following items:

- the final control element, which controls the output voltage to a specified value,
- the reference source – a source of voltage to the value of which the output in closed loop stabilization is referred,
- the comparator, which compares the value of the output voltage with a reference voltage and produces a difference signal,
- the error amplifier, which amplifies a difference signal (error signal),
- the output control device, which measures the output voltage.

This chapter presents the main ideas and theoretical problems which were solved by the authors to implement multifunction calibrators, three-phase power calibrators and industrial signals calibrators.

4.2. Static model of the voltage calibrator

4.2.1. Definitions of the calibrator

The Polish DC and AC voltage calibrator definitions were published in the instructions issued by the Polish Central Office of Measures at the end of 1970 (DzNiM, 1978). The voltage calibrator is an electronic control voltage source which has possibility to obtain the output voltage with a specified value and accuracy without the necessity of carrying out measurements and manual corrections. This "information" definition of the calibrator is up-to-date and may be expanded for electrical values calibrators such as current calibrators, phase angle calibrators, power and energy calibrators, and even resistance, capacity and inductance calibrators.

In the next calibrator definition (DzNiM, 1984) – voltage, current, power and resistance calibrators are devices which have the possibility to obtain the output quantities value according to digital setting without the necessity of performing measurements and corrections. A general and simple functional scheme of the calibrator is shown in Fig. 4.2 and consists of the following blocks:

- the Digital to Analogue Converter, DAC, which converts the input quantity value X (input setting) in a digital form to the output quantity value Y (output quantity) in an analogue form,
- the readout device for the indication of the input setting X or the output quantity value Y,
- the supply source, which, according to the "energetic" definition, takes electrical energy from a power network or battery and supplies it to the DAC and the readout device.

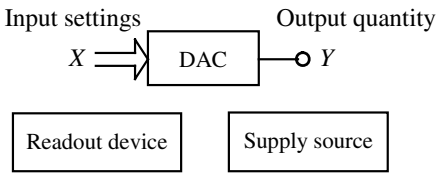


Fig. 4.2. Simple functional scheme of the calibrator according to the definition

4.2.2. Model of the multifunction (DC and AC voltage and current) calibrator

A block scheme of the voltage and current calibrator model (Olencki, 1991; Olencki and Szmytkiewicz, 1999; Szmytkiewicz, 2000) is shown in Fig. 4.3. It consists of digital and analogue parts. The forward branch, which transfers the input setting X to the output quantity Y, consists of two converters: the digital to digital converter (control system) and the multi range digital to analogue converter (digitally programmed voltage and current source).

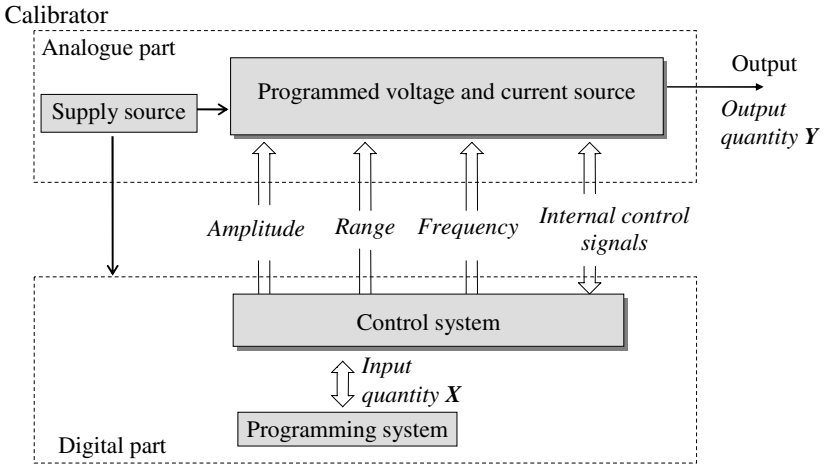


Fig. 4.3. Block scheme of the calibrator model

The transfer function, which describes mathematically the relationship between output and input signals of both converters, is linear, therefore the nominal transfer function $Y_N = f(X)$ of the calibrator model is (Olencki and Szmytkiewicz, 1999):

$$Y_N = X. \tag{4.1}$$

The real transfer function $Y_R = f(X)$ of the calibrator is

$$Y_R = (\delta_M Y + 1)X + \Delta_A Y, \tag{4.2}$$

where $\Delta_A Y$ is the additive part of the error, and $\delta_M Y$ is the multiplicative part of the error.

The error equation of the calibrator is the subtraction of the real and nominal transfer functions:

$$\delta Y = \delta_M Y + (\Delta_A Y / X), \tag{4.3}$$

and describes static features of the calibrator.

4.2.3. Open structure of the calibrator

The typical open structure of the DC and AC voltage and current calibrator is illustrated in a block diagram form in Fig. 4.4 and consists of the digital to analogue converter DAC and the Output System OS. The output system can be

- a voltage amplifier in DC voltage calibrators,
- a transconductance amplifier in DC current calibrators,
- a DC to AC converter plus an AC voltage amplifier in AC voltage calibrators,
- a DC to an AC converter plus AC transconductance amplifier in AC current calibrators.

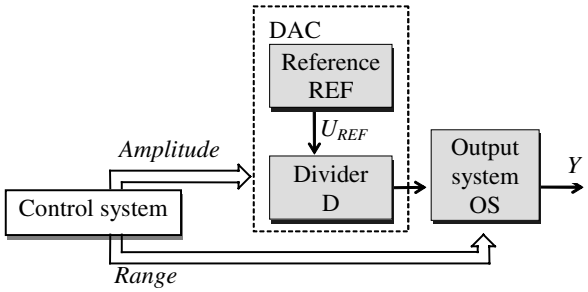


Fig. 4.4. General block diagram for an open structure calibrator with a DAC

The nominal transfer function and error equation of the open structure are (Olencki and Szmytkiewicz, 1999):

$$Y = U_{\mathrm{REF}}\, K_{\mathrm{D}}\, K_{\mathrm{OS}}, \tag{4.4}$$

$$\delta Y = \delta U_{\mathrm{REF}} + \delta K_{\mathrm{D}} + \delta K_{\mathrm{OS}} + \frac{\Delta U_{\mathrm{D}} + \Delta U_{\mathrm{OS}}}{U_{\mathrm{REF}}\, K_{\mathrm{D}}}, \tag{4.5}$$

where δU_{REF} is the error of the reference voltage U_{REF}, δK_{D} is the multiplicative error of the divider D, ΔU_{D} is the additive error of the divider D referenced to its output, δK_{OS} is the multiplicative error of the output system, and ΔU_{OS} is additive error of the output system OS referenced to its input.

The open structure is very simple but the error equation contains errors of all blocks of the structure, particularly output system errors: δK_{OS} and ΔU_{OS}.

4.2.4. Closed loop structure of the calibrator and error analysis

The closed loop structure of the AC calibrator is illustrated in Fig. 4.5 and consists of the DAC and the output system OS, as presented in Fig. 4.3. The output system OS uses a single-loop control system with a comparison of DC signals. The DAC's output voltage, proportional to the amplitude setting, is compared with a DC voltage proportional to the output quantity Y from a feedback branch. The feedback consists of the sense system SS and the AC to DC converter (in the AC calibrator). The sense system measures the output voltage or current Y, by means of precision voltage dividers or current shunts. Any difference between the two comparator inputs is amplified by the controller CTR, to produce a controlling signal for driving the output converter DC to AC (DC/DC in the DC calibrator) and the power amplifier PA. The output of the DC/AC converter consists of a generator G, which generates a fixed, low-distortion sine wave, and a modulator M, which is a voltage controlled divider.

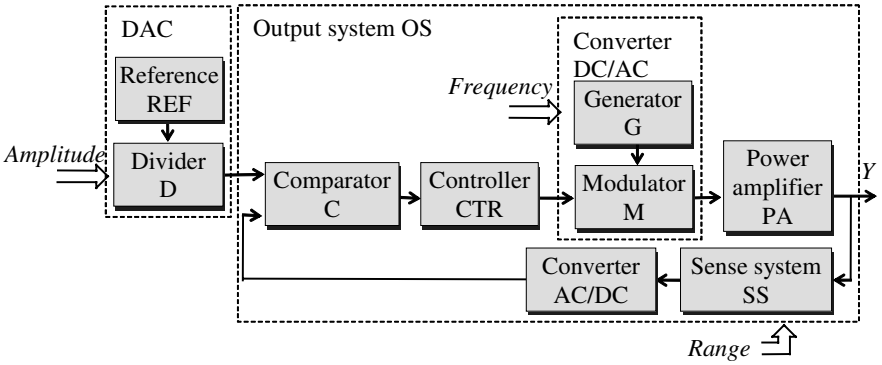


Fig. 4.5. AC calibrator's block diagram for a closed loop structure

In proportional plus integral feedback control systems with astatic control characteristics, the controller CTR consists of an integrator. The error equation of this system is

$$\delta Y = \delta U_{\mathrm{REF}} + \delta K_{\mathrm{D}} - \delta K_{\mathrm{SS}} - \delta K_{\mathrm{ACDC}} + \frac{\Delta U_{\mathrm{D}} + \Delta U_{\mathrm{C}} - \Delta U_{\mathrm{SS}}\, K_{\mathrm{ACDC}} - \Delta U_{\mathrm{ACDC}} + \Delta U_{\mathrm{CTR}}/K_{\mathrm{C}}}{U_{\mathrm{REF}}\, K_{\mathrm{D}}}, \tag{4.6}$$

where δU_{REF}, δK_{D}, ΔU_{D} are multiplicative errors of the voltage reference and divider and additive error of the divider, which represents the accuracy of the DAC, ΔU_{C} is the additive error of the comparator C with reference to its input, δK_{UN}, ΔU_{UN} are multiplicative and additive errors of the sense system SS with reference to its output, δK_{ACDC}, ΔU_{ACDC} are multiplicative and additive errors of the AC to DC converter, ΔU_{UC} is the additive error of the controller (integrator), and K_{C} is the amplitude gain of the comparator C.

The relation (4.6) shows that errors of blocks between the controller and the calibrator output are absent. This is the property of proportional plus integral feedback control systems with astatic control characteristics.

4.3. Dynamic properties of calibrators using the closed loop structure

The AC voltage calibrator may be presented as an automatic control system illustrated ln Fig. 4.6 (Olencki and Szmytkiewicz, 1999), where U_{DAC} is the DAC's output voltage, T_{CTR} is the controller's time constant, T_{ACDC} is the AC/DC converter's time constant.

The best output transient is when

$$\frac{K_{\mathrm{C}}\, K_{\mathrm{DCAC}}\, K_{\mathrm{SS}}\, K_{\mathrm{ACDC}}}{T_{\mathrm{CTR}}} = \frac{1}{4\, T_{\mathrm{ACDC}}}. \tag{4.7}$$

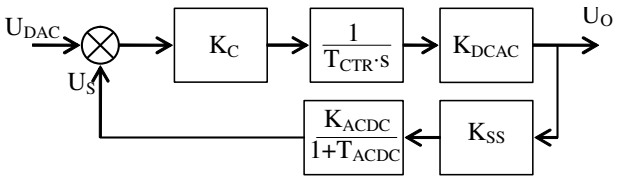


Fig. 4.6. AC voltage calibrator as an automatic control system

Settling time (transient recovery time) as an answer to a step change of the setting is computed from the AC calibrator dynamic equation:

$$t_{\mathrm{O}} \geq \frac{K_{\mathrm{C}}}{f\ \mathrm{THD}}, \qquad \text{for } K_{\mathrm{C}} \gg 1, \tag{4.8}$$

$$t_{\mathrm{O}} \geq \frac{1}{2f\sqrt{\mathrm{THD}}} \ln \frac{4}{\delta}, \quad \text{for } K_{\mathrm{C}} \approx 1, \tag{4.9}$$

where f is the frequency of the output signal, THD is the total harmonic distortion of the output signal, and δ represents error limits placed around rated output value.

From the following DC calibrator dynamic equation the settling time of the DC calibrator using a DC/AC/DC converter in the output system of the calibrator can be calculated:

$$t_{\mathrm{O}} \geq \frac{2K_{\mathrm{C}}}{f_{\mathrm{G}}\mathrm{PARD}}, \qquad \text{for } K_{\mathrm{C}} \gg 1, \tag{4.10}$$

$$t_{\mathrm{O}} \geq \frac{1}{f_{\mathrm{G}}\mathrm{PARD}} \ln \frac{4}{\delta}, \quad \text{for } K_{\mathrm{C}} \approx 1, \tag{4.11}$$

where f_{G} is the converting frequency of the DC/AC/DC converter, and PARD is the periodic and random deviation of an output quantity from its average value.

The equations of dynamic properties (4.8)–(4.11) (Olencki and Szmytkiewicz, 1999) describe limitations and settling time short cut possibilities for calibrators made by means of one closed loop, closed tracking structure with PI control and output stage designed by means of a DC/AC converter for an AC calibrator and a DC/AC/DC converter for a DC calibrator.

4.4. Digital to analogue converters used in calibrators

4.4.1. Basic requirements

Errors of the DAC's reference and divider are included in the error equations (4.5) and (4.6) of the calibrators, which work on the basis of open and closed loop structures. DACs used in multifunction calibrators have high resolution, from 18 to 24 bits, and are built by means of applying the precision DC voltage reference source and of the digitally programmed precision divider.

In the first calibrators there were applied D/A converters with a resistive divider. Today, those converters are used as integrated circuits in calibrators with low and

medium resolution of settings up to 16 bits e.g. in industry signals calibrators, or with low and medium accuracy up to 0,01%, e.g. in three phase power calibrators.

4.4.2. PWM DACs

Most calibrators use D/A converters based on Pulse-Width Modulation, PWM, (Fluke, 1979; Grimbleby, 2004) idea illustrated in Fig. 4.7. The DAC consists of digital and analogue sections. The digital section uses a PWM converter to generate a digital waveform with a mark/space ratio (duty cycle τ/T) proportional to the input setting S_{IN}. This waveform is converted to an analogue signal by a fast switch with precisely known resistance, and law-pass filtered by a τ/T to DC converter.

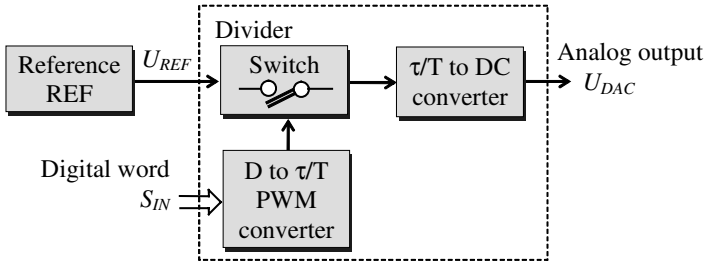


Fig. 4.7. Block diagram of the PWM DAC

The average analogue output voltage U_{DAC} is

$$U_{\mathrm{DAC}} = U_{\mathrm{REF}} \frac{\mathrm{SIN}}{2^n}, \tag{4.12}$$

where S_{IN} is a n-bit binary word. The frequency of the PWM waveform is

$$f_{\mathrm{PWM}} = f_{\mathrm{C}}/2^n, \tag{4.13}$$

where f_{C} is the frequency of a clock. Components at the f_{PWM} frequency must be removed by low-pass filtering. Filter cutt-off frequency must be much less than PWM frequency the f_{PWM} to achieve sufficient rejection of PWM component. The largest AC component occurs, when the input setting S_{IN} is 2^{n-1} and the amplitude of the fundamental component is (Grimbleby, 2004):

$$A + 1 = \frac{1}{\Pi} \int_0^{\Pi} U_{\mathrm{REF}} \sin t \, \mathrm{d}t = \frac{2U_{\mathrm{REF}}}{\Pi}. \tag{4.14}$$

PWM DACs are simple to implement and have very good linearity. The relation (4.13) shows that the increasing resolution will decrease the PWM frequency. Good solution is a PWM weighted-resistor DAC, which uses few PWM DACs with low resolution and sums the PWM waveforms. PWM weighted-resistor DACs have very good linearity and can handle higher f_{PWM} frequency than PWM DACs.

By applying the three PWM DACs with 7-bit precision it is possible to achieve a PWM weighted-resistor DAC with 21-bit precision and 16 384 times higher PWM

frequency. Integral linearity errors (maximum difference between the actual analogue voltage and the straight line between endpoints) of PWM weighted-resistor DACs are presented in Fig. 4.8 (Szmytkiewicz, 2000) and were obtained in the C101 multifunction calibrator (Olencki, 1998).

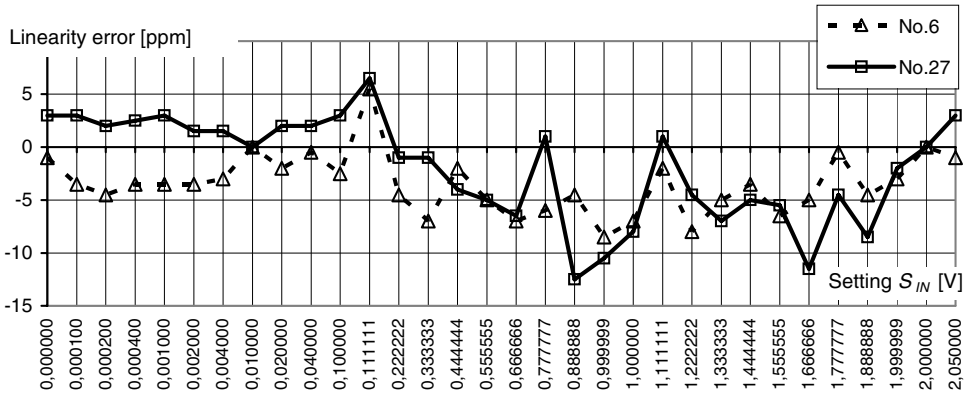


Fig. 4.8. Measurement results of the DAC's integral linearity errors

4.4.3. DACs with inductive voltage dividers

In the first Polish multifunction calibrators: model GA1 (Olencki, 1982) and the model SQ10 (Szmytkiewicz, 1990), there were applied D/A converters with a 20-bit precision inductive voltage divider (Olencki, 1983; 1984), in which there were achieved 10 ppm of Full Scale (FS) integral linearity and errors at the same level, too. The DAC's divider (see Fig. 4.9) consists of a DC/AC converter, an inductive voltage divider and an AC/DC converter. The DC/AC converter is used to generate a trapezoidal bipolar waveform with a precision peak to peak value (Lange and Olencki, 1986a). The amplitude of this waveform is divided with the use of the precision inductive divider. The peak to peak value of an output divider voltage is converted do the proportional DC output voltage U_{DAC} (Lange and Olencki, 1986b).

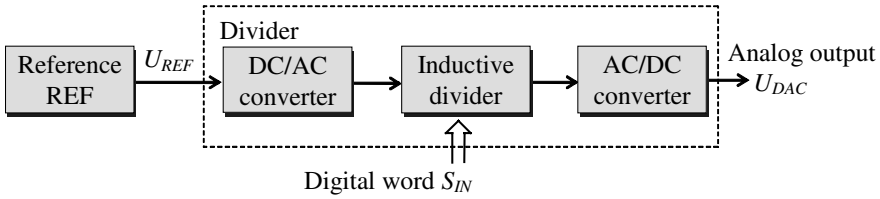


Fig. 4.9. Block diagram of the DAC with an inductive voltage divider

The method used in this DAC, like differential methods, has very good immunity from law frequency disturbances. The disturbances between the DC/AC converter output and the AC/DC converter input are rejected with a Disturbance Rejection

Ratio, DRR, calculated from the formula (Olencki, 1984):

$$\mathrm{DRR} = \frac{f_{\mathrm{TW}}}{2\Pi f_{\mathrm{D}}}, \tag{4.15}$$

where f_{TW} is a trapezoidal waveform frequency and f_{D} is a sinusoidal disturbances frequency. From (4.15) it is clear that the DC and Low Frequency (LF) parts of disturbances are many times rejected.

4.5. Increasing the accuracy of calibrators

One of the calibrators development directions is improving their metrological parameters and, above all, decreasing the output error. The output error of calibrator is the sum of the reference source error and other analog circuit errors. The reference source error δU_{REF} ((4.5) and (4.6)) defines a theoretical low limit of calibrator accuracy.

There are two groups of methods which can decrease the output error: design or technological methods, and structural-algorithm methods. If the difference between the value of the output error and the reference source error is close to zero, then there is only one way to increase the accuracy – improvement design or technology. These can be implemented by the application of higher quality parts or materials, better design of calibrator construction, and protection against influence quantities.

Structural-algorithm methods are divided into the statistical method and the correction of influence quantity effects. The statistical method permits to reduce only random errors, which do not have a conclusive consequence for calibrator accuracy. But the correction of influence factors effects involves the implementation excess of electrical circuits or time. Depending on the participation digital part of the calibrator in the execution of structural methods of automatic correction, we can divide those methods into analogue methods and digital methods.

Depending on the type of reference standard application we can divide digital methods of error correction into

- methods which are called digital adjust with the use of the outside standard,
- methods which are called autocalibration with the use of the inside standard.

Both of these methods consist of two stages: the stage of correction calculating and the stage of error correction. In digital adjustment methods the corrections are calculated by the outside standard. Then the digital adjustment completely substitutes the analogue adjustment. In autocalibration methods, the corrections are calculated by the internal standard. Then the autocalibration can be done by users themselves. In both methods the correction coefficients are first calculated and then applied to error correction.

The corrected value of the setting X_K is calculated in such a way that the real output value Y_R, for any setting, reconstructs the nominal profile of the process. It means that the real output value Y_R meets the following requirement:

$$Y_R = X_K. \tag{4.16}$$

On the basis of the equation (4.3) we can calculate the value of X_K from the following equation:

$$X_K = (\delta_M Y + 1)\ X + \Delta_A Y, \tag{4.17}$$

where $\Delta_A Y$ is the additive component of the error related to the output, $\delta_M Y$ is the multiplicative component of the error related to the output, and X is the setting.

The coefficients $\Delta_A Y$ and $\delta_M Y$ are calculated from the following set of equations (Szmytkiewicz, 1998):

$$\begin{aligned} X_O &= (\delta_M Y + 1)\, X_1 + \Delta_A Y, \\ X_M &= (\delta_M Y + 1)\, X_2 + \Delta_A Y, \end{aligned} \tag{4.18}$$

where Y_O and Y_M are the values of output quantity, for which the adjust is executed (points of adjustment), and X_1 and X_2 are the settings, which are modified settings X_O and X_M. The settings X_O and X_M are set via keyboard and they are modified in such a way that the results of the measurement are equal Y_O and Y_M.

When we solve the above set of equations and we substitute $X_M = Y_M$ and $X_O = Y_O$, we can calculate the coefficients $\Delta_A Y$ and $\delta_M Y$:

$$\delta_M Y + 1 = \frac{X_M - X_O}{X_2 - X_1} \quad \text{and} \quad \Delta A_Y = X_O - X_1 \frac{X_M - X_O}{X_2 - X_1}. \tag{4.19}$$

The coefficients are calculated and stored in the digital part of the calibrator. At the stage of correction, on the basis of coefficients calculated from the system of equations (4.19), the corrected values of the settings from the equation (4.17) are calculated.

In Fig. 4.10 are presented the nominal $Y_N(X)$ and the example of a real $Y_R(X)$ profile of the calibration process. This figure describes the algorithm of digital adjustment. In the figure the are the following symbols: Y_O and Y_M are the values of the output quantity chosen as the measurement point for which the adjust is executed (points of adjustment), X_O and X_M are the values of the setting fit the values of the output quantity Y_O and Y_M when the calibrator works according to the nominal profile of the process, X_1 and X_2 are the values of the setting fit the values of the output quantity Y_O and Y_M when the calibrator works according to the real profile of the process, X and Y represent any value of the setting and the fitting value of the

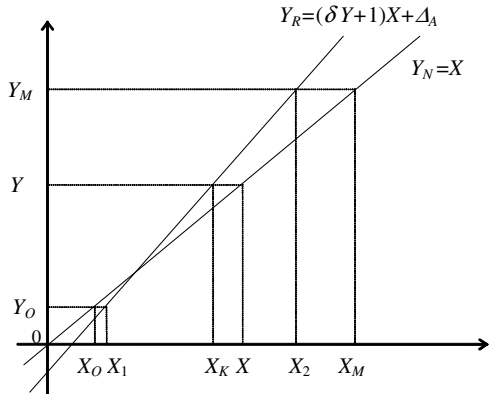


Fig. 4.10. Nominal Y_N and sample real Y_R profile of the process

output quantity when the calibrator works according to the nominal profile, and X_K is calculated at the stage of the correction value of X.

The realization of the adjustment is a laborious and responsible task. The stage of calculating correction determines the laboriousness of the adjustment. Then the ergonomics of adjustment needs special attention when the algorithm is worked out. There are two possible ways to design the stage of calculating correction:

- first – manual from the calibrator keyboard, implemented in the C101 multifunction calibrator (Szmytkiewicz, 1998; 2000),
- second – half automatic from the computer via an interface connection, implemented in the C300 three phase power calibrator (Calmet, 2006).

The advantage of the first selection is the possibility to perform the calibrator adjustment in different laboratories without using computer equipment or special software. The advantage of the second selection is

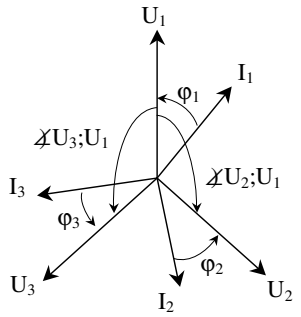
- decreasing the probability of mistakes when the stage of calculating correction is reached,
- saving in memory the correction coefficients allows, after a few years, using these memorizes coefficients to be used for the analysis of the change of calibrator errors,
- after any damage of the digital part of the calibrator it is enough to recall the correction coefficient from computer memory.

4.6. Multiple output calibrators

Typical multifunction calibrators can generate only one quantity at a time – a voltage or a current (Fluke, 1979; Olencki, 1983). For many applications are needed precision sources with more than one channel with an accurately generated value of the voltage or the current (Carullo *et al.*, 1998). Practically in laboratories the are used two or six channel calibrators. The so-called "one phase" power calibrator has two outputs and can generate at the same time an AC voltage and current (Fluke, 2006a; Rotek, 1989). Also the phase angle between them can be set with high accuracy, so we get the possibility of power simulation. The six channel version can generate three voltages shifted usually by 120^o and 240^o and three currents shifted like the voltages plus an additional phase shift between the voltage and the current. Such a system allows simulating a three phase power network (Calmet, 2006).

One phase and three phase calibrators can be used for adjustment and checking measurement equipment, especially electricity meters and power analyzers, etc (Coombes, 2006; Fluke, 2006b; 2006c). They can generate reference vectors of the voltage and current, which is shown in Fig. 4.11.

Figure 4.12 presents a block structure of the one phase calibrator (Olencki and Urbański, 1998). The structure can be divided into a digital part with control unit CU and an analogue part. This part consists of a generator G, a phase shifter PS, a voltage power amplifier VP and a current power amplifier CP. At the output of calibrator terminals, active power P is simulated according to the equation (4.20) and



U_1; U_2; U_3	sinusoidal phase voltage
I_1; I_2; I_3	sinusoidal phase current
φ_1; φ_2; φ_3	phase angle shift
$\measuredangle U_2$; U_1	angle shift between voltage U_2 and U_1
$\measuredangle U_3$; U_1	angle shift between voltage U_3 and U_1
f	frequency of voltage and current

Fig. 4.11. Vector diagram of three phase power calibrator output

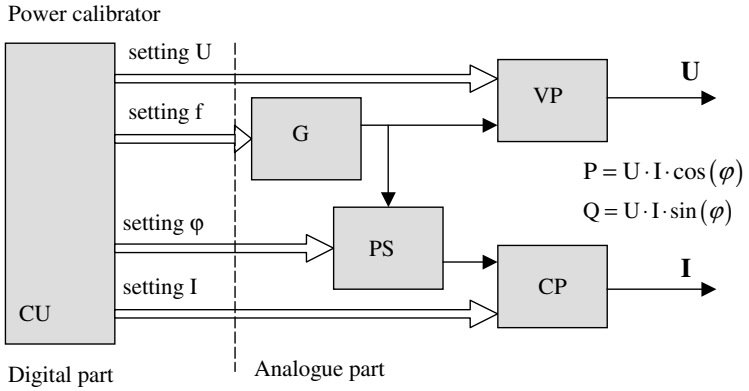


Fig. 4.12. Structural scheme of a power calibrator

reactive power (Q) according to the equation (4.21):

$$P = UI\cos(\varphi), \tag{4.20}$$

$$Q = UI\sin(\varphi), \tag{4.21}$$

where U is the setting value of the sinusoidal voltage, I is the setting value of the sinusoidal current, f is setting value of the frequency, and φ is the setting value of the phase shift between U and I.

From the control unit CU to the analogue part of the calibrator there are connected signals as follows: the setting of the voltage (U), the setting of the current (I), setting of the phase shift (φ), and setting of the frequency (f).

The nominal characteristics (4.20) and (4.21) describe the relationship between the output quantity (power) and the settings at the input. The real characteristics P_R and Q_R of the power calibrator have a systematic error of adjustment and are given by the equations (4.22) and (4.23):

$$P_R = \left[(\delta U + 1)\,U + \Delta U\right]\left[(\delta I + 1)\,I + \Delta I\right]\cos\left(\varphi + \Delta\varphi\right), \tag{4.22}$$

$$Q_R = \left[(\delta U + 1)\,U + \Delta U\right]\left[(\delta I + 1)\,I + \Delta I\right]\sin\left(\varphi + \Delta\varphi\right), \tag{4.23}$$

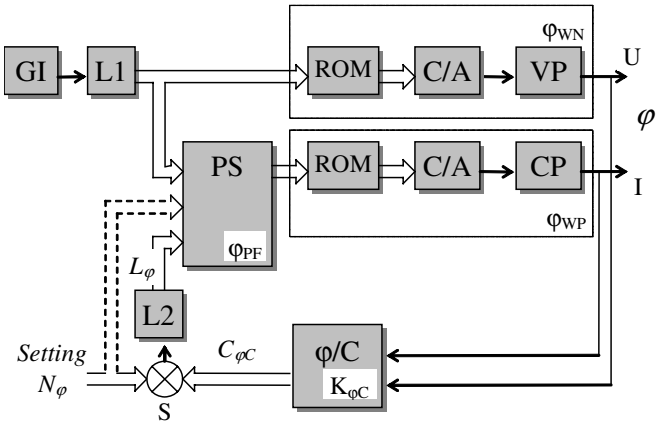


Fig. 4.13. Block diagram of a power calibrator with a closed tracking structure and error corrections

where δU, ΔU represent the multiplicative and the additive part of the voltage U error, δI, ΔI represent the multiplicative and the additive part of the current I error, and $\Delta\varphi$ is the error of the phase shift angle φ.

The difference between real and nominal characteristics gives an active power error δP and a reactive power error δQ of the power simulated at the output of the calibrator. The equations of these errors are (Urbański, 2001):

$$\delta P = \frac{P_R - P}{P} = \delta U + \frac{\Delta U}{U} + \delta I + \frac{\Delta I}{I} + \frac{\cos(\varphi + \Delta\varphi) - \cos\varphi}{\cos\varphi}, \tag{4.24}$$

$$\delta Q = \frac{Q_R - Q}{Q} = \delta U + \frac{\Delta U}{U} + \delta I + \frac{\Delta I}{I} + \frac{\sin(\varphi + \Delta\varphi) - \sin\varphi}{\sin\varphi}. \tag{4.25}$$

The equations (4.22)–(4.25) describe the static properties of the power calibrator by connecting the output quantities P or Q with the settings U, I, φ and f.

Figure 4.13 presents a block diagram of a real power calibrator developed by means of a closed tracking structure with error corrections (Olencki and Urbański, 1994a; 1994b). The signal of the setting N_φ is compared with the output signal φ converted by the phase shift angle converter (φ/C) to a digital code. The result of the comparison, as a coefficient L_φ, is added by the phase shifter PS to the main signal (output of the counter L1). The presented structure consists of a voltage channel (U), which contains an impulse generator GI, a counter L1, memory ROM with a stored shape of the signal, a digital to analog converter C/A and a voltage power amplifier VP. The second channel I consists of phase shifter PS (as a code adder), read only memory ROM, a digital to analog converter C/A and a current power amplifier CP. Astatic characteristics of control are delivered by means of a counter L2 connected to the correction path and additionally, the adder S and a phase shift angle to digital code converter (φ/C). In a stable state there is given set of equations (Urbański,

2001):

$$\begin{cases} C_{\varphi\mathrm{C}} = N_{\varphi}, \\ C_{\varphi\mathrm{C}} = \varphi\, K_{\varphi\mathrm{C}}, \end{cases} \tag{4.26}$$

which allows us to describe the nominal characteristics of the calibrator and its error by the equations

$$\varphi = \frac{N_{\varphi}}{K_{\varphi\mathrm{C}}}, \tag{4.27}$$

$$\Delta\varphi = R_{\mathrm{PF}} - \delta K_{\varphi\mathrm{C}}\, N_{\varphi} - \Delta_{\varphi\mathrm{C}} - R_{\varphi\mathrm{C}} + R_{\mathrm{L2}}, \tag{4.28}$$

where $K_{\varphi\mathrm{C}}$ is the converting coefficient of the phase shift angle converter φ/C, $\delta K_{\varphi\mathrm{C}}$ is the φ/C converter multiplicative part of the error, $\Delta_{\varphi\mathrm{C}}$ is the φ/C converter additive part of the error, $R_{\varphi\mathrm{C}}$ is the resolution of the φ/C converter, R_{PF} is the resolution of the adder PS, and R_{L2} is the resolution of the counter L1.

In the equation (4.28), there are no parasitical phase shifts φ_{WN} and φ_{WP} of output amplifiers, which is an advantage of this structure. The phase shifter PS and the counter L2 are made as digital circuits, and their resolution R is related to the frequency of the impulse generator GI according to the equation

$$R\,[^{\circ}] = \frac{360 f}{f_{\mathrm{GI}}}, \tag{4.29}$$

where f is the frequency of the output signals, and f_{GI} is the frequency of the impulse generator GI. So errors caused by the limited resolution R_{PF} and R_{L2} of the phase shifter PS and the counter L2 can be made considerably small.

4.7. Calibrator as a test system

The idea of a connection three phase calibrator and additional meters with a computer equipped with specialized software gives a new kind of device – a three phase power calibrator and an electric equipment automatic tester (Olencki, 2006). In Fig. 4.14. this calibration and test system is presented. It consists of a calibrator and a computer with software. The calibrator has a precision three phase generator and a set of additional inputs and meters:

- impulse counter S0 for counting the output impulses from electricity meters,
- direct current ammeter Idc for checking industrial transducers,
- DC voltmeter Udc for checking industrial transducers or DC current clamps,
- AC ammeter Iac for checking current clamps or current transformers for measurements,
- timer t_O for starting relay time measurements.

This idea is applied to design a three phase automatic calibration system model C300 (Calmet, 2006) called the "calibrator". This calibrator can be used for measurements of two kinds of calibration characteristics:

- error curves (see Fig. 4.15(a)) of electricity meters, measurement industrial transducers, current clamps and current transformers in a fully automatic way,

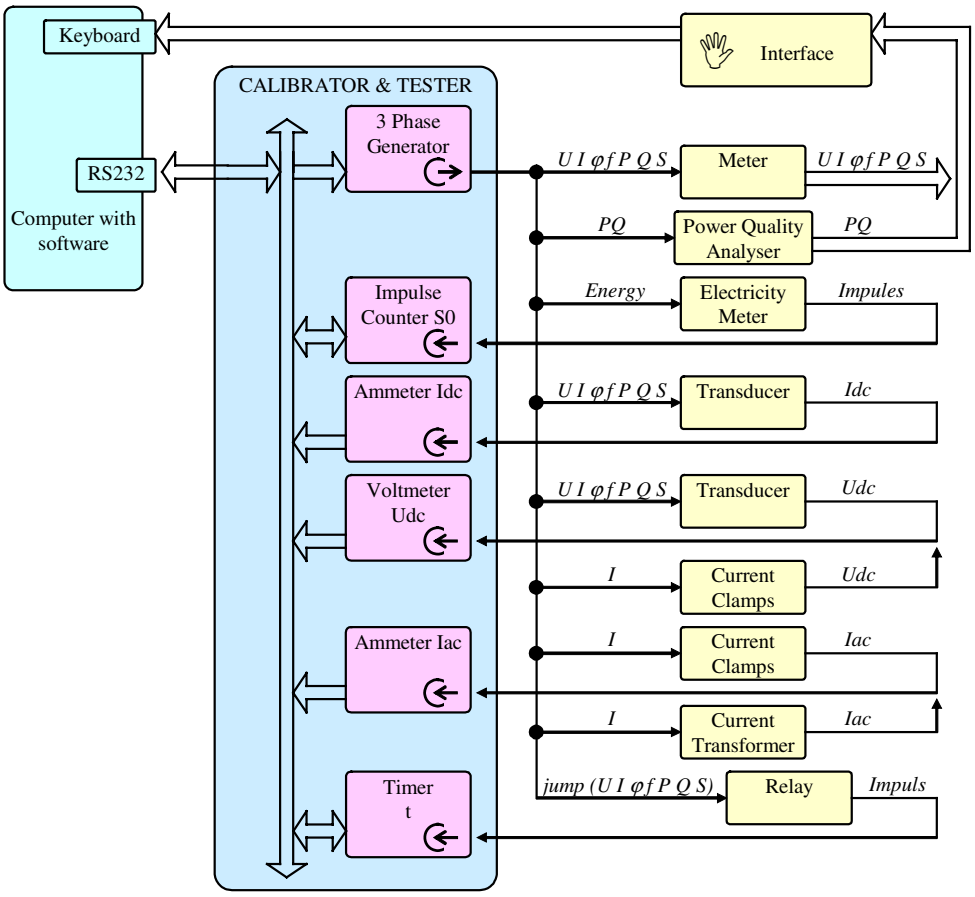
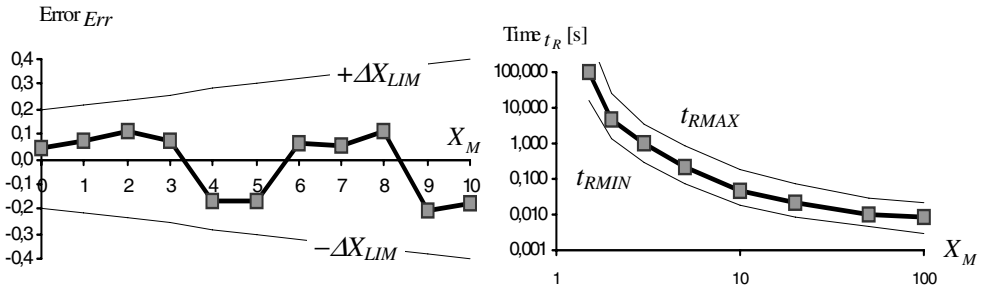


Fig. 4.14. Block diagram of the three phase automatic calibration system



(a) (b)

Fig. 4.15. Characteristics of the equipment under the test:
(a) error curves, (b) response time curves

- time curves (see Fig. 4.15(b)) of protection relays, e.g. overcurrent relays, in a fully automatic way,
- error curves of the analogue or digital meters and power quality analysers in a semi automatic way.

The error Err and response time t_R characteristics can be presented as a graph or a table. Efficient testing and performance analysis require well-defined reference values ΔX_{LIM} in Fig. 4.15(a) and t_{RMIN}, t_{RMAX} in Fig. 4.15(b). The C300 calibration system can automatically create the reference values on the basis of customer requirements. Specialized software will compare the actual measurement result with the reference values and check for any deviations from the reference values. The results are correct for the following requirements:

$$-X_{LIM}(X_M) < Err(X_M) < +X_{LIM}(X_M), \tag{4.30}$$

$$t_{R\ MIN}(X_M) < t_R(X_M) < t_{R\ MAX}(X_M). \tag{4.31}$$

4.8. Conclusions

The first calibrators were designed with structures similar to manually controlled systems. In the next calibrators, for stabilizing the amplitude of voltages or currents in multifunction and power wide range calibrators are used closed single or multi loop structures and PI controllers. For stabilizing phase angles in power calibrators there are used closed structures with an additive correction of errors. Static and dynamic analyses are required to achieved good calibrator parameters: high accuracy and short settling times. In most of the modern calibrators there are used PWM DAC, and autocalibration and digitally adjustment methods are implemented to achieved high accuracy. The automation of calibration procedures is particularly important for checking three phase electrical devices, e.g. energy meters or protection relays. For this purpose there is presented the idea of the three phase automatic calibration system, which is implemented in the compact three phase power calibrator model C300, dedicated to calibrate extremely wide range electrical devices.

References

Calmet (2006): *Three phase power calibrator and meter tester.* — Calmet, `http://www.calmet.com.pl`.

Carullo A., Ferraris F., Parvis M. and Vallan A. (1998): *Phantom power generator for the calibration of wattmeters in distorted environments.* — Proc. IMECO TC-4 Symp. *Development in Digital Measuring Instrumentation* and 3-rd Workshop *ADC Modeling and Testing*, Naples, Italy, pp. 67–71.

Coombes D. (2006): *Improving accuracy of power and power quality measurements.* — Fluke Precision Measurement Hurricane Way, Norwich, UK, `http://assets.fluke.com`.

Dziennik Normalizacji i Miar (DzNiM) (1978): *Zarządzenie Nr 13 Prezesa PKNMiJ z dnia 9 lutego 1978 r. w sprawie ustalania przepisów o sterowanych źródłach odniesienia (kalibratorach) napięcia stałego.* — No. 4, Warsaw, (in Polish).

Dziennik Normalizacji i Miar (DzNiM) (1984): *Zarządzenie Nr 55 Prezesa PKNMiJ z dnia 3 grudnia 1984 r. w sprawie ustalania przepisów o kontrolnych źródłach odniesienia wartości skutecznej napięcia sinusoidalnego w paśmie częstotliwości od 10Hz do 100kHz. Przepisy o legalizacji i sprawdzaniu narzędzi pomiarowych.* — No. 16, Warsaw, (in Polish).

Fluke (1979): *5100 Series B Calibrators.* — Instruction Manual, John Fluke Inc.

Fluke (2006a): *The Fluke 6100A Electrical Power Standard.* — Fluke, `http://www.fluke.com/library`.

Fluke (2006b): *9100 Universal Calibration System.* — `http://us.fluke.com`.

Fluke (2006c): *Using the 6100A Electrical Power Standard to calibrate energy meters.* — Fluke, Application Note, `http://www.fluke.com/library`.

Grimbleby J. (2004): *Digital-to-Analogue Conversion. Analogue-to-Digital Conversion.* — The University of Reading, Unit EE2C2:1, `http://www.elec.rdg.ac.uk`.

IEC 443 (1974): *Stabilized supply apparatus for measurement.* — Publication IEC 443, Geneve.

Lange Z. and Olencki A. (1986a): *Circuit for reference alternate waveform generation.* — Patent PL No 134147, (in Polish).

Lange Z. and Olencki A. (1986b): *Circuit for digital to analog conversion.* — Patent PL No. 133457, (in Polish).

Olencki A. (1982): *Calibrator of constant voltage and current and alternate sinusoidal waveforms type GA1.* — Pomiary, Automatyka, Kontrola, PAK, No. 8–9, pp. 280–281, (in Polish).

Olencki A. (1983): *Digital to analog converter for calibrator.* — Pomiary, Automatyka, Kontrola, PAK, No. 8, pp. 259–260, (in Polish).

Olencki A. (1984): *Problem of digital to analog converter dynamic properties improovement in multifunction voltage and current calibrator.* — Wrocław University of Technology, PhD thesis, (in Polish).

Olencki A. (1991): *Calibrators of electric quantities.* — Kiev: Institute of Electrodynamics of the Ukrainian Academy of Science, D.Sc., (in Russian).

Olencki A. (1998): *Philosophy of voltage and current meter testing – the art of multifunction calibrators design.* — Pomiary, Automatyka, Kontrola, PAK, No. 9, pp. 358–361, (in Polish).

Olencki A. (2006): *Three phase power calibrator and automatic tester of electric devices.* — Pomiary, Automatyka, Kontrola, PAK, No. 6 bis, pp. 106–108, (in Polish).

Olencki A. and Szmytkiewicz J. (1999): *Analysis of possibilities of metrological parameters improvement in multifunction AC/DC voltage and current calibrator.* — Computer Aided Metrology Conference, Rynia near Warsaw, Vol. 2, pp. 123–130, (in Polish).

Olencki A. and Urbański K. (1994a): *One Phase Calibrator.* — Patent PL No. 163005, (in Polish).

Olencki A. and Urbański K. (1994b): *Three Phase Calibrator.* — Patent PL No. 163006, (in Polish).

Olencki A. and Urbański K. (1998): *AC power calibrators.* — Proc. Polish *Metrology Congress*, Gdańsk, Poland, Vol. 3, pp. 233–241, (in Polish).

Rotek (1989): *Precision Wattmeter and Watthour Meter Calibrators series 800A/811A.* — Instruction Manual, Rotek Instrument Corp.

Szmytkiewicz J. (1990): *Calibrator type SQ10.* — Wiadomości Elektrotechniczne, No. 1–2, pp. 24–25, (in Polish).

Szmytkiewicz J. (1998): *Digital adjustment of multifunction calibrator.* — Zeszyty Naukowe Politechniki Śląskiej, Series *Elektryka*, No. 162, Gliwice, pp. 115–122, (in Polish).

Szmytkiewicz J. (2000): *Analysis of Possibilities of Metrological Parameters Improvement in Multifunction AC/DC Voltage and Current Calibrator.* — Technical University of Zielona Góra, Faculty of Electrical Engineering, PhD thesis, (in Polish).

Urbański K. (2001): *AC Power Calibrator Structures and Algorithms of Work. Development and Properties Analysis.* — Ph.D. thesis, Technical University of Zielona Góra, Faculty of Electrical Engineering, (in Polish).

Chapter 5

ASSIGNING TIME PARAMETERS OF DISTRIBUTED MEASUREMENT–CONTROL SYSTEMS

Emil MICHTA*, Adam MARKOWSKI*

5.1. Introduction

The evolution of Distributed Measurement-Control Systems (DMCSs) structures and the availability of advanced electronic circuits solutions for communication in industrial networks, observed in the few last years, have created convenient circumstances for building networked measurement-control systems. The basic components of such systems are intelligent nodes based on microprocessors. An important feature of an intelligent node is data processing and bidirectional communication with other DMCS nodes. During the last years the significance of DMCSs has increased because, besides having a direct influence on a controlled or supervised process, they are a fundamental source of measurement data for many applications such as a visualization systems, diagnostic systems or expert systems. As a consequence, it is important to combine the performance of nodes with the communication part of DMCS a in such way that all tasks will be done on time and the response time will assure that the deadline won't be exceeded.

DMCS evolution, from the so-called "multiplexer structure" to the "network structure", created a quite new setting for its design strategy (Michta, 2000). The basic demand is that processing and communication tasks executed in DMCS abide deadlines. That is why deadline analysis should be done at the system design stage. To asses correctly if a task's deadline is not exceeded, it is necessary to see a system as a set of simultaneously shared and cooperating components. DMCS task deadline analysis inseparably involves the measuring of data transmission delays, from their source to the destination point in a system. Computer aided DMCS design tools should allow modeling such situations.

* Institute of Electrical Metrology
e-mails: {E.Michta, A.Markowski}@ime.uz.zgora.pl

A brief review of the state-of-the-art field of DMCSs shows that, as has already been emphasized, more attention is paid to the problem of time parameters calculation (Jakubiec and Al-Raimi, 1999; Kwiecień, 2000; Michta, 2002; 2005; Syndenham and Thorn, 2005). Except the time parameters, plenty of other parameters describing such a systems, e.g. performance, utilization, are defined. From the engineering point of view, the most important DMCS parameter is the response time of a system to the events in a object, process or environment. It could be pointed out that most papers dealing with DMCSs concentrate on the investigation of the system communication part. In some cases such an approach is not sufficient. Besides, it lacks quality and quantity indicators of system correctness.

Among the cases described in the literature, research approaches in the field of data transmission delay take into account all kinds of final results. Several groups can be distinguished. To the first one belong analytical methods, which enable delays calculation based on the "Worst-Case-Executing Time" (WCET) (Audsley *et al.*, 1997; Michta, 2000). Such an approach allows calculating delays for particular nodes and for the whole system. The WCET approach gives results very quickly but sometimes they are very pessimistic and lead to poor utilization of processing and communication resources, particularly if a task executing time C is varying in a wide range and worst cases are seldom. Analytical WCET methods do not allow obtaining delay histograms, which are often very important for DMCS designers.

Besides the analytical approach, the determination of DMCS time parameters can be done by means of simulation or experimental approaches (Andersson *et al.*, 2005). The simulation technique is a powerful one that can be used at several stages of DMCS development. It should be possible to simulate the computation that takes place within the nodes and communication between the nodes, simultaneously. This chapter presents a simulation environment that facilitates the simulation of a DMCS with measuring and actuator nodes and a communication network. Case studies covering analytical, simulation and experimental approaches are presented.

5.2. Reasons of delays in DMCSs

Often, measuring data that arise in one system node and are then passed on by the system communication part and used for control purposes in another node are a source of delays (Kim and Lee, 2002; Markowski, 2004; Żaba, 2003). To define delays in a DMCS, usually we take into account the time passing from data sampling instant (t_1) in the measuring node to use in the actuator node (t_2); Fig. 5.1. Usually a few to several dozens of nodes are connected to the network segment. To exchange messages, nodes use a shared communication medium. Rules of sharing the network medium are set by the communication protocol applied. The periodic message transmission delay t_d introduced by the communication system depends on all other transmissions done in the DMCS, and during succeeding transmissions can be different and changes from $t_{2\,\min}$ to $t_{2\,\max}$. In a real system there are more shared resources. First of all in the nodes there are microprocessors which are shared by executed programs, realize different tasks done by nodes, e.g. measuring, control, data processing or service of communication. Another group of shared resources are data sending buffers, which

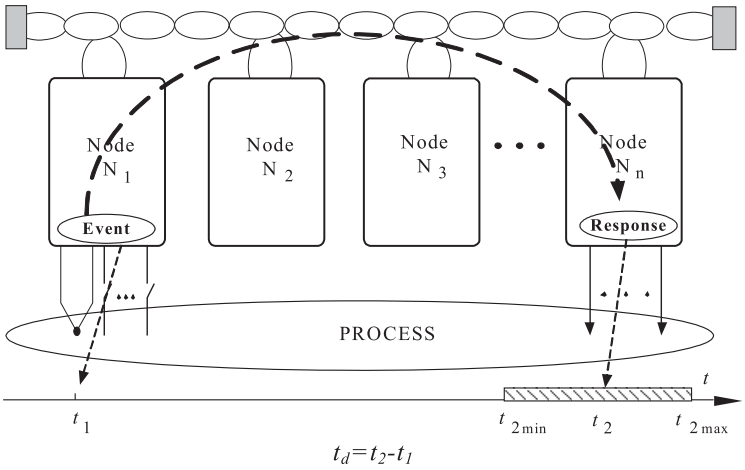


Fig. 5.1. Shared segment of the DMCS

are used to store data before their transmission. During DMCS timing analysis the influence of all shared resources should be taken into account.

To do a timing analysis of the shared resources in a DMCS, the use of a scheduling theory, a well-known approach from the field of computer science, can be very useful. The scheduling method allows allocating processor time or other shared resources to the tasks in such a way that all task deadlines are met. In this chapter, node delays caused by programs executed by the processor and delays in the communication system are taken into account during the timing analysis. The basic aim of our work was searching for relations between delays in DMCS data transmission, looking for their causes and influence on DMCS functioning. There is a diversity of situations in which delays are present. The complexity of analytical description of this kind of relations and the requirement for the knowledge of the quantity and kind of delay at the stage of design determine the range, importance and way of conducting the research using a DMCS simulation model. This approach is very useful because the results obtained during simulation can be used by DMCS designers to choose a suitable structure of the designed system or node, and can indicate if task deadlines executed by the node processor and communication tasks deadlines are met.

5.3. Time parameters assigning approaches

Currently designed measurement-control systems (MCSs) are often distributed systems, where sensors and actuators are located in different nodes connected by a network. Distributed systems with decentralized intelligence are more flexible but also more non-deterministic. Communication protocols used in the DMCS introduce timing variations that influence task scheduling in the nodes (Michta, 2000).

In many applications, the correctness of the DMCS depends not only on the logical results of the computations but also on the time at which the results are

produced. To design such a system, with real-time constraints, the use of aided tools is required. Many different methods and tools are used in the development of a DMCS. In this chapter, three approaches to assigning time parameters of a DMCS are presented. The first one is an analytical approach based on the well-known scheduling theory. The next one is an experimental approach that requires building a real system copy and investigating of it in pursuit of time parameters. The most attention is given to simulation approaches because they are a powerful technique that can be used at several stages of DMCS development. It should be possible to simultaneously simulate the computations that take place within the nodes, and the communication between the nodes, the sensor and the actuator.

A large number of general network simulators exists today. One of the most well known is the ns-2, which is a discrete-event simulator for both wired and wireless networks, supporting routing, transport and multicast protocols. However, its usefulness for the simulation of a DMCS is limited. There are also same network simulators dedicated to the sensor network domain. *TOSSIM* derives directly from the TinyOS code and is quite useful (Lewis *et al.*, 2003). *Network in A Box* (*NAB*) is another simulator for a large-scale sensor network. Another example is *J-Sim*, a general simulation environment that includes a packet-switched network model that may be used to simulate sensor networks. The types of simulators mentioned above generally lack the ability to simulate continuous-time dynamics and the inner structure of the nodes that is present in our simulator presented in the chapter.

The next group of simulators is dedicated to DMCS environment. The simulator *RTSIM* has a module that allows system dynamics to be simulated in parallel with scheduling algorithms. *XILO* supports the simulation of system dynamics in Controller Area Network (CAN) networks with priority-pre-emptive scheduling. *Polotemy II* is a general purpose multi-domain modeling and simulation environment that includes a continuous-time and a simple Real-Time Operating System (RTOS) domain (Baldwin *et al.*, 2004). Recently it has been extended to the sensor network domain also. Two MATLAB-based toolboxes: *Jitterbug* and *TrueTime*, for the analysis and simulation of real-time control systems have been developed by the Swedish Lund Institute of Technology. These tools make it possible to investigate the impact of delay, jitter, data loss on control performance. The scheduling and execution of control tasks is simulated in parallel with network communication. The strength of *TrueTime* are co-simulation facilities that make it possible to simulate latency-related aspects of network communication in combination with node computations.

The chapter presents a new simulation environment that facilitates DMCS simulation with measurement and actuator nodes and communication networks, developed in our Institute. The structure of the simulation model, its basic features and examples of simulation case studies will be presented.

5.4. DMCS communication model

5.4.1. Communication model

In the works (Michta, 2000), a communication model of a DMCS, presented in Fig. 5.2, is proposed. The model contains nodes connected by a communication system. Usu-

ally, most of the nodes are installed in the process, object or environment. However, a few from them can be used to run applications which use data fetched from the measurement nodes. Scheduling and task switching functions are realized by processors in the nodes, taking their to complete those and blocking other tasks from being executed.

The DMCS shown in Fig. 5.2 is implemented as a set of embedded nodes. Each node runs a number of measuring, actuator and local tasks. These tasks communicate with each other by passing messages between the nodes across a shared network. In order to meet timing requirements, end-to-end communication delay composed by sender node generation delay, queuing delay during accessing the network, transmission delay and delivery delay in the receiver, the node must be bounded.

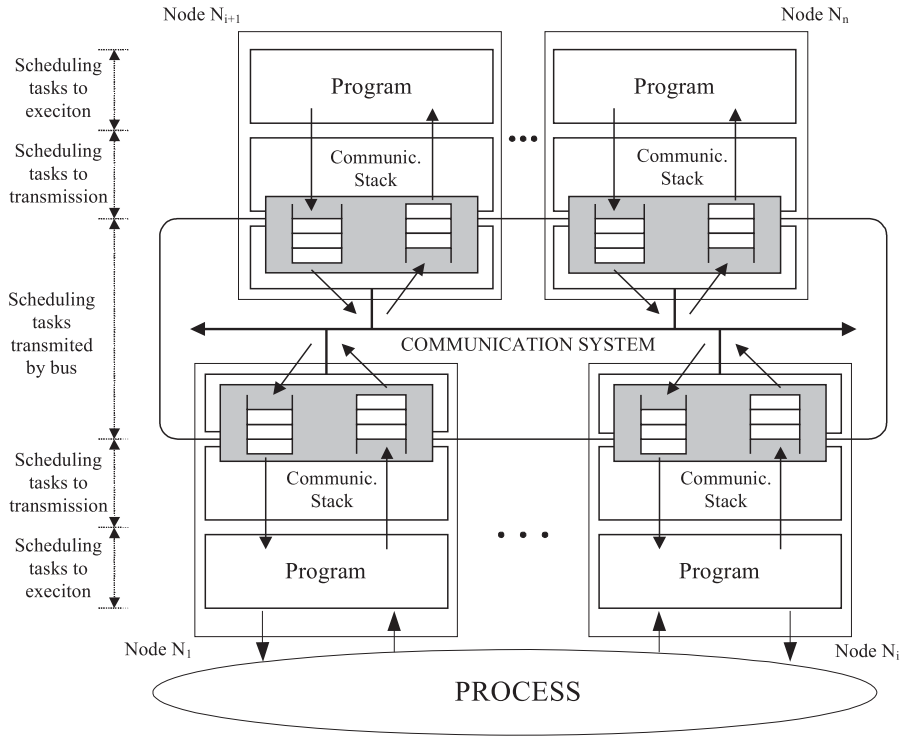


Fig. 5.2. DMCS communication model

Let us consider techniques that can be applied both to software within nodes and to messages passing on shared buses to assure that the deadlines are met. To analyse time constraints of a networked, distributed system, we have to distinguish three functional layers: the node application layer, the communication stack layer and the network layer, shown in Fig. 5.2. During the analysis of each layer, a combination of different scheduling policies can be used. For example, the network layer may be priority driven with the use of non-pre-emptive Rate Monotonic (RM) or Deadline Monotonic (DM) polices while nodes may be priority or time driven with the use of both pre-emptive and non-pre-emptive approaches. Different nodes may have different

scheduling policies due to manufacturer preferences, thus during the DMCS designing phase we have a real influence mainly on the choice of a communication system. RM, DM and EDF (Earliest Deadline First) are priority based scheduling algorithms so the system must have an adequate number of priority levels that can be assigned to tasks in each layer.

The primary objective during the design phase is to decouple the scheduling of resources and to analyse the scheduling of each node processor and each network. As we can see in Fig. 5.2, the same techniques of the analysis of task scheduling on the processor node and message scheduling across the network can be used. The decoupling of resources allows us to divide the scheduling problems and simplify the development and maintenance of nodes and the NMCS itself.

5.4.2. System task model

Let us consider a simple message passing cycle in the DMCS shown in Fig. 5.3 with three basic components, i.e. communication bus and two nodes: the Measuring Node (MN) and the Actuator Node (AN). The nodes cooperate through the network to provide end-to-end functionality. In a real-time system this end-to-end functionality, from event to response, must be provided within a specified deadline. The response time R_i on the event i should be shorter than the deadline D_i. Timing analysis is applied to a resource shared between multiple activities. According to the DMCS model shown in Fig. 5.2, on an application level, in the sensor and actuator node several functions are executed by a single microprocessor. On a communication stack level, output and input messages wait in a queue to be served. Finally, on the network layer one bus carries a number of messages sent by the nodes.

To simplify the timing analysis of measuring node to actuator node communication, we can conduct a partition analysis of components that can be analysed independently, as shown in Fig. 5.3. For the task τ_i, release jitter time, blocking time and interference time are joined together and represented by the grey area (waiting time) to the left in the execution time windows shown in Fig. 5.3. End-to-end response time depends on the response time of each component involved in the deadline to be met. Different offline scheduling policies could be investigated using this simple control loop as an example.

Shown in Fig. 5.3 are the execution times of tasks on the two nodes (measuring and actuator) and the communication task execution time on the bus. An execution window represents each activity. The left end of the box represents the arrival of the task and the right end represents the latest task completion. The length of the box represents the task response time on a given level. The activity may be finished at any time after the minimum execution time represented by the length of the white box. In the worst case, release jitter and blocking and interference from other tasks may delay task execution.

5.5. Scheduling theory in DMCS analysis

Schedulability analysis can be performed online or offline. In the online case the schedulability of the task set is analysed at run-time, whereas in the offline case it is

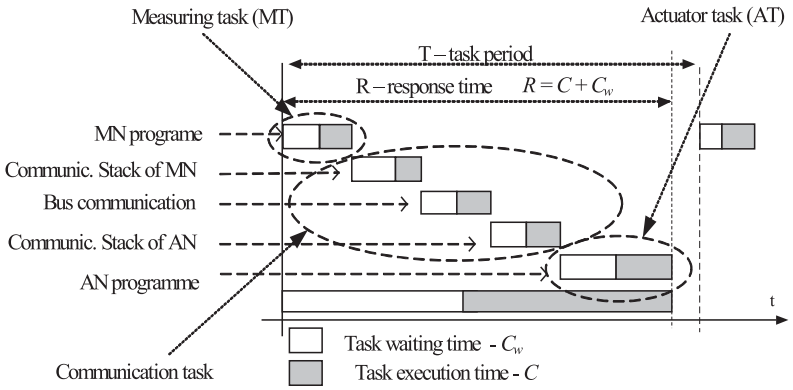


Fig. 5.3. Task model

performed prior to run-time. The offline scheduling requires little run-time overhead and the schedulability of the task set is guaranteed before execution. However, it requires a prior knowledge of tasks timing characteristics. If the tasks characteristics are not known prior to run-time, schedulability analysis must be performed online. There are two basic types of online schedulers: *planning-based* and *best-effort* schedulers. In the *planning-based* type, when a new task arrives, the scheduler tries to re-define a new schedule, which is able to comply with both the requirements of the new task and the requirements of the previously scheduled tasks. The new task is only accepted for execution if the schedule is found feasible. In the best-effort type, when a new task arrives, the scheduler does not try to perform a new schedule. The new task is accepted for execution, and the system tries to do its best to meet task deadlines (Michta, 2000; Syndenham and Thorn, 2005).

Offline scheduling paradigms depend on whether the schedulability analysis produces itself a schedule according to which tasks are dispatched at run-time. The *table-driven* approach is the best-known example of offline scheduling that produces a schedule. The *priority-based* approach is an example of offline scheduling where no explicit schedule is constructed. At run-time, tasks are executed in a highest-priority-first basis. Priority-based approaches are much more flexible and accommodating than *table-driven* approaches.

5.5.1. Task priority assignment schemes

The most popular priority assignment scheme is to give the tasks a priority level based on its period: the smaller the period, the higher the priority. Thus, if they have smaller periods, their worst-case response time must also be smaller. This type of priority assignment is known as the RM rate monotonic assignment. RM scheduling theory ensures that, for a given fixed set of tasks with fixed and known priority ordering, as long as system utilization of all tasks lies below a certain bound and appropriate scheduling algorithms are used, all tasks meet their deadlines. This put the development and maintenance of real-time systems on an analytic basis, making these systems easier to develop and maintain. RM theory was introduced for schedul-

ing independent, periodic tasks with end of period deadlines. RM is an optimal static priority scheduling algorithm for independent periodic tasks with end of period deadlines.

If some tasks are sporadic, it may not be reasonable to consider the relative deadline equal to the period. A different priority assignment can then be to give the tasks a priority level based on its relative deadline: the smaller the relative deadline, the higher the priority. This type of priority assignment is known as DM deadline monotonic assignment.

In both the RM and DM priority assignments, priorities are fixed, in the sense that they do not vary along time. At run-time, tasks are dispatched highest priority first. A similar dispatching policy can be used if the task which is chosen to run is the one with the earliest deadline. This also corresponds to priority-driven scheduling, where the priorities of the tasks vary along time. Thus, the EDF earliest deadline first is a dynamic priority assignment scheme. In all three cases, the dispatching phase will take place either when a new task is released or the execution of the running task ends.

5.5.2. Pre-emptive and non-pre-emptive systems

A common implementation for embedded node software and for a *master-slave* communication system is to use a Static Cyclic (SC) system. This involves static creation of the schedule that typically consists of a number of tasks running one after another to form an overall schedule. A cyclic system does not make effective use of the CPU and the worst-case response time for each task exceeds its period. In a cycling system all tasks must run at harmonic processing rates. Such a solution can be used for nodes with a small number of tasks and for implementations without real-time requirements. To improve effective use of the node CPU and to optimise worst-case response time, tasks priority levels with a pre-emptive or a non-pre-emptive scheduling policy can be used.

In a priority-based scheduler, a higher-priority task may be released during the execution of a lower-priority one. If the tasks are being executed in a pre-emptive context, the higher-priority task will pre-empt the lower-priority one. In a non-pre-emptive context, the lower-priority task will be allowed to complete its execution before the higher-priority task starts execution. This situation can be described as priority inversion because a lower-priority task delays a higher-priority task.

5.5.3. Offline schedulability analysis

There are two basic types of analytical methods to perform pre-run-time schedulability analysis. One is based on the analysis of the processor or network utilization. The other one is based on the response time analysis for each individual task. By considering only the processor or network utilization of a task set, a test for pre-run-time schedulability analysis could be obtained. An analytical approach is used to predict the worst-case response time of each task. The obtained values are then compared with the relative deadlines of the tasks.

The utilization-based test is a simple computation procedure, which is applied to an overall task set. For this reason, such tests are very useful for implementing

schedulers that check feasibility online. However, utilization-based tests do not give tasks response time values. Then constitute sufficient but not necessary conditions. For RM priority assignment it was proved that a set of N independent periodic tasks characterized by a worst case computation time C and a period T scheduled by the rate monotonic algorithm will always meet its deadlines for all tasks, if an utilization-based pre-run-time schedulability test is as follows:

$$U = \sum_{i=1}^{N} \frac{C_i}{T_i} \leq N \times \left(2^{1/N} - 1\right), \tag{5.1}$$

where C_i is the worst-case computation time of the task i, and T_i is the minimum time between task i releases (period).

Utilization-based tests are sufficient but not necessary conditions. This utilization-based test is valid for periodic independent tasks with relative deadlines equal to the period and for pre-emptive systems. For the non-pre-emptive case, a similar analysis can be adapted to include the task i blocking time B_i, during which high priority tasks are blocked by low priority tasks:

$$U = \sum_{i=1}^{i} \left(\frac{C_i}{T_i}\right) + \frac{B_i}{T_i} \leq i \times \left(2^{1/i} - 1\right), \forall_{i,1 \leq i \leq N}, \tag{5.2}$$

where B_i is the maximum blocking a task i by lower priority tasks than the task i.

5.5.4. Response time tests

Task response time or the interval between its minimum and worst-case value is a very important time parameter for hard and safety-critical real-time objects. To confirm that each task meets its deadline, the worst-case completion time of each task is calculated taking into account the influence of the other tasks. It was proved that the worst-case response time R_i, of the task i is found when all tasks are synchronously released at their maximum rate. For that instance R_i can be computed by the following recursive equation:

$$R_i^{n+1} = \sum_{j \in hp(i)} \left(\left\lceil \frac{R_i^n}{T_j} \right\rceil \times C_j \right) + C_i, \tag{5.3}$$

where $hp(i)$ denotes the set of tasks of higher priority than task i priority, the symbol $\lceil \cdot \rceil$ is the ceiling function.

The initial value for R_i is zero. The recursion ends when $R^{n+1} = R^n = R_i$. If the worst-case response time R_i exceeds T_t (in the case of RM priority assignment) or D_i, (in the case of DM priority assignment), the task i is not schedulable. This result is valid for the pre-emptive context. For the case of non-pre-emptive tasks, the utilisation-based test (5.3) was updated to include a blocking factor, during which higher priority tasks can be blocked by low-priority tasks. Taking into account the blocking factor B_i, worst-case response time for the nonpre-emptive approach can be derived from the following recursive equation:

$$R_i^{n+1} = B_i + \sum_{j \in hp(i)} \left(\left\lceil \frac{R_i^n}{T_j} \right\rceil \times C_j \right) + C_i, \tag{5.4}$$

where $B_i = \max\{C_j\}$ for $lp(i)$, where $lp(i)$ denotes tasks with lower priority than the task i.

The above equation was used for analytical analysis of the DMCS. The results of the analysis achieved by analytical, experimental and simulation approaches are presented in Section 7.

5.6. DMCS simulation model

The structure of a DMCS communication model including all shared resources presented in Fig. 5.2 was used to formulate a DMCS simulation model (Markowski, 2005). The simulation model is based on the following assumptions:

- all model parameters are deterministic,
- discrete event simulation will be used: the activity inspection method and the event planning method,
- structure of the model should be scalable according to the possibility of using different task scheduling methods in each simulation model block,
- two media access strategies should be done: *master-slave* and *peer-to-peer*,
- in the nodes, the following kinds of tasks will be executed: local tasks, measuring tasks, actuator tasks and buffer tasks.

5.6.1. DMCS model structure

The simulation environment has been developed based on the DMCS model structure presented in Fig. 5.4.

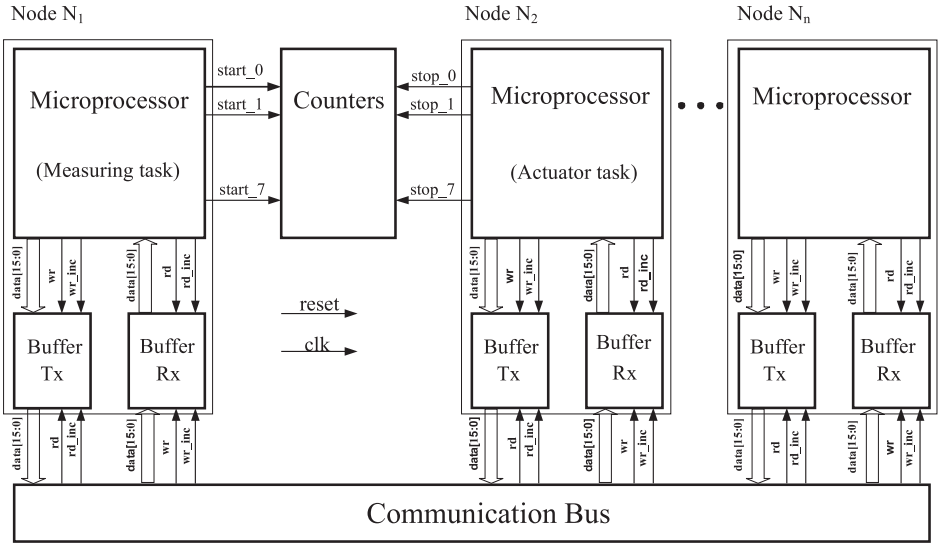


Fig. 5.4. DMCS model structure

To achieve DMCS model universality regarding its different structures and ability to use different task scheduling methods in each module, cooperation between model modules was carried out by means of suitable signals. The data bus, control signals and signals auxiliary to counter service were distinguished. Such a solution gives the opportunity for independent functioning of DMCS model modules and enables each module to be tested independently.

In the nodes of the DMCS model the scheduling of the node tasks is done. Each node cooperates with the assigned receiver and transmit buffer, by means of control signals and the data bus. The buffer stores the data received and the data ready to be sent. The node buffers cooperate with the bus module, which simulates the media access control method for *peer-to-peer* or *master-slave* networks. The system tasks are divided into two kinds of tasks located in two different nodes. The first one is the measuring task, which acquires data from the process. The second one is the actuator task, which uses measured data for process control. Both tasks are joined by the communication task executed in the simulation tool in the bus module.

5.6.2. Simulation model based on the activity inspection method

The algorithm of the simulator program, based on the *activity inspection method*, realized for the above structural model of a system (Fig. 5.4) is presented in Fig. 5.5. After each loop, the clock state is incremented. Simulation is done simultaneously for all nodes, all receiver and transmission buffers and the communication bus. The main simulation program loop is repeated until the loop counter reaches a set value of the clock cycles.

As the simulation is started, the "*model parameterization*" block is executed. Initial values for all variables used during the simulation, such as the number of nodes, node tasks parameters, the communication relationship between the tasks in different nodes, are being set. Then the system clock T is incremented and the simulation "stop condition" is tested. In the next step, tasks in each node are scheduled according to the chosen scheduling strategy (RM or DM). Because some tasks would like to retrieve data from the receiver buffers or would like to write down data to the transmitter buffers, during task scheduling the buffers are checked also and, if it is necessary, suitable output signals are set. After the task scheduling is done in each node, the input signals states of all communication buffers are checked and, depending on the state, an appropriate action is taken. In the simulator, the data write, write pointer incrementation, data read and read pointer incrementation, are distinguished. After updating the state of the communication buffer, the state of the bus is updated. The bus operation depends on the kind of media access control method and relays on data exchange between the nodes. Data exchange is done by reading the data from chosen transmit buffers allocated by the arbitration (for the peer-to-peer access method) or by the exchange scenario (for the master-slave access method). After transmission delay time elapses, the transmitted data are written in the receiver buffer in the destination node. Both read data from the transmitter buffer and write data to the receiver buffer cause setting up the output signals in the bus module.

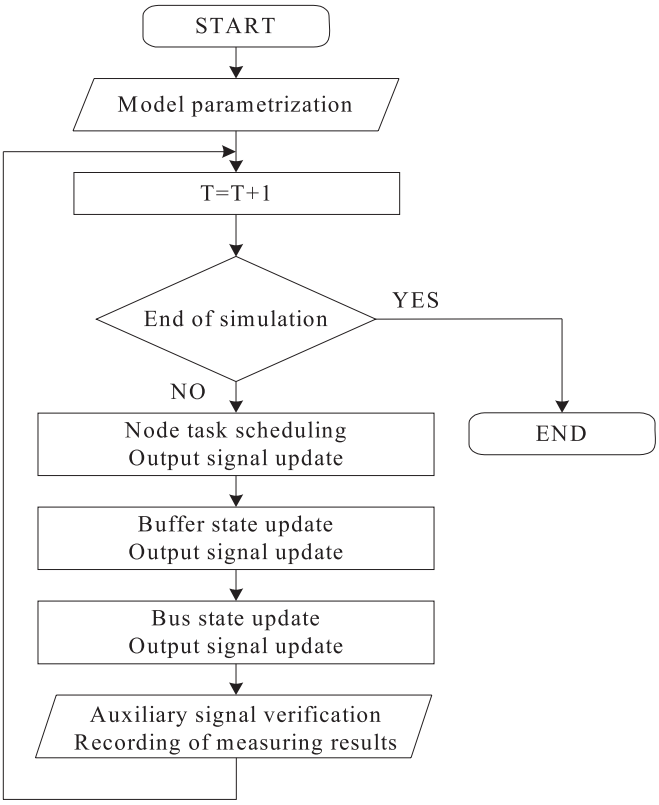


Fig. 5.5. Algorithm of a simulation model program

5.6.3. Results of simulation

Before simulation start, the nodes, communication bus parameters and the time of simulation should be set. When the simulation is running, data illustrating system behavior are gathered. Taking into account the way the gathered data are presented and processed, two groups are distinguished:

(i) data presented in a number form,

(ii) data presented in a histogram form.

For each system task, the following number form parameters are registered: L_{TP} – number of measuring task execution, L_{NP} – number of the task overwritten at the measuring node, L_{TW} – number of actuator (control) task execution, L_{POBO} – number of transmitter buffer correct read-outs, L_{NBN} – number of overwrites in transmitter buffer, $LNBO$ – number of overwrites in receiver buffer, L_{OBN} – number of transmitter buffer reads (for master-slave), and L_{POBN} – number of transmitter buffer correct reads (for master-slave).

Based on the above parameters, the following coefficients are calculated:

$$k_P = \frac{L_{TP} - L_{NBN} - L_{NBO}}{L_{TP}} = 1 - \left(\frac{L_{NBN} + L_{NBO}}{L_{TP}} \right), \tag{5.5}$$

or

$$k_P = \frac{L_{POBO}}{L_{TP}}, \tag{5.6}$$

and

$$k_s = \frac{\sum_{i=1}^{i=L_z} k_{pi}}{L_Z}, \tag{5.7}$$

where k_p is the transition coefficient for the system task, k_s is the system transition coefficient, k_{pi} is the transition coefficient for the system task i, and L_z is the number of tasks executed in the system. Moreover, a measuring task excess, a bus excess in the measuring side, a bus excess in the actuator side, and an actuator task excess are calculated as well.

The transition coefficient k_p for the system task is defined as a rate of data loss in the system, as a result of transmitter or receiver buffer overwrite. The system transition coefficient k_s is defined as the mean value of all system tasks coefficients. This coefficient enables us to prove that all system tasks are executed in a required time and can be used to improve or fine tune the designed measurement-control systems. The second group of parameters obtained during the simulation is involved in a different delay calculation, which is presented in Fig. 5.6. In the DMCS simulation model the response time for all tasks (local, measuring and actuating) executed in the nodes is determined.

The next calculated delays deal with system tasks. For a measuring task, which is a part of a system task, delays between the end of two succeeding executions of the task. The next delay taken into account is the delay between the end of two succeeding executions of measuring tasks. Successful measuring task execution means that the measured data written to the transmitter buffer are not lost in the system on its way to the actuator node and are read by the actuator task. Analogous calculations are done for actuator tasks.

The last delay determined by the DMCS simulator is a communication task delay. This delay is measured from the moment that the measuring task (part of a system task) writes data to in transmitter buffer to the moment these data are read from the receiver buffer by the actuator task of the same system task. For each simulated DMCS system task, seven different delays can be determined.

The values of a delay measured during the simulation are stored and then the following histograms are created :

- histogram of node tasks delays (for the measuring and actuator part),
- histogram of system communication tasks delay,
- histogram of system task data fetching,
- histogram of real system task data fetching,
- histogram of real system task data control,
- histogram of system task control.

Based on the obtained delay distributions, it is possible to determine numbers such as a mean value, minimum and maximum values, etc.

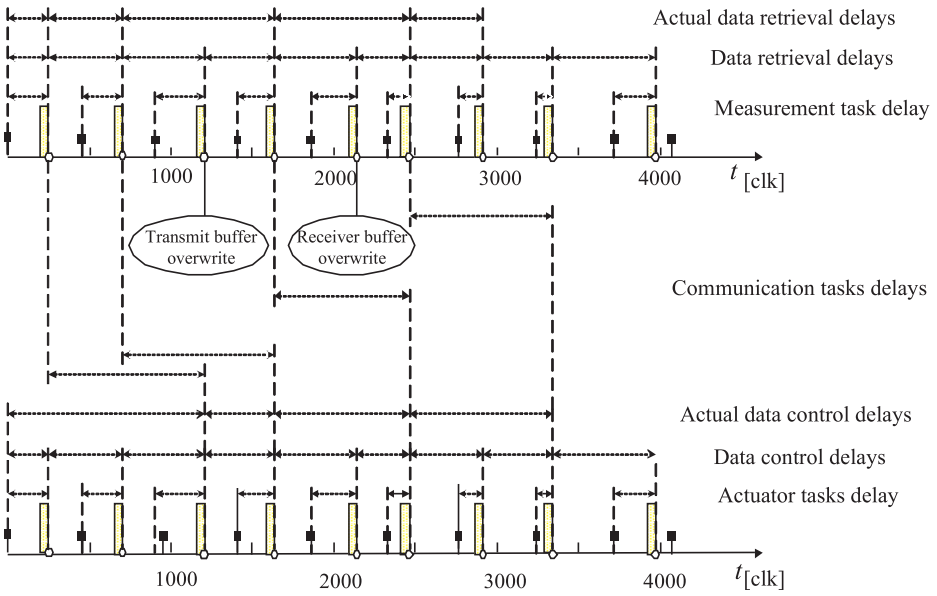


Fig. 5.6. Illustration of delays appearing during a single system task simulation

5.7. Verification of a simulation model

To be sure that the simulation model is correct, an assessment of the worked-out simulation model should be done. The DMCS simulation model, presented in this chapter, was assessed by means of analytical and experimental methods.

5.7.1. Analytical methods

The verification of the correctness of the developed simulation model was done by using the scheduling theory outlined in Section 5.5. From the practical point of view, for the system designer the most important parameter is the task response time. This parameter was used to compare the results obtained during DMCS simulation with those obtained in the analytical and experimental way. As an analytical method, task scheduling theory was used. This method makes it possible, for a single shared resource, to calculate the minimum and maximum of the response time R for each task executed by the processor or the communication system. In the case when system task execution depends on the execution of a few succeeding partial tasks, the minimal value of the system task response time $R_{\min}$ is just a sum of the minimal values of the partial tasks and the maximal value of the response time is a sum of the maximal values of the partial tasks. In the example presented in the chapter, analytical verification of the worked-out simulation model was done for a system structure with sixteen system tasks (see Table 5.1). Each system task contains a measuring and actuator part, which are executed by different nodes.

Table 5.1. System task time parameters

System task no. (priority)	Measuring part of a system task		Actuator part of a system task	
	T [clk]	C [clk]	T [clk]	C [clk]
1..4 (0..3)	1110	130	1110	130
5..8 (4..7)	2110	240	2110	240
9..12 (8..11)	3110	330	3110	330
13..16(12..15)	4110	410	4110	410

Table 5.2. Tasks allocation in measuring nodes

Node 0 C, (*priority*)	Node 1 C, (*priority*)	Node 2 C, (*priority*)	Node 3 C, (*priority*)
4110 (12)	4110 (13)	4110 (14)	4110 (15)
3110 (8)	3110 (9)	3110 (10)	3110 (11)
2110 (4)	2110 (5)	2110 (6)	2110 (7)
1110 (0)	1110 (1)	1110 (2)	1110 (3)

In the presented example, the measuring tasks are placed in four nodes (measuring nodes) and the actuator tasks are placed in eight nodes (actuator nodes). Besides, the system task covering both measuring and actuator nodes runs two local tasks with the following values of the period and executing time: $T_1 = 500$, $C_1 = 50$, $T_2 = 700$ and $C_2 = 70$. The communication part of a simulated DMCS system is running according to the *peer-to-peer* strategy and the realized CAN protocol (128 kbps). Tasks in all nodes are scheduled according to the RM approach without task preempting. The rule of measuring task distribution between four nodes and its priority allocation is the following: the tasks number 1 (priority 0), 5 (4), 9 (8) and 14 (12) are placed in the measuring node 0 and so on (see Table 5.2).

The rule of actuator task distribution between eight nodes and its priority allocation is shown in Table 5.3.

The minimal and maximal response time values for the system task 4, for the DMCS structure presented in the chapter, can be calculated from the following relationship:

$$R_{4\,\min} = R_{Z_4W_3\,\min} + R_{Zk_4\,\min} + R_{Z_4W_5\,\min}, \tag{5.8}$$

where $R_{Z_4W_3\,\min}$ denotes minimal response time of the task 4 in the node 3, $R_{Zk_4\,\min}$ denotes minimal response time of the communication task 4, $R_{Z_4W_5\,\min}$ denotes minimal response time of the task 4 in the node 5, and

$$R_{4\,\max} = R_{Z_4W_3\,\max} + B_{zk_4} + R_{Zk_4\,\max} + R_{Z_4W_5\,\max}, \tag{5.9}$$

where $R_{Z_4W_3\,\max}$ denotes maximal response time of the task 4 in the node 3, $R_{Zk_4\,\max}$ denotes maximal response time of the communication task 4, $R_{Z_4W_5\,\max}$ denotes max-

Table 5.3. Tasks allocation in actuator nodes

Node 4 C, (*priority*)	Node 5 C, (*priority*)	Node 6 C, (*priority*)	Node 7 C, (*priority*)
3110 (8)	3110 (9)	3110 (10)	3110 (11)
1110 (0)	1110 (1)	1110 (2)	1110 (3)
Node 8 C, (*priority*)	Node 9 C, (*priority*)	Node 10 C, (*priority*)	Node 11 C, (*priority*)
4110 (12)	4110 (13)	4110 (14)	4110 (15)
2110 (4)	2110 (5)	2110 (6)	2110 (7)

imal response time of the task 4 in node 5, and B_{zk_4} denotes blocking time of the communication task 4.

The communication task ready for transmission can be blocked by a lower priority task which just occupies the bus. The blocking time for the discussed task 4 is calculated from the following relationship:

$$B_{Zk_4} = \max\{C_{Zk_{16}}; C_{Zk_{12}}; C_{Zk_8}\}, \tag{5.10}$$

where $C_{Zk_{8,12,16}}$ stands for times of the communication tasks.

Response times for each node task are calculated from the recurrence equation (5.4). The measuring task of the system task 4 considered is executed in the node 3 as a task 4. The actuator task of the system task 4 is executed in the node 5 as a task 4. According to the equation (5.8), minimal response time for the system task 4 amounts to

$$R_{4\,\min} = 540 + 128 + 130 = 798.$$

According to the relation (5.9) we receive the following maximal response time for the system task 4:

$$R_{4\,\max} = 780 + 128 + 320 + 380 = 1608.$$

The minimal and maximal response time obtained from the simulation model for the same parameters for the discussed system task is the following: $R'_{4\,\min} = 1370$ and $R'_{4\,\max} = 1490$. All the above time values are expressed in numbers of simulation model clock cycles.

Scheduling theory can be treated as a "worst-case execution time" analysis, therefore results obtained in this way are much more pesimistic, which is apparent if compared with results obtained by simulation and in the analytical way. In both cases, minimal and maximal response times for the system task 4 considered are significantly different. In the chapter there are presented results only for the chosen task, but analysis was done for all system tasks and the achieved results were similar. From the practical point of view more useful are simulation results because in the designed system the "worst-case" assumed by scheduling theory can never appear in a real system.

5.7.2. Experimental approach

To measure the transmitted data delays in a real system, a DMCS physical model was derived. The DMCS block diagram and the picture of the physical model are presented in Fig. 5.7. The DMCS physical model consists of the following four nodes: the measuring node, the actuator node, the simulation node for the communication bus loading simulation and the counter node (Markowski, 2004; 2005).

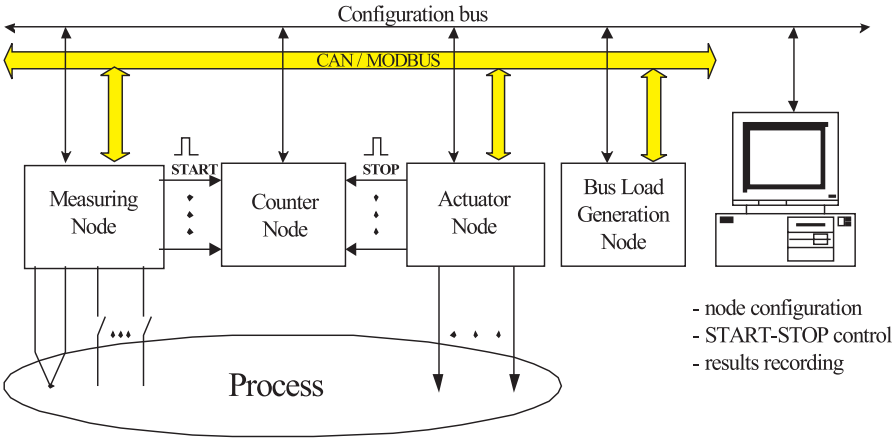


Fig. 5.7. DMCS block schema

The physical model's nodes are configured and managed by dedicated software running on a PC. This PC is used also for gathering the data from all nodes during system operation. Each node is connected to three communication buses: the configuration bus, the master/slave bus, and the peer-to-peer bus. After suitable configuration of nodes, the physical model gives the opportunity to research the delay histograms for a chosen system task. The delay is measured from the appearance of the impulse *start* on the counter input up till the impulse *stop* comes. The *start* signal is set by the measuring node and the *stop* signal is set by the actuator node.

For the configuration of the DMCS physical model presented in the chapter, there were carried out 9000 realizations of the chosen system task. In Fig. 5.8, the delay histogram acquired during physical model operation for the system task with the priority 3 is presented. To compare the same parameters of the obtained histograms, in Tables 5.4 and 5.5 the minimal and maximal values of the searched task delay registered during physical model operation and simulation model operation are written down.

Table 5.4. Delays in the physical model

τ_{min} [ms]	τ_{mean} [ms]	τ_{max} [ms]
6.91	11.93	14.61

Table 5.5. Delays in the simulation model

τ_{min} [ms]	τ_{mean} [ms]	τ_{max} [ms]
6.81	12.17	14.01

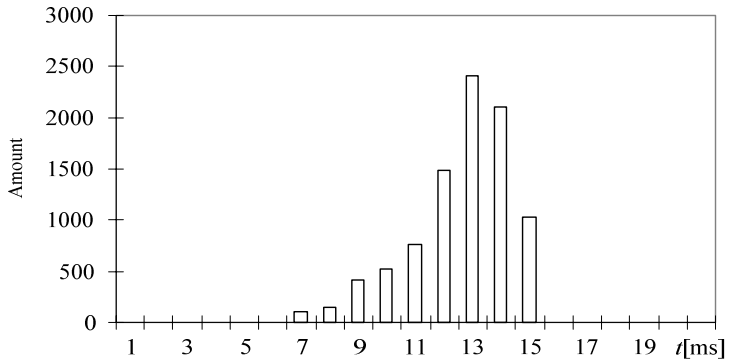


Fig. 5.8. Physical model – transmitted data delay

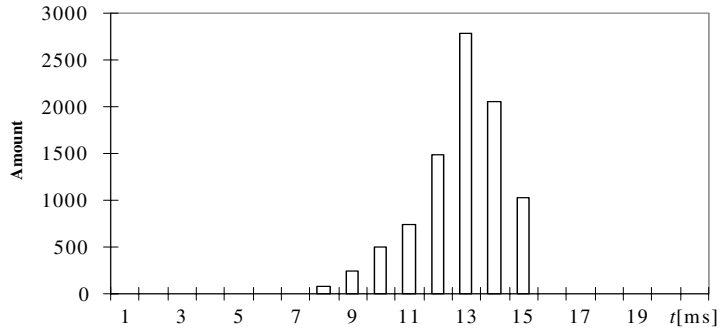


Fig. 5.9. Simulation model – transmitted data delay

In Fig. 5.9 we present the data transmission delay histogram received in the simulation model. In both cases the histograms are almost the same. The histograms in Figs. 5.8 and 5.9 are right side symmetric and single-modal. The histogram obtained from the physical model has a lower minimal value and a greater maximal value than that obtained in simulation model. Differences appear from the following simplifications assumed during the creating of the DMCS simulation model:

- neglecting time for switching tasks executed by the processor in nodes,
- neglecting the program cooperating with the communication processor,
- neglecting delays introduced by the communication processor.

5.8. Simulation of DMCS

The worked out DMCS simulation model is based on the action reviewing and event planning method. This models allows carrying out simulation research of the DMCS in a wide range, e.g. the influence of the number and kind of system tasks on DMCS features or the influence of the system and node structure on DMCS performance. Below, the results of a simulation illustrating the influence of the system and node structure on DMCS time parameters are presented.

5.8.1. Influence of the DMCS and node structure on time system parameters

The main aim of simulation research was to asses the influence of the system and node structure on the designed DMCS time parameters. Simulation for different structures were done for the same number of system tasks. The simulator allows us to trace the loss of transmitted data. After the designed system simulation, we know what data were lost, where it happened and what the reason was. The loss of the data is described by the cross task coefficient (k_p) and the cross system coefficient (k_s), which are defined by the relationships (5.5) and (5.7). If the value of the cross system coefficient is $k_s = 1$, it means that all data acquired by the measuring tasks pass the system communication part, reach their destination in the actuator nodes and these nodes successfully execute the the control task. If the value of the cross system coefficient is $k_s < 1$, it means that the data were lost.

5.8.2. Parameterization of the DMCS system model

The results of simulations presented in this section allowing to asses the influence of the node and DMCS structure on its time parameters were done using the following assumptions:

(i) Number of system tasks: 16.

(ii) Method of node task scheduling: RM without preemption.

(iii) Method of media access control: *peer-to-peer*.

(iv) Number of data bits in each communication task: 16.

(v) Bus utilization coefficient u_m: 0.5 or 1.

(vi) Time of simulation running: 10 millions cycles of the simulation model clock.

System tasks time parameters for which the simulation was carried out are presented in Table 5.1. Each system task contains the measuring and actuator part executed in different nodes. The values of the parameters C and T are expressed in cycles numbers of the simulation model clock. The execution of the each system task involves a communication task which moves data across the bus from a measuring node to an actuator node.

Let us assume that the measuring parts of the system tasks are placed in four nodes according to the strategy "*pp*" but the actuator parts of the system tasks are placed in actuator nodes during succeeding simulations, as is shown in Table 5.6. Additionally, the measuring and actuator nodes are executing two local tasks. Simulations were done for two example transmission speeds on the communication bus, such that the bus utilization coefficient u_m figure out 0.5 or 1. Task grouping in the measuring and actuator nodes was done according to the "*pp*" or "*sp*" strategy. In the case of tasks grouping for four nodes according to to the "pp" strategy, in the first node there were placed tasks with priority 0, 4, 8 and 12. If tasks are groupped for four nodes according to the "sp" strategy, in the first node there are placed tasks with priority 0, 2, 3 and 4. In Fig. 5.10, the structures of the simulation system used during the simulations are shown. The results of simulations presented in this chapter were achieved assuming the simplification that the loading of all simulated nodes is the same (two local tasks and four measuring tasks).

Table 5.6. Actuator task grouping schema in succeeding simulations

Simulation no.	Number of actuator nodes.	Number of tasks in actuator node.	Strategy of task grouping	Bus utilization factor u_m
1	16	1		0.5
2	16	1		1
3	8	2	pp	0.5
4	8	2	pp	1
5	4	4	pp	0.5
6	4	4	pp	1
7	8	2	sp	0.5
8	8	2	sp	1
9	4	4	sp	0.5
10	4	4	sp	1
11	4	4	pp	0.5
12	4	4	pp	1

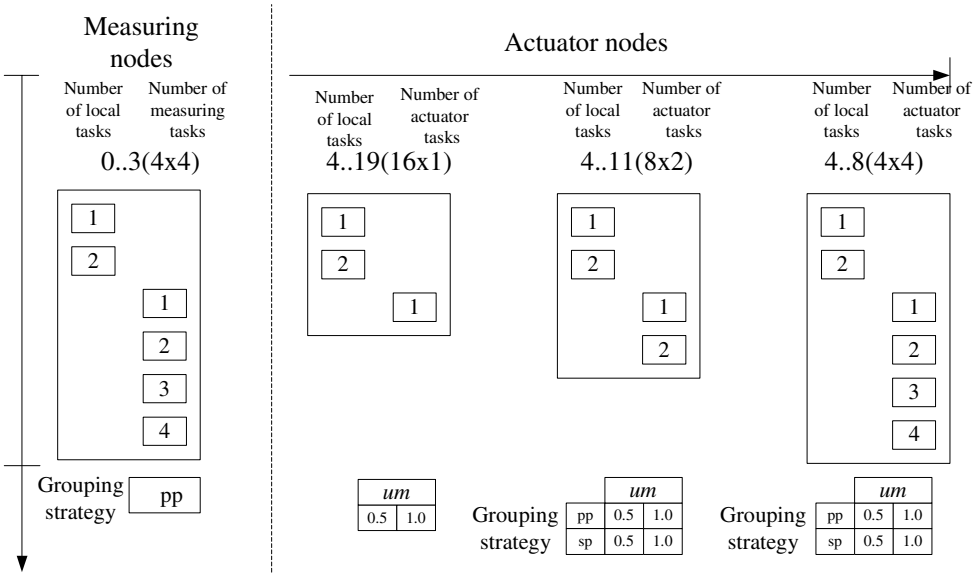


Fig. 5.10. Simulated DMCS structure

Each actuator task has its own buffer. Such a solution leads to a situation where the transmitted measuring data, after they reach actuator node, are additionally processed by the node local task and then the actuator task. Such a node structure modification allows us to find all tasks to be served without loosing the data. In

Table 5.7. System task cross coefficient in succeeding simulations

No of task	Simulation No.											
	1	2	3	4	5	6	7	8	9	10	11	12
	Number of actuator task and grouping strategy											
	16	16	8pp	8pp	4pp	4pp	8sp	8sp	4sp	4sp	4pp	4pp
	Bus utilization coefficient u_m											
	0,5	1	0,5	1	0,5	1	0,5	1	0,5	1	0,5	1
	Cross system task coefficient k_p											
0	1	1	0.57	0.49	0.45	0.03	1	1	1	0.87	1	1
1	1	1	0.64	0.52	0.45	0.16	1	1	1	0.7	1	1
2	1	1	0.57	0.55	0	0.2	1	0.91	1	0.5	1	1
3	1	1	0.53	0.59	0.03	0.21	1	0.52	1	0.34	1	1
4	1	1	0.53	0.57	0.57	0.24	1	1	1	0.96	1	1
5	1	1	0.56	0.61	0.57	0.24	1	1	1	0.68	1	1
6	1	1	0.61	0.61	0.5	0.3	1	0.99	1	0.57	1	1
7	1	1	0.59	0.64	0.03	0.29	1	0.49	1	0.42	1	1
8	1	1	0.85	0.95	0.42	0.63	1	1	1	1	1	1
9	1	1	0.8	0.91	0.42	0.52	1	1	1	0.82	1	1
10	1	1	0.9	0.89	0.48	0.49	1	1	1	0.83	1	1
11	1	1	0.92	0.9	0.55	0.49	1	0.82	1	0.79	1	1
12	1	1	0.89	0.93	0.56	0.73	1	1	1	1	1	1
13	1	1	0.92	0.87	0.56	0.75	1	1	1	1	1	1
14	1	1	0.9	0.83	0.64	0.75	1	1	1	1	1	1
15	1	1	0.9	0.93	0.97	0.76	1	1	1	1	1	1

Table 5.7, the values of the cross system coefficient obtained during the following simulation are presented. In Table 5.7, chosen results of simulations, pointed by numbers from 1 to 10, which were done for an actuator node structure where all actuator tasks read data from the same receiver buffer, which is served according to the FIFO rule, are presented. In the simulations pointed by 11 and 12 there was introduced a change which consists in the fact that in the node structure the local task reads data from the common buffer FIFO and then places these data in the other buffer dedicated for one actuator task.

In Fig. 5.11, the values of the cross system coefficient obtained during the following simulation, which enable quality assessment of the simulated system with a given structure, are presented.

The final results of the cross system coefficient k_s, presented in Fig. 5.11, show that for six system structures numbered as simulations 1, 2, 7, 9, 11 and 12 this coefficient is equal to 1. It means that in these cases all data acquired by measuring

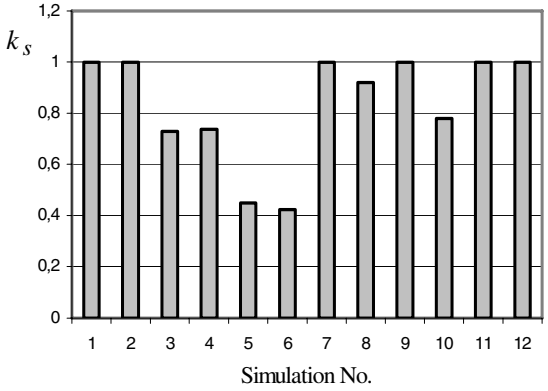


Fig. 5.11. Cross system coefficient

tasks successfully pass their way from the measuring point to the actuator node and this node executes the control task. In the rest of the simulated structures the passing data were lost. To see what the loosing rate is and for which task and which structure it has happened, the cross task coefficient k_p is calculated. The cross task coefficients k_p for the simulations are presented in Table 5.5. The analysis of the simulation results done on the basis of the achieved number of the transmit and receiver FIFO buffers overwrites shows that the loss of data occurred in actuator nodes. In the FIFO buffer, data for tasks with a long period are blocking data for tasks with a shorter period. This leads to buffers fill and same writings done from the bus side can be lost.

The worst simulation results of the cross system coefficient factor k_s were achieved in the simulations no. 5 and 6 (Fig. 5.11) for the node structure with a common buffer. In both cases, in the measuring and actuator nodes there are grouped four tasks which use the same communication buffers. To cope with this problem, a new structure of the actuator node was proposed and the simulations no. 5 and 6 were run once again. In the new structure of the actuator node, additional dedicated buffers for each task were introduced. Simulation results are now better than in the previous case and the cross system coefficient factor k_s is equal to 1, which is shown in the simulations no. 11 and 12 in Fig. 5.11. The structure node modification improves not only the k_s and k_p coefficients but also has influence on the transmitted data delay. The mean values of the data transmission delays for tasks belonging to the task group with a shorter period of task execution are presented in Fig. 5.12(a) and to the task group with a longer period of task execution in Fig. 5.1(b). For the above cases, the bus utilization coefficient had the value $u_m = 0.5$, while the cross system coefficient $k_s = 1$. In the next task realizations, the time from writing data to the transmission buffer in the measuring node to the moment when the data were read from the receiver buffer in the actuator node was measured. Data was written to the transmitter buffer in the end phase of measuring task execution. Data were read from the receiver buffer in the end phase of actuator task execution.

The introduction of dedicated buffers in actuator nodes eliminates the discussed system structure disadvantage but leads to mean delay in data transmission. This is observed particularly in the case of tasks with a short period of execution. For tasks

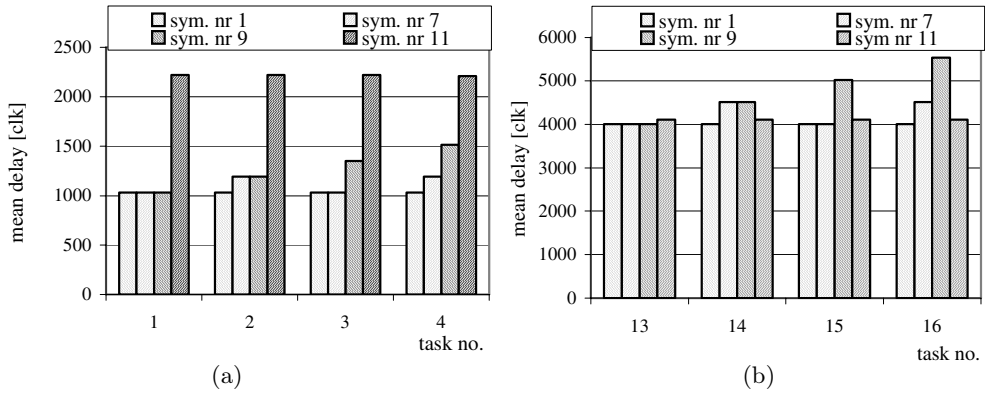


Fig. 5.12. Mean delay in data transmission obtained during the simulations no. 1, 7, 9 and 11 for system tasks with a shorter (a) and a longer (b) period

with a long period of execution, delay introduced by the use of dedicated buffers is considerably smaller.

Research on DMCS time parameters carried out by means of the elaborated simulator shows the usefulness of simulation models during the DMCS with real-time requirements design stage. Such a simulation tool is very useful, especially if we have to make a decision regarding the choice of the right designed system structure. Knowledge about the rate and the losing data place in the designed system is indispensable for system operation assessment. Knowledge of delay values or, better, its distributions allows assesings how the real-time requirements are fulfilled and can be useful for time system parameters tuning. Moreover, such knowledge can be used for metrological assessment of the designed system.

5.9. Summary

In this chapter we have discussed three basic methods used for DMCS time parameters assigning, which can help the DMCS designer to make right decisions concerning the nodes and system structure. The presented, results of our research show that the most useful method appears to be the discrete tasks simulation method based on the activity review method and the event planning method. The easiest way of DMCS analysis is the usage of the analytical approach based on scheduling theory, but the obtained results are pessimistic because of the use of the "worst-case tasks execution time" strategy, which leads to non-optimal solutions.

The most difficult way of DMCS analysis is the usage of the experimental method, because it requires a physical construction of the DMCS. The presented approach, verified in our laboratory, shows that such a solution of the physical DMCS model has a grade of universality and, in an easy way, can be adopted to different DMCS structures. The usefulness of the worked out simulator for DMCS analysis was proved and the obtained results are comparable to those achieved by means of a simulation method.

The use of the simulation method requires the knowledge of many tasks time parameters at the beginning of the simulation process but registered during the simulation: the rate, the place of the possible data loss or the obtained task delay histograms are very important for the DMCS designer. Often the problem is in how to present very complicated matters in a simple way in order to simplify the assessment of designed systems. Two coefficients based on data acquired during DMCS simulation: the system task transition coefficient k_p and the system transition coefficient k_s are presented in the chapter. On one simple diagram with system task transition, even for complex systems, it is possible, in an easy way, to find out if the expected time parameters of a designed system are achieved. When some data can be lost in the transmission or receiver buffer in the nodes, we can see on the diagram both – where it happened and how often.

The solution presented in this chapter was verified for CAN and Modbus networks, but the method of DMCS simulation and the way of obtaining the results and presentation can be used for other fieldbuses.

References

Andersson M., Henriksson D., Cervin A. and Arzen K.E. (2005): *Simulation of wireless networked control systems.* — Proc. 44th IEEE Conf. *Decision and Control and European Control Conference, ECC*, Seville, Spain, CD-ROM.

Audsley N., Burns A., Richardson M., Tindell K. and Wellings A. (1997): *Applying new scheduling theory to static priority pre-emptive scheduling.* — Software Engineering J., Vol. 8, No. 5, pp. 285–292.

Baldwin P., Kohli S., Lee E.A., Liu X. and Zhao Y. (2004): *Modeling of sensor nets in Potolemy II.* — Proc. 3rd Symp. *Information Processing in Sensor Networks*, Berkeley, California, USA, ACM Press, pp. 359–368.

Jakubiec J. and Al-Raimi H. (1999): *Modeling data transmission delays in real-time measurement systems.* — Zeszyty Naukowe Politechniki Śląskiej, Series: Elektryka, No. 165, Gliwice, pp. 109–120, (in Polish).

Kim D. and Lee Y. (2002): *Periodic and aperiodic task scheduling in strongly partitioned integrated real-time systems.* — The Computer J., Vol. 45, No. 4, pp. 395–409.

Kwiecień A. (2000): *Analysis of Information Flow in Industrial Computer Networks.* — Wydawnictwo Pracowni Komputerowej Jacka Skalmierskiego, Gliwice, (in Polish).

Lewis P., Lee N., Welsh M. and Culler D. (2003): *Accurate and scalable simulation of entire TinyOS applications.* — Proc. 1st Conf. *Embedded Networked Sensor Systems*, Los Angeles, USA, pp. 126–137.

Markowski A. (2004): *Assigning transmitted data delays in networked measurement – control systems.* — Pomiary, Automatyka, Kontrola, Nos. 7–8, pp. 95–99, (in Polish).

Markowski A. (2005): *Stand to measuring transmitted data delay in networked measurement – control systems.* — Prace Komisji Metrologii Oddziału PAN w Katowicach, Seria Konferencje, No. 9, Zielona Góra, pp. 297–304, (in Polish).

Michta E. (2000): *Communication Models of Networked Measurement – Control Systems.* — Technical University of Zielona Góra Press, Monograph No. 99, (in Polish).

Michta E. (2002): *Scheduling theory in networked measurement – control systems design*. — Proc. Symp. *Measurement Science in the Information Era, IMECO TC7*, Cracow, Poland, pp. 197–202.

Michta E. (2005): *Scheduling systems*, In: Handbook of Measuring System Design (Syndenham P. and Thorn R., Eds.). — New York: John Wiley & Sons, Ltd.

Sydenham P. and Thorn R. (Eds.) (2005): *Handbook of Measuring System Design*. — New York: John Wiley & Sons.

Żaba S. (2003): *Time analysis of selected fieldbuses*. — Pomiary, Automatyka, Robotyka, No. 6, pp. 12–15, (in Polish).

Chapter 6

SENSOR NETWORK DESIGN FOR IDENTIFICATION OF DISTRIBUTED PARAMETER SYSTEMS

Dariusz UCIŃSKI*, Maciej PATAN*, Bartosz KUCZEWSKI*

6.1. Introduction

6.1.1. Inverse problems for distributed parameter systems

Distributed-Parameter Systems (DPSs) are dynamical systems whose state depends not only on time but also on spatial coordinates. They are frequently encountered in practical engineering problems. Examples of a thermal nature are furnaces for heating metal slabs or heat exchangers; examples of a mechanical nature are large flexible antennas, aircrafts and robot arms; examples of an electrical nature are energy transmission lines.

Appropriate mathematical modelling of DPSs most often yields Partial Differential Equations (PDEs), but descriptions by integral equations or integro-differential equations can sometimes be considered. Clearly, such models involve using very sophisticated mathematical methods, but in recompense for this effort we are in a position to describe the process more accurately and to implement more effective control strategies. Early lumping, which means approximation of a PDE by ordinary differential equations of possibly high order, may completely mask the distributed nature of the system and therefore is not always satisfactory.

For the past forty years DPSs have occupied an important place in control and systems theory. This position has grown in relevance due to the ever-expanding classes of engineering systems which are distributed in nature, and for which estimation and control are desired. DPSs or, more generally, infinite-dimensional systems are now an established area of research with a long list of journal articles, conference proceedings and several textbooks to its credit (Curtain and Zwart, 1995; El Jai and Amouroux, 1990; Emirsajłow, 1991; Grabowski, 1999; Klamka, 1991; Korbicz and Zgurowski,

* Institute of Control and Computation Engineering
e-mails: {d.ucinski, m.patan, b.kuczewski}@issi.uz.zgora.pl

1991; Kowalewski, 2001; Lasiecka and Triggiani, 2000; Malanowski *et al.*, 1996; Omatu and Seinfeld, 1989; Sokołowski and Zolesio, 1992; Zwart and Bontsema, 1997), so the field of potential applications could hardly be considered complete (Banks *et al.*, 1996; Lasiecka, 1998; Uciński and El Jai, 1997; Uciński and El Yacoubi, 1998; 1999).

One of the basic and most important questions in DPSs is parameter estimation, which refers to determination from observed data of unknown parameters in the system model such that the predicted response of the model is close, in some well-defined sense, to the process observations (Omatu and Seinfeld, 1989). The parameter-estimation problem is also referred to as parameter identification or simply the inverse problem (Isakov, 1998). There are many areas of technological importance in which identification problems are of crucial significance. The importance of inverse problems in the petroleum industry, for example, is well documented (Ewing and George, 1984; Korbicz and Zgurowski, 1991). One class of such problems involves determination of the porosity (the ratio of the pore volume to the total volume) and permeability (a parameter measuring the ease with which the fluids flow through the porous medium) of a petroleum reservoir based on field production data. Another class of inverse problems of interest in a variety of areas includes determining the elastic properties of an inhomogeneous medium from observations of reflections of waves travelling through the medium. The literature on the subject of DPS identification is considerable. Kubrusly (1977) and Polis (1982) surveyed the field by systematically classifying the various techniques. A more recent book by Banks and Kunisch (1989) is an attempt to present a thorough and unifying account of a broad class of identification techniques for DPS models, also see (Banks, 1992; Uciński and Korbicz, 1990).

6.1.2. Sensor location for parameter estimation

In order to identify the unknown parameters (in other words, to calibrate the model considered), the system's behaviour or response is observed with the aid of some suitable collection of sensors termed the measurement or observation system. In many industrial processes the nature of state variables does not allow much flexibility as to which they can be measured. For variables which can be measured online, it is usually possible to make the measurements continuously in time. However, it is generally impossible to measure process states over the entire spatial domain. For example, in (Phillipson, 1971), the temperature of molten glass flowing slowly in a forehearth is described by a linear parabolic PDE, whereas the displacements occasioned by dynamic loading on a slender airframe can be described by linear second-order hyperbolic PDEs. In the former example, temperature measurements are available at selected points along the spatial domain (obtained by a pyrometer or some other device), whereas in the latter case strain gauge measurements at selected points on the airframe are reduced to yield the deflection data. In both cases the measurements are incomplete in the sense that the entire spatial profile is not available. Moreover, the measurements are inexact by virtue of inherent errors of measurement associated with transducing elements, and also because of the measurement environment.

The inability to take distributed measurements of process states leads to the question of where to locate sensors so that the information content of the resulting signals with respect to the distributed state and PDE model is as high as possible. This is

an appealing problem since in most applications these locations are not prespecified and therefore provide design parameters. The location of sensors is not necessarily dictated by physical considerations or by intuition and, therefore, some systematic approaches should still be developed in order to reduce the cost of instrumentation and to increase the efficiency of identifiers.

As has already been mentioned, the motivations to study the sensor-location problem stem from practical engineering issues. Optimization of air quality-monitoring networks is among the most interesting ones. As is well known, due to traffic emissions, residential combustion and industry emissions, air pollution has become a big social problem. One of the tasks of environmental protection systems is to provide expected levels of pollutant concentrations. In case smog is expected, a local community can be warned and some measures can be taken to prevent or minimize the release of prescribed substances and to render such substances harmless. But to produce such a forecast, a smog-prediction model is necessary (Holnicki *et al.*, 1986; Sydow *et al.*, 1997; 1998; van Loon, 1994), which is usually chosen in the form of an advection-diffusion PDE. Its calibration requires parameter estimation (e.g., the unknown spatially varying turbulent diffusivity tensor should be identfied based on the measurements from monitoring stations (Omatu and Matumoto, 1991b; 1991a)). Since measurement transducers are usually rather costly and their number is limited, we are inevitably faced with the problem of how to optimize their locations in order to obtain the most precise model. A need for appropriate strategies of optimally allocating monitoring stations is constantly indicated in the works which report the implementations of systems to perform air-quality management (Andó *et al.*, 1999; Müller, 2001; Nychka *et al.*, 1998; Sturm *et al.*, 1994; van Loon, 1995). Of course, some approaches have already been advanced (Fedorov, 1996; Müller, 2001). Due to both the complexity of urban and industrial areas and the influence of meteorological quantities, the suggested techniques are not easy to apply, and further research effort is required.

Another stimulating application concerns groundwater modelling employed in the study of groundwater-resources management, seawater intrusion, aquifer remediation, etc. To build a model for a real groundwater system, some observations of state variables such as the head and concentration are needed. But the cost of experiments is usually very high, which results in many efforts regarding, e.g., optimizing the decisions on the state variables to be observed, the number and location of observation wells and the observation frequency (see (Sun, 1994) and the references given therein). Besides, it is easy to imagine that similar problems appear, e.g., in the recovery of valuable minerals and hydrocarbon from underground permeable reservoirs (Ewing and George, 1984), in gathering measurement data for calibration of mathematical models used in meteorology and oceanography (Bennett, 1992; Daley, 1991; Hogg, 1996; Malanotte-Rizzoli, 1996; Navon, 1997), in automated inspection in static and active environments, or in hazardous environments where trial-and-error sensor planning cannot be used (e.g., in nuclear power plants (Korbicz and Zgurowski, 1991; Korbicz *et al.*, 1993)), or, in recent years, in emerging smart material systems (Banks *et al.*, 1996; Lasiecka, 1998). The interested reader is also referred to the works (Christofides, 2001; Jeremić and Nehorai, 1998; 2000; Nehorai *et al.*, 1995; Porat and Nehorai, 1996) for even more sophisticated settings.

6.1.3. Previous work on optimal sensor location

Over the past years, applications have stimulated laborious research on the development of strategies for efficient sensor placement (for reviews, see the papers (Kubrusly and Malebranche, 1985; van de Wal and de Jager, 2001) and the comprehensive monographs (Uciński, 1999; 2005). Nevertheless, although the need for systematic methods was widely recognized, most techniques communicated by various authors usually rely on exhaustive search over a predefined set of candidates and the combinatorial nature of the design problem is taken into account very occasionally (van de Wal and de Jager, 2001). Needless to say that this approach, which is feasible for a relatively small number of possible locations, soon becomes useless as the number of possible location candidates increases.

Exceptions to this naive approach constitute the works originating in statistical optimum experimental design (Atkinson and Donev, 1992; Fedorov and Hackl, 1997; Pázman, 1986; Pukelsheim, 1993; Uciński and Atkinson, 2004; Uciński and Bogacka, 2005; Walter and Pronzato, 1997) and its extensions to models for dynamic systems, especially in the context of the optimal choice of sampling instants and input signals (Gevers, 2005; Goodwin and Payne, 1977; Hjalmarsson, 2005; Ljung, 1999; Titterington, 1980). In this vein, various computational schemes have been developed to attack directly the original problem or its convenient approximation. The adopted optimization criteria are essentially the same, i.e., various scalar measures of performance based on the Fisher Information Matrix (FIM) associated with the parameters to be identified are maximized. The underlying idea is to express the goodness of parameter estimates in terms of the covariance matrix of the estimates. For sensor-location purposes, one assumes that an unbiased and efficient (or minimum-variance) estimator is employed. This leads to a great simplification since the Cramér-Rao lower bound for the aforementioned covariance matrix is merely the inverse of the FIM, which can be computed with relative ease, even though the exact covariance matrix of a particular estimator is very difficult to obtain.

As regards dynamic DPSs, the first treatment of this type for the sensor-location problem was proposed by Uspenskii and Fedorov (1975), who maximized the D-optimality criterion, being the determinant of the FIM associated with the estimated parameters characterizing the source term in a simple one-dimensional linear diffusion equation. The authors observed that the linear dependence of the observed outputs on these parameters makes it possible to directly apply the machinery of optimum experimental design theory. The delineated approach was extended by Rafajłowicz (1981) to cover a class of DPSs described by linear hyperbolic equations with known eigenfunctions and unknown eigenvalues. The aim was to find conditions for the optimality of the measurement design and the spectral density of the stochastic input. It was indicated that common numerical procedures from classical experimental design for linear regression models could be adopted to find optimal sensor location. Moreover, the demonstrated optimality conditions imply that the optimal input comprises a finite number of sinusoidal signals and that optimal sensor positions are not difficult to find in some cases. A similar problem was studied in (Rafajłowicz, 1983) in a more general framework of DPSs which can be described in terms of Green's functions.

Over the past two decades, this methodology has been substantially refined to extend its applicability. A comprehensive treatment of both theoretical and algorith-

mic aspects of the resulting sensor location strategies is contained in the monograph (Uciński, 2005) and the doctoral dissertations (Kuczewski, 2006; Patan, 2004). The potential of the approach for generalizations was exploited, e.g., by Patan M. and Patan K. (2005), who developed a fault detection scheme for DPSs based on the maximization of the power of a parametric hypothesis test regarding the nominal state of a given DPS. The approach based on maximization of the determinant of the appropriate FIM is by no means restricted to theoretical deliberations and there are examples which do confirm its effectiveness in practical applications. Thus, in (Munack, 1984), a given number of stationary sensors were optimally located using nonlinear programming techniques for a biotechnological system consisting of a bubble column loop fermenter. On the other hand, Sun (1994) advocates using optimum experimental design techniques to solve inverse problems in groundwater modelling. How to monitor the water quality around a landfill place is an example of such a network design. Nonlinear programming techniques are also used there to find numerical approximations to the respective exact solutions.

A similar approach was used in (Kammer, 1990; 1992) for on-orbit modal identification of large space structures. Although the respective models are not PDEs, but their discretized versions obtained through the finite-element method, the proposed solutions can still be of interest owing to the striking similitude of both formulations. A fast and efficient approach was delineated for reducing a relatively large initial candidate sensor-location set to a much smaller optimum set which retains the linear independence of the target modes and does not lead to a substantial deterioration in the accuracy of modal-response estimates, which is quantified by the determinant of the FIM. Some improvements on this approach by incorporating the basic elements of tabu search were proposed by Kincaid and Padula (2002).

A related optimality criterion was given by Point *et al.* (1996), who investigated maximization of the Gram determinant being a measure of the independence of the sensitivity functions evaluated at sensor locations. The authors argue that such a procedure guarantees that the parameters are identifiable and the correlation between the sensor outputs is minimized. The form of the criterion itself resembles the D-optimality criterion, but the counterpart of the FIM takes on much larger dimensions, which suggests that the approach may involve more cumbersome calculations. Nevertheless, the delineated technique was successfully applied to a laboratory-scale, catalytic fixed-bed reactor (Vande Wouwer *et al.*, 1999).

At this juncture, it should be noted that spatial design methods related to the design of monitoring networks are also of great interest to statisticians and a vast amount of literature on the subject already exists (Müller, 2001; Nychka and Saltzman, 1998; Nychka *et al.*, 1998), contributing to the research field of spatial statistics (Cressie, 1993) motivated by practical problems in agriculture, geology, meteorology, environmental sciences and economics. However, the models considered in the statistical literature are quite different from the dynamic models described by PDEs discussed here. Spatiotemporal data are not considered in this context and the main purpose is to model the spatial process by a spatial random field, incorporate prior knowledge and select the best subset of points of a desired cardinality to best represent the field in question. The motivation is a need to interpolate the observed behaviour of a process at unobserved spatial locations, as well as to design a network of op-

timal observation locations which allows an accurate representation of the process. The field itself is modelled by some multivariate distribution, usually Gaussian (Armstrong, 1998). Designs for spatial trend and variogram estimation can be considered. The basic theory of optimal design for spatial random fields is outlined in the excellent monograph by Müller (2001), which bridges the gap between spatial statistics and classical optimum experimental design theory. The optimal design problem can also be formulated in terms of information-based criteria whose application amounts to maximizing the amount of information (of the Kullback-Leibler type) to be gained from an experiment (Caselton and Zidek, 1984; Caselton *et al.*, 1992). However, the applicability of all those fine statistical results in the engineering context discussed here is not clear for now and more detailed research into this direction should be pursued in the near future (specifically, generalizations regarding time dynamics are not obvious).

Let us remark that an appealing alternative to stationary sensors is to apply spatially movable ones, which leads to the so-called continuous scanning observations. The complexity of the resulting optimization problem is compensated by a number of benefits. Specifically, sensors are not assigned to fixed positions which are optimal only on the average, but are capable of tracking points which provide at a given time instant the best information about the parameters to be identified. Consequently, by actively reconfiguring a sensor system we can expect the minimal value of an adopted design criterion to be lower than the one for the stationary case. What is more, technological advances in communication systems and the growing ease in making small, low power and inexpensive mobile systems now make it feasible to deploy a group of networked vehicles in a number of environments (Cassandras and Li, 2005; Chong and Kumar, 2003; Martínez and Bullo, 2006; Ögren *et al.*, 2004; Sinopoli *et al.*, 2003). In the seminal article (Rafajłowicz, 1986), the D-optimality criterion is considered and an optimal time-dependent measure is sought, rather than the trajectories themselves. On the other hand, Uciński (2000; 2005; Uciński and Korbicz, 2001), apart from generalizations of Rafajłowicz's results, develops some computational algorithms based on the FIM. He reduces the problem to a state-constrained optimal-control one for which solutions are obtained via the methods of successive linearizations, and which is capable of handling various constraints imposed on sensor motions. In turn, the work (Uciński and Chen, 2005) was intended as an attempt to properly formulate and solve the time-optimal problem for moving sensors which observe the state of a DPS so as to estimate some of its parameters. Recently, Patan (2006) demonstrated an efficient algorithm to solve the scanning sensor scheduling problem using the control parameterization-enhancing technique (Lee *et al.*, 1999; 2001).

6.1.4. Our results

In this chapter we shall outline a practical approach to the sensor location problem for parameter estimation which, while being independent of a particular model of the dynamic DPS in question, is versatile enough to cope with practical monitoring networks consisting of many sensors. Owing to volume limitations, we are going to primarily focus on the following problem: we consider N possible sites at which to locate a sensor, but restrictions on the number of sensors at our disposal allow only n of them to be selected. Consequently, the problem is to divide the N available

sites between n gauged sites and the remaining $N - n$ ungauged sites so as to maximize the determinant of the FIM associated with the parameters to be estimated. Since selecting the best subset of sites to locate the sensors constitutes an inherently discrete large-scale resource allocation problem whose solution may be prohibitively time-consuming, an efficient guided search algorithm based on the branch-and-bound method is first developed, which implicitly enumerates all the feasible sensor configurations, using relaxed optimization problems that involve no integer constraints. Although branch-and-bound constitutes one of the most frequent approaches to solve discrete optimization problems and it has indeed been used in the context of network design, cf., e.g., (Boer *et al.*, 2001), the originality here consists in the development of a simple, yet powerful, computational scheme to obtain upper bounds to the optimal values of the D-optimality criterion for the restricted problems. These bounds are obtained by relaxing the 0–1 constraints on the design variables, thereby allowing them to take any value in the interval $[0, 1]$ and resulting in a concave problem of determinant maximization over the set of all linear combinations of a finite number of nonnegative definite matrices, subject to additional linear constraints on the coefficients of those combinations. Then a simplicial decomposition algorithm is proposed for its solution with the restricted master problem reduced to solving an uncomplicated multiplicative weight optimization algorithm. The resulting procedure is guaranteed to produce iterates converging to the solution of the relaxed restricted problem. The detailed proofs of all mathematical results will appear in the forthcoming publication (Uciński and Patan, 2007).

As an alternative approach which is suitable for situations when the numbers of candidate and gauged sites are rather large, we present a method whose idea is to operate on sensor densities per unit area instead of the individual sensor positions. This convenient reformulation makes it possible to apply some powerful tools of convex analysis and derive elegant characterizations of optimal solutions. Apart from that, an extremely simple exchange algorithm is exposed to find the best sensor configurations.

To illustrate the use of both algorithms, we report some numerical experience on a sensor network design problem regarding a two-dimensional convective diffusion process.

The chapter is structured as follows: Section 6.2 states formally the sensor network design problem as a discrete resource allocation problem. The branch-and-bound algorithm for its solution is discussed in Section 6.3. Section 6.4 develops the clusterization-free approach to determine optimal sensor configurations. In Section 6.5, we report the numerical results obtained by applying the algorithms described in Sections 6.3 and 6.4 on an optimal design application. We conclude in Section 6.6 with some comments and conclusions.

6.1.5. Notation

Our notation is more or less standard. Given a set H, $|H|$ and $\bar{H}$ signify its cardinality and closure, respectively. We use $\mathbb{R}$ to denote the set of real numbers and $\mathbb{R}_+$ to denote the set of nonnegative real numbers. The n-dimensional Euclidean vector space is denoted by $\mathbb{R}^n$, and the Euclidean matrix space of real matrices with n rows and k columns is denoted by $\mathbb{R}^{m\times k}$. We will write $\mathbb{S}^n$ for the subspace of $\mathbb{R}^{n\times n}$ consisting of all symmetric matrices. The identity matrix of order n is denoted by E_n. In $\mathbb{S}^n$,

two sets are of special importance: the cone of nonnegative definite matrices and the cone of positive definite matrices, denoted by $\mathbb{S}^n_+$ and $\mathbb{S}^n_{++}$, respectively. The curly inequality symbol $\succeq$ and its strict form $\succ$ are used to denote the Loewner partial ordering of symmetric matrices: For $A, B \in \mathbb{S}^n$, we have

$$A \succeq B \iff A - B \in \mathbb{S}^n_+,$$
$$A \succ B \iff A - B \in \mathbb{S}^n_{++}.$$

We call a point of the form $\alpha_1 u_1 + \cdots + \alpha_\ell u_\ell$, where $\alpha_1 + \cdots + \alpha_\ell = 1$ and $\alpha_i \geq 0$, $i = 1, \ldots, \ell$, a convex combination of the points $u_1, \ldots, u_\ell$ (it can be thought of as a mixture or a weighted average of the points, with α_i being the fraction of u_i in the mixture). Given a set of points U, $\mathrm{co}(U)$ stands for its convex hull, i.e., the set of all convex combinations of the elements of U,

$$\mathrm{co}(U) = \left\{ \sum_{i=1}^{\ell} \alpha_i u_i \,\middle|\, u_i \in U,\ \alpha_i \geq 0,\ i = 1, \ldots, \ell;\ \sum_{i=1}^{\ell} \alpha_i = 1,\ \ell = 1, 2, 3, \ldots \right\}.$$

The probability (or canonical) simplex in $\mathbb{R}^n$ is defined as

$$P_n = \mathrm{co}\big(\{e_1, \ldots, e_n\}\big) = \left\{ p \in \mathbb{R}^n_+ \,\middle|\, \sum_{i=1}^{n} p_i = 1 \right\},$$

where e_j is the usual unit vector along the j-th coordinate of $\mathbb{R}^n$.

6.2. Sensor location problem in question

Let us consider a bounded spatial domain $\Omega \subset \mathbb{R}^d$ with a sufficiently smooth boundary Γ, a bounded time interval $T = (0, t_f]$, and a distributed parameter system whose scalar state at a spatial point $x \in \bar{\Omega} \subset \mathbb{R}^d$ and a time instant $t \in \bar{T}$ is denoted by $y(x, t)$. Mathematically, the system state is governed by the partial differential equation

$$\frac{\partial y}{\partial t} = \mathcal{F}\big(x, t, y, \theta\big) \quad \text{in } \Omega \times T, \tag{6.1}$$

where $\mathcal{F}$ is a well-posed, possibly nonlinear, differential operator which involves first- and second-order spatial derivatives and may include terms accounting for forcing inputs specified *a priori*. The PDE (6.1) is accompanied by the appropriate boundary and initial conditions

$$\mathcal{B}(x, t, y, \theta) = 0 \quad \text{on } \Gamma \times T, \tag{6.2}$$

$$y = y_0 \quad \text{in } \Omega \times \{t = 0\}, \tag{6.3}$$

respectively, $\mathcal{B}$ being an operator acting on the boundary Γ and $y_0 = y_0(x)$ a given function. Conditions (6.2) and (6.3) complement (6.1) such that the existence of a sufficiently smooth and unique solution is guaranteed. We assume that the forms of $\mathcal{F}$ and $\mathcal{B}$ are given explicitly up to an m-dimensional vector of unknown constant parameters θ which must be estimated using observations of the system. The implicit

dependence of the state y on the parameter vector θ will be be reflected by the notation $y(x,t;\theta)$.

In what follows, we consider the discrete-continuous observations provided by n stationary pointwise sensors, namely,

$$z_{\mathrm{m}}^{\ell}(t) = y(x^{\ell},t;\theta) + \varepsilon(x^{\ell},t), \quad t \in T, \tag{6.4}$$

where $z_{\mathrm{m}}^{\ell}(t)$ is the scalar output and $x^{\ell} \in X$ stands for the location of the ℓ-th sensor ($\ell = 1,\ldots,n$), X signifies the part of the spatial domain Ω where the measurements can be made and $\varepsilon(x^{\ell},t)$ denotes the measurement noise. This relatively simple conceptual framework involves no loss of generality since it can be easily generalized to incorporate, e.g., multiresponse systems or inaccessibility of state measurements, cf. (Uciński, 2005, p. 95).

It is customary to assume that the measurement noise is zero-mean, Gaussian, spatial uncorrelated and white (Amouroux and Babary, 1988; Omatu and Seinfeld, 1989; Quereshi *et al.*, 1980), i.e.,

$$\mathrm{E}\big\{\varepsilon(x^{\ell},t)\varepsilon(x^{\ell'},t')\big\} = \sigma^2\delta_{\ell\ell'}\delta(t-t'), \tag{6.5}$$

where σ^2 defines the intensity of the noise, δ_{ij} and $\delta(\,\cdot\,)$ standing for the Kronecker and Dirac delta functions, respectively. Although white noise is a physically impossible process, it constitutes a reasonable approximation to a disturbance whose adjacent samples are uncorrelated at all time instants for which the time increment exceeds some value which is small compared with the time constants of the DPS. A rigorous formulation for a time-correlated setting (cf. Appendix C1 of (Uciński, 2005)) is well beyond the mathematical framework of this paper, but the attendant difficulties are mainly technical and do not substantially affect the basic results to be obtained. What is more, the white-noise assumption is consistent with most of the literature on the subject.

The most widely used formulation of the parameter estimation problem is as follows: Given the model (6.1)–(6.3) and the outcomes of the measurements $z_{\mathrm{m}}^{\ell}(\,\cdot\,)$, $\ell = 1,\ldots,n$, estimate θ by $\widehat{\theta}$, a global minimizer of the output least-squares error criterion

$$\mathcal{J}(\vartheta) = \sum_{\ell=1}^{n} \int_{T} \big\{z_{\mathrm{m}}^{\ell}(t) - y(x^{\ell},t;\vartheta)\big\}^2 \,\mathrm{d}t, \tag{6.6}$$

where $y(\,\cdot\,,\,\cdot\,;\vartheta)$ denotes the solution to (6.1)–(6.3) for a given value of the parameter vector ϑ. In practice, a regularized version of the above problem is often considered by adding to $\mathcal{J}(\vartheta)$ a term imposing stability or *apriori* information or both (Banks and Kunisch, 1989; Vogel, 2002).

Inevitably, the covariance matrix $\mathrm{cov}(\widehat{\theta})$ of the above least-squares estimator depends on the sensor locations x^{ℓ}. This fact suggests that we may attempt to select them so as to yield the best estimates of the system parameters. To form a basis for the comparison of different locations, a quantitative measure of the 'goodness' of particular sensor configurations is required. Such a measure is customarily based on the concept of the *Fisher information matrix*, which is widely used in optimum experimental design theory for lumped systems (Atkinson and Donev, 1992; Fedorov

and Hackl, 1997; Pázman, 1986; Pukelsheim, 1993; Walter and Pronzato, 1997). In our setting, the FIM is given by (Quereshi *et al.*, 1980):

$$M(x^1,\dots,x^n)=\frac{1}{n}\sum_{\ell=1}^{n}\frac{1}{t_f}\int_T g(x^\ell,t)g^{\mathsf{T}}(x^\ell,t)\,\mathrm{d}t, \tag{6.7}$$

where

$$g(x,t)=\left[\frac{\partial y(x,t;\vartheta)}{\partial\vartheta_1},\quad\dots,\quad\frac{\partial y(x,t;\vartheta)}{\partial\vartheta_m}\right]^{\mathsf{T}}_{\vartheta=\theta^0} \tag{6.8}$$

stands for the so-called *sensitivity vector*, θ^0 being a prior estimate to the unknown parameter vector θ (Uciński, 2005; Rafajłowicz, 1981; 1983; Sun, 1994). The rationale behind this choice is the fact that, up to a constant scalar multiplier, the inverse of the FIM constitutes a good approximation of $\mathrm{cov}(\widehat{\theta})$ provided that the time horizon is large, the nonlinearity of the model with respect to its parameters is mild, and the measurement errors are independently distributed and have small magnitudes (Fedorov and Hackl, 1997; Walter and Pronzato, 1997).

As for a specific form of Ψ, various options exist (Atkinson and Donev, 1992; Fedorov and Hackl, 1997; Walter and Pronzato, 1997), but the most popular criterion, called the D-optimality criterion, is the log-determinant of the FIM:

$$\Psi(M)=\log\det(M). \tag{6.9}$$

The resulting D-optimum sensor configuration leads to the minimum volume of the uncertainty ellipsoid for the estimates.

The introduction of an optimality criterion renders it possible to formulate the sensor location problem as maximization of the performance measure

$$\mathcal{R}(x^1,\dots,x^n):=\Psi\big[M(x^1,\dots,x^n)\big] \tag{6.10}$$

with respect to x^ℓ, $\ell=1,\dots,n$ belonging to the admissible set X. This apparently simple formulation may lead to the conclusion that the only question remaining is that of selecting an appropriate solver from a library of numerical optimization routines. Unfortunately, an in-depth analysis reveals complications which accompany this way of thinking.

A key difficulty in developing successful numerical techniques for sensor location is that the number of sensors to be placed in a given region may be quite large. For example, in the research carried out to find spatial predictions for ozone in the Great Lakes of the United States, measurements made by approximately 160 monitoring stations were used (Nychka and Saltzman, 1998). When trying to treat the task as a constrained nonlinear programming problem, the actual number of variables is even doubled, since the position of each sensor is determined by its two spatial coordinates, so that the resulting problem is rather of a large scale. What is more, a desired global extremum is usually hidden among many poorer local extrema. Consequently, to directly find a numerical solution may be extremely difficult. Additionally, a technical complication might also be sensor clusterization, which constitutes a price to pay for the simplifying assumption that the measurement noise is spatially uncorrelated. This means that in an optimal solution different sensors often tend to take measurements at one point, and this is acceptable in applications rather occasionally.

In the literature, a common remedy for the last predicament is to guess *a priori* a set of N possible candidate locations, where $N > n$, and then to seek the best subset of n locations from among the N possible ones, so that the problem is then reduced to a combinatorial one. In other words, the problem is to divide the N available sites between n gauged sites and the remaining $N - n$ ungauged sites so as to maximize the determinant of the FIM associated with the parameters to be estimated. This formulation will be also adopted here.

Specifically, let x^i, $i = 1, \ldots, N$ denote the positions of sites where sensors can potentially be placed. Now that our design criterion has been established, the problem is to find an optimal allocation of n available sensors to x^i, $i = 1, \ldots, N$ so as to maximize the value of the design criterion incurred by the allocation. In order to formulate this mathematically, introduce for each possible location x^i a variable v_i which takes the value 1 or 0 depending on whether a sensor is or is not located at x^i, respectively. The FIM in (6.7) can then be rewritten as

$$M(v_1, \ldots, v_N) = \sum_{i=1}^{N} v_i M_i, \tag{6.11}$$

where

$$M_i = \frac{1}{nt_f} \int_T g(x^i, t) g^{\mathsf{T}}(x^i, t)\, \mathrm{d}t. \tag{6.12}$$

It is straightforward to verify that the $m \times m$ matrices M_i are nonnegative definite and, therefore, so is $M(v_1, \ldots, v_N)$.

Then our design problem takes the form:

Problem P: Find the sequence $v = (v_1, \ldots, v_N)$ to maximize

$$\mathcal{P}(v) = \log\det\big(M(v)\big) \tag{6.13}$$

subject to the constraints

$$\sum_{i=1}^{N} v_i = n, \tag{6.14}$$

$$v_i = 0 \text{ or } 1, \quad i = 1, \ldots, N. \tag{6.15}$$

This constitutes a 0–1 integer programming problem which necessitates an ingenious solution. In what follows, we propose to solve it using the branch-and-bound method, which is a standard technique for solving integer-programming problems.

6.3. Exact solution by branch-and-bound

6.3.1. Outline

Branch-and-Bound (BB) constitutes a general algorithmic technique for finding optimal solutions of various optimization problems, especially discrete or combinatorial (Bertsekas, 1999; Floudas, 2001). If applied carefully, it can lead to algorithms that run reasonably fast on average.

Principally, the BB method is a tree-search algorithm combined with a rule for pruning subtrees. Suppose we wish to maximize an objective function $\mathcal{P}(v)$ over a finite set V of admissible values of the argument v called the feasible region. BB then progresses by iteratively applying two procedures: branching and bounding. *Branching* starts with smartly covering the feasible region by two or more smaller feasible subregions (ideally, partitioning into disjoint subregions). It is then repeated recursively to each of the subregions until no more division is possible, which leads to a progressively finer partition of V. The consecutively produced subregions naturally generate a tree structure called the BB tree. Its nodes correspond to the constructed subregions, with the feasible set V as the root node and the singleton solutions $\{v\}$, $v \in V$ as terminal nodes. In turn, the core of *bounding* is a fast method of finding upper and lower bounds to the maximum value of the objective function over a feasible subdomain. The idea is to use these bounds to economize computation by eliminating nodes of the BB tree that have no chance of containing an optimal solution. If the upper bound for a subregion V_A from the search tree is lower than the lower bound for any other (previously examined) subregion V_B, then V_A and all its descendant nodes may be safely discarded from the search. This step, termed *pruning*, is usually implemented by maintaining a global variable that records the maximum lower bound encountered among all subregions examined so far. Any node whose upper bound is lower than this value need not be considered further and thereby can be eliminated. It may happen that the lower bound for a node matches its upper bound. That value is then the maximum of the function within the corresponding subregion and the node is said to be solved. The search proceeds until all nodes have been solved or pruned, or until some specified threshold is met between the best solution found and the upper bounds on all unsolved problems.

In what follows, we will use the symbol I to denote the index set $\{1, \ldots, N\}$ of possible sensor locations. Our implementation of BB for Problem P involves the partition of the feasible set

$$V = \left\{ (v_1, \ldots, v_N) \,\middle|\, \sum_{i=1}^{N} v_i = n,\ v_i = 0 \text{ or } 1,\ \forall i \in I \right\} \tag{6.16}$$

into subsets. It is customary to select subsets of the form (Bertsekas, 1999):

$$V(I_0, I_1) = \{ v \in V \mid v_i = 0,\ \forall i \in I_0,\ v_i = 1,\ \forall i \in I_1 \}, \tag{6.17}$$

where I_0 and I_1 are disjoint subsets of I. Consequently, $V(I_0, I_1)$ is the subset of V such that a sensor is placed at the locations with indices in I_1, no sensor is placed at the locations with indices in I_0, and a sensor may or may not be placed at the remaining locations.

Each subset $V(I_0, I_1)$ is identified with a node in the BB tree. The key assumption in the BB method is that for every nonterminal node $V(I_0, I_1)$, i.e., the node for which $I_0 \cup I_1 \neq I$, there is an algorithm that determines an upper bound $\bar{\mathcal{P}}(I_0, I_1)$ to the maximum design criterion over $V(I_0, I_1)$, i.e.,

$$\bar{\mathcal{P}}(I_0, I_1) \geq \max_{v \in V(I_0, I_1)} \mathcal{P}(v), \tag{6.18}$$

and a feasible solution $\underline{v} \in V$ for which $\mathcal{P}(\underline{v})$ can serve as a lower bound to the maximum design criterion over V. We may compute $\bar{\mathcal{P}}(I_0, I_1)$ by solving the following relaxed problem:

Problem $\mathbf{R}(I_0, I_1)$: Find the sequence $\bar{v}$ to maximize (6.13) subject to the constraints

$$\sum_{i=1}^{N} v_i = n, \tag{6.19}$$

$$v_i = 0, \quad i \in I_0, \tag{6.20}$$

$$v_i = 1, \quad i \in I_1, \tag{6.21}$$

$$0 \leq v_i \leq 1, \quad i \in I \setminus (I_0 \cup I_1). \tag{6.22}$$

In Problem $\mathrm{R}(I_0, I_1)$, all 0–1 constraints on the variables v_i are relaxed by allowing them to take any value in the interval $[0, 1]$, except that the variables v_i, $i \in I_0 \cup I_1$ are fixed at either 0 or 1. A simple and efficient method of solving it is given in Section 6.3.3. As a result of its application, we set $\bar{\mathcal{P}}(I_0, I_1) = \mathcal{P}(\bar{v})$.

As for $\underline{v}$, we can specify it as the best feasible solution (i.e., an element of V) found so far. If no solution has been found yet, we can either set the lower bound to $-\infty$, or use an initial guess about the optimal solution (experience provides evidence that the latter choice leads to much more rapid convergence).

6.3.2. Branching rule

The result of solving Problem $\mathrm{R}(I_0, I_1)$ can serve as a basis to construct a branching rule for the binary BB tree. We adopt here the approach in which the node/subset $V(I_0, I_1)$ is expanded (i.e., partitioned) by first picking out all fractional values from among the values of the relaxed variables, and then rounding to 0 and 1 a value which is the most distant from both 0 and 1. Specifically, we apply the following steps:

(i) Determine

$$i_\star = \arg \min_{i \in I \setminus (I_0 \cup I_1)} |v_i - 0.5|. \tag{6.23}$$

(In case there are several minimizers, randomly pick one of them.)

(ii) Partition $V(I_0, I_1)$ into $V(I_0 \cup \{i_\star\}, I_1)$ and $V(I_0, I_1 \cup \{i_\star\})$ whereby two descendants of the node in question are defined.

A recursive application of the branching rule starts from the root of the BB tree, which corresponds to the trivial subset $V(\emptyset, \emptyset) = V$ and the fully relaxed problem. Each node of the BB tree corresponds to a continuous relaxed problem, $\mathrm{R}(I_0, I_1)$, while each edge corresponds to fixing one relaxed variable at 0 or 1.

The above scheme has to be complemented with a search strategy to incrementally explore all the nodes of the BB tree. Here we use a common depth-first technique (Reinefeld, 2001; Russell and Norvig, 2003) which always expands the deepest node in the current fringe of the search tree. The reason behind this decision is that the search proceeds immediately to the deepest level of the search tree, where the nodes

Algorithm 1. Recursive version of the depth-first branch-and-bound method. It uses two global variables, *LOWER* and *v_ best*, which are respectively the maximal value of the FIM determinant over the feasible solutions found so far and the solution at which it is attained.

```
1: procedure RECURSIVE-DFBB(I_0, I_1)
2:     if |I_0| > N − n or |I_1| > n then
3:         return                          ▷ Constraint (6.19) would be violated
4:     end if
5:     if SINGULARITY-TEST(I_0, I_1) then
6:         return                          ▷ Only zero determinants can be expected
7:     end if
8:     v_relaxed ← RELAXED-SOLUTION(I_0, I_1)
9:     det_relaxed ← DET-FIM(v_relaxed)
10:    if det_relaxed ≤ LOWER then
11:        return                                                  ▷ Pruning
12:    else if INTEGRAL-TEST(v_relaxed) then
13:        v_best ← v_relaxed
14:        LOWER ← det_relaxed
15:        return                                  ▷ Relaxed solution is integral
16:    else
17:        i_⋆ ← INDEX-BRANCH(v_relaxed)           ▷ Partition into two descendants
18:        RECURSIVE-DFBB(I_0 ∪ {i_⋆}, I_1)
19:        RECURSIVE-DFBB(I_0, I_1 ∪ {i_⋆})
20:    end if
21: end procedure
```

have no successors (Gerdts, 2005). In this way, lower bounds on the optimal solution can be found or improved as fast as possible.

A recursive version of the resulting depth-first branch-and-bound is implemented in Algorithm 1. The operators involved in this implementation are as follows:

- SINGULARITY-TEST(I_0, I_1) returns true only if the expansion of the current node will result in a singular FIM, see Section 6.3.3 for details.
- RELAXED-SOLUTION(I_0, I_1) returns a solution to Problem R(I_0, I_1).
- DET-FIM(v) returns the log-determinant of the FIM corresponding to v.
- INTEGRAL-TEST(v) returns true only if the current solution v is integral.
- INDEX-BRANCH(v) returns the index defined by (6.23).

6.3.3. Solving the relaxed problem via simplicial decomposition

Optimality conditions. The non-leaf nodes of the BB tree are processed by relaxing the original combinatorial problem, which directly leads to Problem R(I_0, I_1). This section provides a detailed exposition of a simplicial decomposition method which is particularly suited for its solution.

For notational convenience, we replace the variables v_i, $i \in I \setminus (I_0 \cup I_1)$ by w_j, $j = 1, \ldots, q$, where $q = |I \setminus (I_0 \cup I_1)|$, since there exists a bijection π from $\{1, \ldots, q\}$ to $I \setminus (I_0 \cup I_1)$ such that $w_j = v_{\pi(j)}$, $j = 1, \ldots, q$. Consequently, we obtain the following formulation:

Problem $\mathbf{R'(I_0, I_1)}$: Find $w \in \mathbb{R}^q$ to maximize

$$\mathcal{Q}(w) = \log\det\big(G(w)\big) \tag{6.24}$$

subject to the constraints

$$\sum_{j=1}^{q} w_j = r, \tag{6.25}$$

$$0 \le w_j \le 1, \quad j = 1, \ldots, q, \tag{6.26}$$

where

$$r = n - |I_1|, \quad G(w) = A + \sum_{j=1}^{q} w_j S_j, \quad A = \sum_{i \in I_1} M_i, \quad S_j = M_{\pi(j)}, \quad j = 1, \ldots, q. \tag{6.27}$$

(Note that the $|I_1|$ sensors have already been assigned to the locations x^i, $i \in I_1$, and thus a decision about the placement of r remaining sensors has to be made.)

In the sequel, W will stand for the set of all vectors $w = (w_1, \ldots, w_q)$ satisfying (6.25) and (6.26). Note that it forms a polygon in $\mathbb{R}^q$. Recall that the log-determinant is concave and strictly concave over the cones $\mathbb{S}^m_+$ and $\mathbb{S}^m_{++}$, respectively, cf. (Boyd and Vandenberghe, 2004; Pukelsheim, 1993). Thus the objective function (6.24) is concave as the composition of the log-determinant with an affine mapping, see (Boyd and Vandenberghe, 2004, p. 79). We wish to maximize it over the polyhedral set W. If the FIM corresponding to an optimal solution $w^\star$ is nonsingular, then an intriguing form of the optimality conditions can be derived, cf. (Uciński and Patan, 2007).

Proposition 6.1. *Suppose that the matrix $G(w^\star)$ is nonsingular for some $w^\star \in W$. The vector $w^\star$ constitutes a global solution to Problem $R'(I_0, I_1)$ if, and only if, there exists a number $\lambda^\star$ such that*

$$\varphi(j, w^\star) \begin{cases} \ge \lambda^\star & \text{if } \ w_j^\star = 1, \\ = \lambda^\star & \text{if } \ 0 < w_j^\star < 1, \\ \le \lambda^\star & \text{if } \ w_j^\star = 0, \end{cases} \tag{6.28}$$

where

$$\varphi(j, w) = \operatorname{trace}\big[G^{-1}(w) S_j\big], \quad j = 1, \ldots, q. \tag{6.29}$$

Proposition 6.1 reveals one characteristic feature of the optimal solutions, namely, that when identifying them the function φ turns out to be crucial and optimality means separability of the components of $w^\star$ in terms of the values of this function. Specifically, the values of $\varphi(\,\cdot\,, w^\star)$ for the indices corresponding to the fractional components of $w^\star$ must be equal to some constant $\lambda^\star$, whereas for the components taking the value 0 or the value 1 the values of $\varphi(\,\cdot\,, w^\star)$ must be no larger and no smaller than $\lambda^\star$, respectively.

Singular information matrices. Note that the assumption that $G(w)$ is nonsingular can be dropped, since there is a very simple method to check whether or not the current relaxed problem will lead to an FIM which is nonsingular.

Proposition 6.2. *The FIM corresponding to the solution to Problem* $R'(I_0, I_1)$ *is singular if and only if so is* $G(\bar{w})$*, where*

$$\bar{w} = \underbrace{(r/q, \ldots, r/q)}_{q \ times}. \tag{6.30}$$

Consequently, a test of the singularity of the matrix $G(\bar{w}) = A + \frac{r}{q}\sum_{j=1}^{q} S_j$ can be built into the BB procedure in order to drop the corresponding node from further consideration and forego the examination of its descendants. Otherwise, the vector $(r/q, \ldots, r/q)$ may serve as a good starting point for the simplicial decomposition algorithm outlined in what follows.

Remark 6.1. *A solution to Problem* $\mathrm{R}'(I_0, I_1)$ *is not necessarily unique. Note, however, that for nonsingular cases (after all, pruning discards such cases from further consideration), the resulting FIM is unique. Indeed, Problem* $\mathrm{R}'(I_0, I_1)$ *can equivalently be viewed as maximization of the log-determinant over the convex and compact set of matrices* $\mathfrak{M} = \big\{G(w) \mid \sum_{j=1}^{q} w_j = r,\ 0 \le w_j \le 1,\ i = 1, \ldots, q\big\}$*. But the log-determinant is strictly concave over the cone of positive-definite matrices,* $\mathbb{S}^m_{++}$*, which constitutes the interior of* $\mathbb{S}^m_+$ *relative to* $\mathbb{S}^m$*, and this fact implies the unicity of the optimal FIM.*

Simplicial decomposition scheme. Simplicial Decomposition (SD) constitutes an important class of methods for solving large-scale continuous problems in mathematical programming with convex feasible sets (Bertsekas, 1999; Patriksson, 2001; von Hohenbalken, 1977). In the original framework, where a concave objective function is to be maximized over a bounded polyhedron, it iterates by alternately solving a linear programming subproblem (the so-called *column generation problem*) which generates an extreme point of the polyhedron, and a nonlinear *Restricted Master Problem* (RMP) which finds the maximum of the objective function over the convex hull (a simplex) of previously defined extreme points. This basic strategy of simplicial decomposition has appeared in numerous references (Hearn *et al.*, 1985; 1987; Ventura and Hearn, 1993) where possible improvements and extensions have also been discussed. A principal characteristic of an SD method is that the sequence of successive solutions to the master problem tends to a solution to the original problem in such a way that the objective function strictly monotonically approaches its optimal value.

Problem $\mathrm{R}'(I_0, I_1)$ is perfectly suited for the application of the SD scheme. In this case, it boils down to Algorithm 2. Here $\nabla\mathcal{Q}(w)$ signifies the gradient of $\mathcal{Q}$ at w, and it is easy to check that

$$\nabla\mathcal{Q}(w) = \Big[\mathrm{trace}\big(G^{-1}(w)S_1\big), \quad \ldots, \quad \mathrm{trace}\big(G^{-1}(w)S_q\big)\Big]^{\mathsf{T}}. \tag{6.31}$$

Since we deal with maximization of a concave function $\mathcal{Q}$ over a bounded polyhedral set W, the convergence of Algorithm 2 in a finite number of RMP steps is automatically guaranteed (Bertsekas, 1999, p. 221; von Hohenbalken, 1977). Observe that Step 3 implements the *column dropping rule* (Patriksson, 2001), according to which any extreme point with zero weight in the expression of $w^{(k)}$ as a convex combination of elements in $Z^{(k)}$ is removed. This rule makes the number of elements in successive sets $Z^{(k)}$ reasonably low.

Algorithm 2. Algorithm model for simplicial decomposition.

Step 0: (Initialization)
Set $w^{(0)} = (r/q, \ldots, r/q)$ and $Z^{(0)} = \{w^{(0)}\}$. Select $0 < \epsilon \ll 1$, a parameter used in the stopping rule, and set $k = 0$.

Step 1: (Solution of the column generation subproblem)
Determine

$$z = \arg \max_{w \in W} \nabla \mathcal{Q}(w^{(k)})^{\mathsf{T}}(w - w^{(k)}). \tag{6.32}$$

Step 2: (Termination check)
If $\nabla \mathcal{Q}(w^{(k)})^{\mathsf{T}}(z - w^{(k)}) \leq \epsilon$, then STOP and $w^{(k)}$ is optimal. Otherwise, set $Z^{(k+1)} = Z^{(k)} \cup \{z\}$.

Step 3: (Solution of the restricted master problem)
Find

$$w^{(k+1)} = \arg \max_{w \in \mathrm{co}(Z^{(k+1)})} \mathcal{Q}(w) \tag{6.33}$$

and purge $Z^{(k+1)}$ of all extreme points with zero weight in the expression of $w^{(k+1)}$ as a convex combination of elements in $Z^{(k+1)}$. Increment k by one and go back to Step 1.

The SD algorithm may be viewed as a form of modular nonlinear programming, provided that one has an effective computer code for solving the restricted master problem, as well as access to a code which can take advantage of the linearity of the column generation subproblem (Hearn *et al.*, 1987). The former issue will be addressed in the next subsection, where an extremely simple and efficient multiplicative algorithm for weight optimization will be discussed. In turn, the latter issue is easily settled, as in the linear programming problem of Step 1 the feasible region W is defined by one equality constraint (6.25) and q bound constraints (6.26). The special form of the constraints can be exploited directly, since numerous techniques have been proposed to achieve considerable speedup, ranging from improvements on the simplex method (cf. its upper-bounding version described by Pierre (1969, p. 224) to large-scale interior-point methods which are accessible in popular numerical packages (cf. the primal-dual interior-point variant of Mehrotra's predictor-corrector algorithm implemented in the MATLAB's Optimization Toolbox (MathWorks, 2000), cf. (Nocedal and Wright, 1999).

Multiplicative algorithm for the RMP. Suppose that in the $(k+1)$-th iteration of Algorithm 2, we have

$$Z^{(k+1)} = \{z_1, \ldots, z_s\}, \tag{6.34}$$

possibly with $s < k+1$ owing to the built-in deletion mechanism of points in $Z^{(i)}$, $1 \leq i \leq k$, which did not contribute to the convex combinations yielding the corresponding iterates $w^{(\ell)}$. Step 3 of Algorithm 2 involves maximization of the design criterion (6.24) over

$$\mathrm{co}(Z^{(k+1)}) = \left\{ \sum_{\ell=1}^{s} \alpha_\ell z_\ell \,\middle|\, \alpha_\ell \geq 0,\ \ell = 1, \ldots, s,\ \sum_{\ell=1}^{s} \alpha_\ell = 1 \right\}. \tag{6.35}$$

From the representation of any $w \in \mathrm{co}(Z^{(k+1)})$ as

$$w = \sum_{\ell=1}^{s} \alpha_\ell z_\ell, \tag{6.36}$$

or, in a component-wise form,

$$w_j = \sum_{\ell=1}^{s} \alpha_\ell z_{\ell,j}, \quad j = 1, \ldots, q, \tag{6.37}$$

$z_{\ell,j}$ being the j-th component of z_ℓ, it follows that

$$G(w) = A + \sum_{j=1}^{q} w_j S_j = \sum_{\ell=1}^{s} \alpha_\ell \Big(A + \sum_{j=1}^{q} z_{\ell,j} S_j \Big) = \sum_{\ell=1}^{s} \alpha_\ell G(z_\ell). \tag{6.38}$$

From this, we see that the RMP can equivalently be formulated as the following problem:

Problem $\mathbf{P_{RMP}}$: Find the sequence of weights $\alpha = (\alpha_1, \ldots, \alpha_s)$ to maximize

$$\mathcal{T}(\alpha) = \log\det\big(H(\alpha)\big) \tag{6.39}$$

subject to the constraints

$$\sum_{\ell=1}^{s} \alpha_\ell = 1, \tag{6.40}$$

$$\alpha_\ell \geq 0, \quad \ell = 1, \ldots, s, \tag{6.41}$$

where

$$H(\alpha) = \sum_{\ell=1}^{s} \alpha_\ell Q_\ell, \qquad Q_\ell = G(z_\ell). \tag{6.42}$$

Basically, since the constraints (6.40) and (6.41) define the probability simplex P_s in $\mathbb{R}^s$, i.e., a very nice convex feasible domain, it is intuitively appealing to determine optimal weights using a numerical algorithm specialized for solving convex optimization problems. But another, much simpler technique can be employed to suitably guide weight calculation. It fully exploits the specific form of the objective function (6.39) by giving Problem $\mathrm{P_{RMP}}$ an equivalent probabilistic formulation. Specifically, the nonnegativeness of the weights $z_{\ell,j}$, $j = 1, \ldots, q$ and the nonnegative definiteness of the matrices A and S_j, $j = 1, \ldots, q$ imply that $Q_\ell \succeq 0$, $\ell = 1, \ldots, q$. Defining $\mathcal{X}$ as a discrete random variable which may take values in the set $\{1, \ldots, s\}$ and treating the weights α_ℓ, $\ell = 1, \ldots, s$ as the probabilities attached to its possible numerical values, i.e.,

$$p_{\mathcal{X}}(\ell) = \mathbb{P}(\mathcal{X} = \ell) = \alpha_\ell, \quad \ell = 1, \ldots, s, \tag{6.43}$$

we can interpret $p_{\mathcal{X}}$ as the probability mass function (pmf) of $\mathcal{X}$ and $H(\alpha) = \sum_{\ell=1}^{s} \alpha_\ell Q_\ell$ in (6.39) as the $\mathbb{P}$-weighted mean of the function $\mathcal{Q} : \ell \mapsto Q_\ell$. Therefore, Problem $\mathcal{P}_{\mathrm{RMP}}$ can be thought of as that of finding a probability mass function

maximizing the log-determinant of the mean of $\mathcal{Q}$. This formulation has captured close attention in optimum experimental design theory, where various characterizations of optimal solutions and efficient computational schemes have been proposed (Atkinson and Donev, 1992; Fedorov and Hackl, 1997; Walter and Pronzato, 1997). In particular, in the case of the D-optimality criterion studied here, we can prove the following conditions for global optimality:

Proposition 6.3. *Suppose that the matrix $H(\alpha^\star)$ is nonsingular for some $\alpha^\star \in P_s$. The vector $\alpha^\star$ constitutes a global solution to Problem P_{RMP} if and only if*

$$\psi(\ell,\alpha^\star)\begin{cases}=m & \text{if } \alpha_\ell^\star > 0,\\ \le m & \text{if } \alpha_\ell^\star = 0\end{cases} \tag{6.44}$$

for each $\ell = 1,\dots,s$, where

$$\psi(\ell,\alpha) = \operatorname{trace}\bigl[H^{-1}(\alpha)Q_\ell\bigr], \quad \ell = 1,\dots,s. \tag{6.45}$$

A very simple and numerically effective sequential procedure was devised and analysed in (Pázman, 1986; Silvey *et al.*, 1978; Torsney, 1988; Torsney and Mandal, 2001; 2004) for the case of rank-one matrices Q_ℓ, which was then extended to the general case by Uciński (2005, p. 62). Its version adapted to the RMP proceeds as summarized in Algorithm 3. Clear advantages here are ease of implementation and negligible additional memory requirements.

Algorithm 3. Algorithm model for the restricted master problem.

Step 0: (Initialization)
Select weights $\alpha_\ell^{(0)} > 0$, $\ell = 1,\dots,s$ which determine the initial pmf $p_{\mathcal{X}^{(0)}}$ for which we must have $\mathcal{T}(\alpha^{(0)}) > -\infty$, e.g., set $\alpha_\ell^{(0)} = 1/s$, $\ell = 1,\dots,s$. Choose $0 < \eta \ll 1$, a parameter used in the stopping rule. Set $\kappa = 0$.

Step 1: (Termination check)
If

$$\frac{\psi(\ell,\alpha^{(\kappa)})}{m} < 1+\eta, \quad \ell = 1,\dots,s, \tag{6.46}$$

then STOP.

Step 2: (Multiplicative update)
Evaluate

$$\alpha_\ell^{(\kappa+1)} = \alpha_\ell^{(\kappa)}\,\frac{\psi(\ell,\alpha^{(\kappa)})}{m}, \quad \ell = 1,\dots,s. \tag{6.47}$$

Increment κ by one and go to Step 1.

The idea is reminiscent of the EM algorithm used for maximum likelihood estimation (Lange, 1999). The properties of this computational scheme are considered in some detail in (Uciński, 2005). Suffice it to say here that Algorithm 3 is globally convergent regardless of the choice of initial weights (they must only be all nonzero and correspond to a nonsingular FIM). Indeed, we have the following result (Uciński, 2005, p. 65):

Proposition 6.4. *Assume that* $\{\alpha^{(\kappa)}\}$ *is a sequence of iterates constructed by Algorithm 3. Then the sequence* $\{\mathcal{T}(\alpha^{(\kappa)})\}$ *is monotone increasing and*

$$\lim_{\kappa\to\infty} \mathcal{T}(\alpha^{(\kappa)}) = \max_{\alpha\in P_s} \mathcal{T}(\alpha). \tag{6.48}$$

The basic scheme of Algorithm 3 can be refined to incorporate various improvements which make convergence much faster. For example, produced solutions often happen to contain many insignificant weights α_ℓ, which results from a limited accuracy of computations and the interruption of Algorithm 3 after a finite number of steps. In practice, these weights may well be disregarded since setting them as zeros and distributing the sum of their values among the remaining weights (so as not to violate (6.40)) involves a negligible change in the value of the performance measure $\mathcal{T}(\alpha^{(\kappa)})$. The sum of the weights removed can be distributed among the other weights for which $\psi(\ell,\alpha^{(\kappa)}) > m$ and, additionally, in a manner proportional to $\psi(\ell,\alpha^{(\kappa)}) - m$.

Another improvement is due to Pronzato (2003), who proposed a simple method to identify elements of $Z^{(k+1)}$ which do not contribute to the sought optimal convex combination in $\mathrm{co}(Z^{(k+1)})$. It can be generalized to the general case considered here and used during the search to discard such useless points 'on the fly', thereby substantially reducing problem dimensionality.

6.4. Approximate solution via continuous relaxation

Now we wish to return to the formulation of Problem P and present an alternative approach to its solution, which is based on the idea of its reduction to a form for which efficient tools of convex analysis can be applied.

6.4.1. Conversion to the problem of finding optimal sensor densities

When the numbers of candidate sites and sensors to be located n are large, which becomes a common situation in applications, we can operate on the spatial density of sensors (i.e., the number of sensors per unit area), rather than on the individual sensor locations. The density of sensors over the time interval T can be approximately described by a probability measure $\xi(\mathrm{d}x)$ on the space $(X,\mathcal{B})$, where $\mathcal{B}$ is the σ-algebra of all Borel subsets of X. Feasible solutions of this form make it possible to apply convenient and efficient mathematical tools of convex programming theory. As regards the practical interpretation of the so produced results (provided that we are in a position to calculate at least their approximations), one possibility is to partition X into nonoverlapping subdomains X_i of relatively small areas and then to allocate to each of them the number

$$N_i = \left\lceil N \int_{X_i} \xi(\mathrm{d}x) \right\rceil \tag{6.49}$$

of sensors (here $\lceil\rho\rceil$ is the smallest integer greater than or equal to ρ).

Accordingly, we define the class of admissible designs as all probability measures ξ over X which are absolutely continuous with respect to the Lebesgue measure and

satisfy by definition the condition

$$\int_X \xi(\mathrm{d}x) = 1. \tag{6.50}$$

Consequently, we replace (6.7) by

$$M(\xi) = \int_X \Upsilon(x)\, \xi(\mathrm{d}x), \tag{6.51}$$

where

$$\Upsilon(x) = \frac{1}{t_f} \int_0^{t_f} g(x,t) g^{\mathsf{T}}(x,t)\, \mathrm{d}t$$

and the integration in (6.50) and (6.51) is to be understood in the Lebesgue-Stieltjes sense. This leads to the so-called *continuous* designs, which constitute the basis of the modern theory of optimal experiments (Fedorov and Hackl, 1997; Walter and Pronzato, 1997).

We impose the crucial restriction that the density of sensor allocation must not exceed some prescribed level. For a design measure $\xi(\mathrm{d}x)$ this amounts to the condition

$$\xi(\mathrm{d}x) \leq \omega(\mathrm{d}x), \tag{6.52}$$

where $\omega(\mathrm{d}x)$ signifies the maximal possible 'number' of sensors per $\mathrm{d}x$ (Fedorov and Hackl, 1997; Uciński, 2005; 1999) such that

$$\int_X \omega(\mathrm{d}x) \geq 1. \tag{6.53}$$

Consequently, we are faced with the following optimization problem:

Problem $\mathbf{P_{relax}}$: Find a design measure $\xi \in \Xi(X)$, $\Xi(X)$ being the set of all probability measures on X, to maximize

$$\mathfrak{J}(\xi) = \log\det(M(\xi)) \tag{6.54}$$

subject to

$$\xi(\mathrm{d}x) \leq \omega(\mathrm{d}x). \tag{6.55}$$

The design $\xi^\star$ above is then said to be a (Ψ, ω)-*optimal design* (Fedorov and Hackl, 1997).

6.4.2. Optimality conditions

Let us make the following assumptions:

(A1) X is compact,

(A2) $g \in C(X \times T; \mathbb{R}^m)$,

(A3) $\big\{\xi : \det(M(\xi)) \neq 0\big\} \neq \emptyset$,

(A4) $\omega(\mathrm{d}x)$ is atomless, i.e., for any $\Delta X \subset X$ there exists a $\Delta X' \subset \Delta X$ such that

$$\int_{\Delta X'} \omega(\mathrm{d}x) < \int_{\Delta X} \omega(\mathrm{d}x). \tag{6.56}$$

In what follows, we write $\bar{\Xi}(X)$ for the collection of all the design measures which satisfy the requirement[1]

$$\xi(\Delta X) = \begin{cases} \omega(\Delta X) & \text{for } \Delta X \subset \operatorname{supp}\xi, \\ 0 & \text{for } \Delta X \subset X \setminus \operatorname{supp}\xi. \end{cases} \tag{6.57}$$

Given a design ξ, we will say that the function $\psi(\,\cdot\,,\xi)$ defined by

$$\phi(x,\xi) = \frac{1}{t_f} \int_0^{t_f} g^{\mathsf{T}}(x,t) M^{-1}(\xi) g(x,t)\,\mathrm{d}t \tag{6.58}$$

separates the sets X_1 and X_2 with respect to $\omega(\mathrm{d}x)$ if for any two sets $\Delta X_1 \subset X_1$ and $\Delta X_2 \subset X_2$ with equal nonzero ω-measures we have

$$\int_{\Delta X_1} \phi(x,\xi)\,\omega(\mathrm{d}x) \geq \int_{\Delta X_2} \phi(x,\xi)\,\omega(\mathrm{d}x). \tag{6.59}$$

We can now formulate the following characterization of (Ψ,ω)-optimal designs, see (Uciński, 1999; 2005).

Proposition 6.5. (Uciński, 1999; 2005) *Let Assumptions (A1)–(A4) hold. Then*

(i) there exists an optimal design $\xi^\star \in \bar{\Xi}(X)$, and

(ii) a necessary and sufficient condition for $\xi^\star \in \bar{\Xi}(X)$ to be (Ψ,ω)-optimal is that $\phi(\,\cdot\,,\xi^\star)$ separates $X^\star = \operatorname{supp}\xi^\star$ and its complement $X \setminus X^\star$ with respect to the measure $\omega(\mathrm{d}x)$.

From a practical point of view, the above result means that at all the support points of an optimal design $\xi^\star$ the mapping $\phi(\,\cdot\,,\xi^\star)$ should be greater than anywhere else, i.e., preferably $\operatorname{supp}\xi^\star$ should coincide with maximum points of $\phi(\,\cdot\,,\xi^\star)$. In practice, this amounts to allocating observations to the points at which we know least of all about the system response.

If we were able to construct a design with this property, then it would be qualified as an optimal design. Clearly, unless the design problem considered is quite simple, we must employ a numerical algorithm to make the outlined idea useful.

[1] The support of a measure ξ is defined as the closed set $\operatorname{supp}\xi = X \setminus \bigcup\{G : \xi(G) = 0,\ G - \text{open}\}$, cf. (Rao, 1987, p.80).

Algorithm 4. Exchange algorithm model to allocate a large number of network nodes.

Step 1: Guess a nondegenerate initial design $\xi^{(0)} \in \bar{\Xi}(X)$ (i.e., it is required that $\det(M(\xi^{(0)})) \neq 0$). Set $k = 0$.

Step 2: Set $X_1^{(k)} = \operatorname{supp} \xi^{(k)}$ and $X_2^{(k)} = X \setminus X_1^{(k)}$. Determine

$$x_1^{(k)} = \arg \min_{x \in X_1^{(k)}} \phi(x, \xi^{(k)}), \quad x_2^{(k)} = \arg \max_{x \in X_2^{(k)}} \phi(x, \xi^{(k)}).$$

If $\phi(x_1^{(k)}, \xi^{(k)}) > \phi(x_2^{(k)}, \xi^{(k)}) - \eta$, where $0 < \eta \ll 1$, then STOP. Otherwise, find two sets $S_1^{(k)} \subset X_1^{(k)}$ and $S_2^{(k)} \subset X_2^{(k)}$ such that $x_1^{(k)} \in S_1^{(k)}$, $x_2^{(k)} \in S_2^{(k)}$ and

$$\int_{S_1^{(k)}} \omega(\mathrm{d}x) = \int_{S_2^{(k)}} \omega(\mathrm{d}x) = \alpha_k$$

(i.e., the ω-measures of $S_1^{(k)}$ and $S_2^{(k)}$ must be identical) for some $\alpha_k > 0$.

Step 3: Construct $\xi^{(k+1)}$ such that

$$\operatorname{supp} \xi^{k+1} = X_1^{(k+1)} = (X_1^{(k)} \setminus S_1^{(k)}) \cup S_2^{(k)}.$$

Increment k and go to Step 2.

6.4.3. Exchange algorithm

Since $\xi^\star(\mathrm{d}x)$ should be nonzero in the areas where $\phi(\,\cdot\,, \xi^\star)$ takes on a larger value, the central idea when constructing a computational algorithm for sensor density optimization is to move some measure from areas with smaller values of $\psi(\,\cdot\,, \xi^{(k)})$ to those with larger values, as we expect that such a procedure will improve $\xi^{(k)}$. This is embodied by Algorithm 4, being an adaptation of the algorithm presented in (Fedorov, 1989):

Convergence is guaranteed if the sequence $\{\alpha\}_{k=0}^{\infty}$ satisfies (Fedorov, 1989):

$$\lim_{k \to \infty} \alpha_k = 0, \quad \sum_{k=0}^{\infty} \alpha_k = \infty. \tag{6.60}$$

Within the framework of sensor placement, we usually have $\omega(\mathrm{d}x) = \varrho(x)\,\mathrm{d}x$, where ϱ is a density function. But in this situation we may restrict our attention to constant ϱ's (indeed, in any case we can perform an appropriate change of coordinates). Moreover, while implementing the algorithm on a computer, all integrals are replaced by sums over some regular grid elements. Analogously, the sets X, $X_1^{(k)}$, $X_2^{(k)}$, $S_1^{(k)}$ and $S_2^{(k)}$ then simply consist of grid elements. Consequently, the above iterative procedure may be considered as an exchange-type algorithm with the additional constraint that every grid element must not contain more than one supporting point and the weights of all supporting points are equal to $1/N$. In practice, α_k is usually fixed and, what is more, one-point exchanges are most often adopted, i.e., $S_1^{(k)} = \{x_1^{(k)}\}$ and $S_2^{(k)} = \{x_2^{(k)}\}$, which substantially simplifies implementation. Let us note, however, that convergence to an optimal design is assured only for decreasing

α_k's and hence some oscillations in $\Psi[M(\xi^{(k)})]$ may sometimes be observed. A denser spatial grid usually constitutes a remedy for this predicament (Müller, 2001).

6.5. Computational results

In order to illustrate and compare the discussed approaches to the sensor placement, they were applied to the problem of optimal estimation of the spatial-varying diffusion coefficient $a(x)$ in the transport process of an air pollutant over a given urban area normalized to the unit square $\Omega = (0,1)^2$. Within this domain an active source of pollution is present, which leads to changes in the pollutant concentration $y = y(x,t)$. The evolution of y over the normalized observation interval $T = (0,1]$ is described by the following advection-diffusion equation:

$$\frac{\partial y(x,t)}{\partial t} + \nabla \cdot \big(v(x)y(x,t)\big) = \nabla \cdot \big(a(x)\nabla y(x,t)\big) + f(x), \quad x \in \Omega, \tag{6.61}$$

subject to the boundary and initial conditions

$$\frac{\partial y(x,t)}{\partial n} = 0, \quad \text{on } \Gamma \times T, \tag{6.62}$$

$$y(x,0) = y_0, \quad \text{in } \Omega, \tag{6.63}$$

where the term $f(x) = 10\exp\big(-50\|x-c\|^2\big)$ represents a source of the pollutant located at the point $c = (0.2, 0.6)$, and $\partial y/\partial n$ stands for the partial derivative of y with respect to the outward normal to the boundary Γ. The mean spatio-temporal changes of the wind velocity field over the area were approximated by $v = (v_1, v_2)$, where

$$v_1 = 2(x_1 + x_2), \quad v_2 = 2(x_1 - x_2 + t) - 1, \tag{6.64}$$

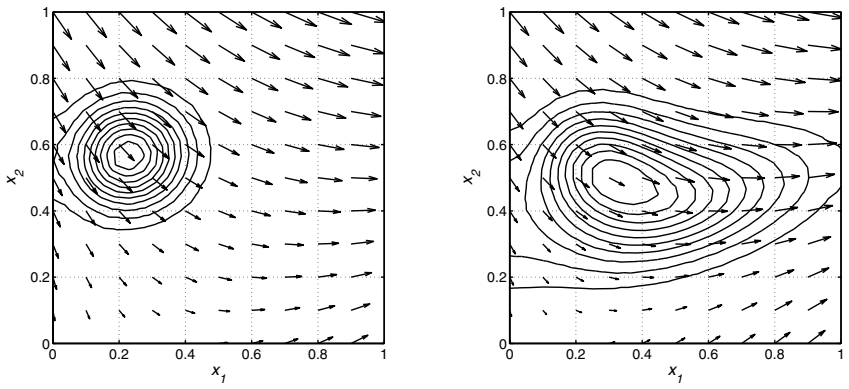
which is also illustrated in Fig. 6.1. The assumed functional form of the spatial-varying diffusion coefficient is

$$a(x) = \theta_1 + \theta_2 x_1 x_2 + \theta_3 x_1^2 + \theta_4 x_2^2, \tag{6.65}$$

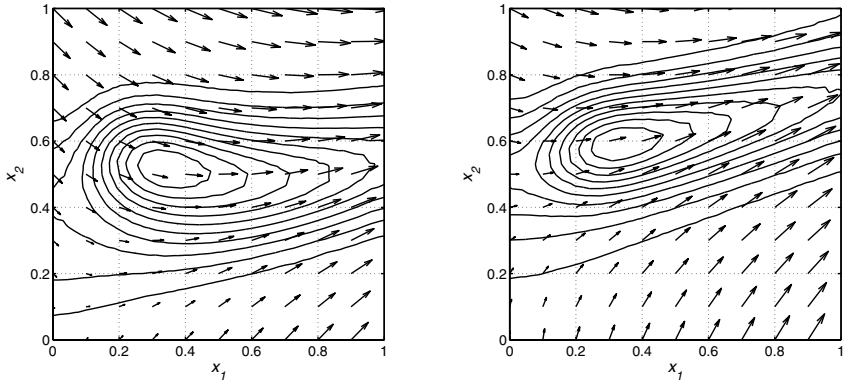
so that the constant parameters $\theta_1, \theta_2, \theta_3$ and θ_4 need estimation based on measurement data from monitoring stations.

In our simulation studies, two described optimization approaches for stationary sensor location were tested, namely, the bound and branch technique and multi-point exchange procedure. Given N prospective sites in $\Omega \cup \Gamma$, we aim at selecting their subset consisting of locations at which the measurements made by n available sensors would lead to D-optimum least-squares estimates of the parameters θ.

In order to determine the elements of the sensitivity vector (6.8) required to calculate FIM, the direct-differentiation method (Uciński, 2005) was applied assuming the nominal values of the parameters $\theta_1^0 = 0.02$, $\theta_2^0 = 0.01$ and $\theta_3^0 = \theta_4^0 = 0.005$. We solved the resulting system of PDEs using some routines of the MATLAB PDE Toolbox for a spatial mesh composed of 1248 triangles and 665 nodes. As for the numerical integration required to evaluate information matrices for admissible observation sites,



(a) $t = 0$ (b) $t = 0.3$



(c) $t = 0.7$ (d) $t = 1$

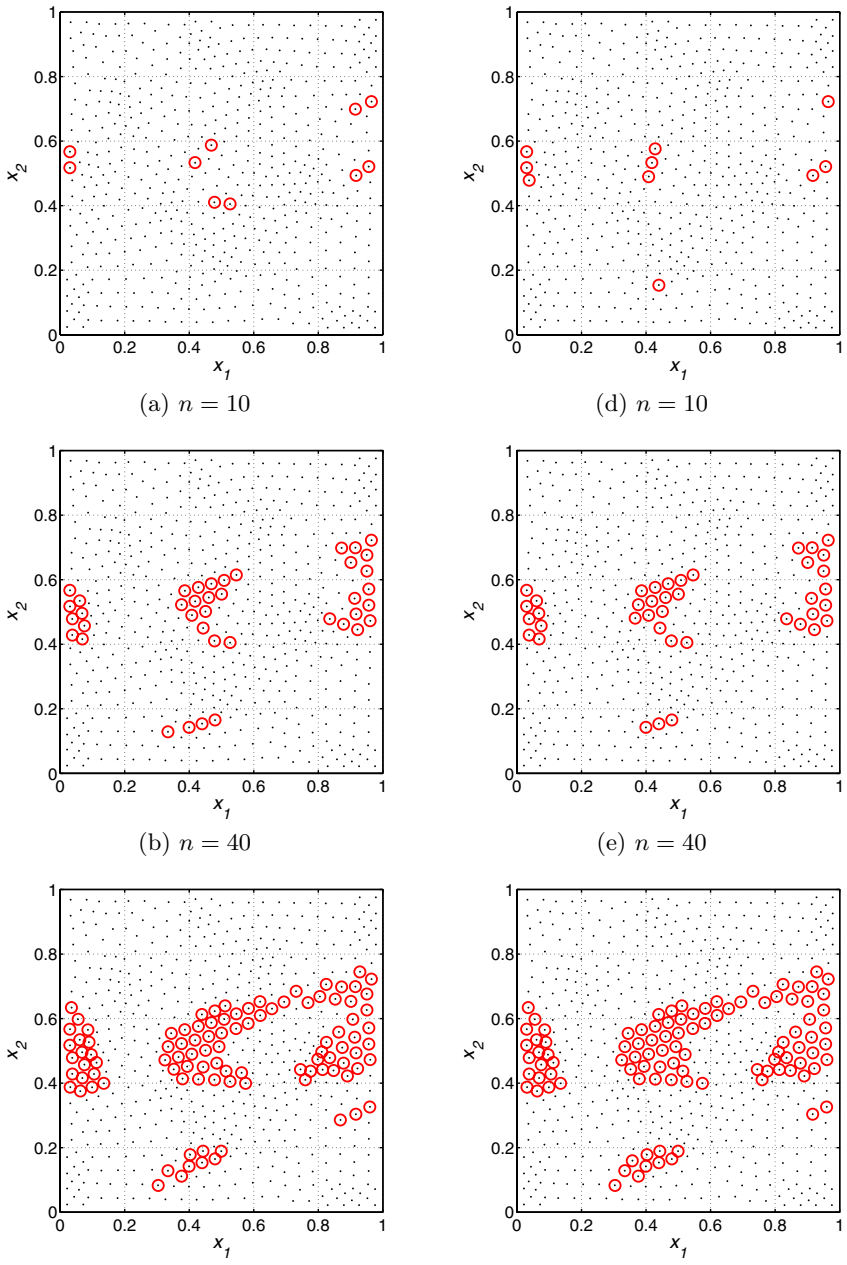
Fig. 6.1. Temporal changes in the wind velocity field and pollutant concentration

the trapezoidal rule was applied with the time step equal to 0.04, based on the sensitivity vector interpolated at the nodes representing the admissible locations x^i, cf. Appendix I in (Uciński, 2005) for details.

The solution to (6.61)–(6.63) is shown in Fig. 6.1, where the complex process dynamics can be easily observed. The pollutant spreads out over the entire domain reflecting the sophisticated combination of diffusion and advection processes and follows the temporary direction of the wind being the dominant transport factor.

In our scenario, the observation grid was assumed to be created at locations selected from among those elements of the above-mentioned 665-point triangulation mesh which do not lie on the outer boundary (there were 585 such nodes, which are indicated with dots in Fig. 6.2). Both tested algorithms were implemented entirely in Matlab 7.1 and tested on a PC equipped with a Pentium IV 1.7 GHz processor and 768 MB RAM, running Windows 2000.

The D-optimal sensor configurations for different numbers of allocated sensors are shown in Fig. 6.2. It is clear that the complexity of the system dynamics makes proper prediction of the observation locations rather difficult and nonintuitive. The sensors



(a) $n = 10$ (d) $n = 10$

(b) $n = 40$ (e) $n = 40$

(c) $n = 100$ (f) $n = 100$

Fig. 6.2. D-optimal allocation of different numbers of sensors for the BB (a)–(c) and exchange (d)–(f) algorithms

Table 6.1. Comparison of algorithms' performance

No. of sensors	BB ($\epsilon = 5 \times 10^{-5}$, $\eta = 10^{-5}$)			multi-point exchange ($\eta = 10^{-5}$)		
	No. of rec. calls	Time [h:m:s]	Criterion value	No. of iterations	Time [h:m:s]	Criterion value
5	795	0:14:37.652	0.2068	7	0:00:00.211	0.1854
10	223	0:05:28.372	0.2058	12	0:00:00.102	0.1959
20	29	0:00:57.523	0.1893	19	0:00:00.114	0.1840
40	3	0:00:05.197	0.1639	40	0:00:00.198	0.1639
60	1	0:00:01.302	0.1425	55	0:00:00.312	0.1425
80	1	0:00:01.291	0.1246	66	0:00:00.353	0.1246
100	1	0:00:01.222	0.1102	82	0:00:00.402	0.1102
150	1	0:00:01.132	0.0829	114	0:00:00.708	0.0829
250	1	0:00:01.191	0.0496	138	0:00:00.924	0.0496

tend to form the pattern reflecting the areas of greatest changes in the pollutant concentration, but the observations are averaged over time and it is not trivial to follow the dynamics of the observation strategy. Surprisingly, the measurements in the closest vicinity of the pollution source turned out to be not very attractive for parameter estimation.

Detailed results concerning the algorithms' performance are presented in Tab. 6.1. Examination of these data leads to some very interesting remarks. Both algorithms seem to work very efficiently for high numbers of sensors and they generate very similar solutions. Although the presented examples are rather of a medium scale, we have to remember that in the worst case (i.e., when the number of sensors is closest to half the number of available sites) the cardinality of the search space for the 585-point grid reaches approximately 4×10^{174}. The multi-point exchange algorithm proves to be extremely rapid but its weakness is the fact that it cannot retrieve the optimum solution in the case of small numbers of sensors, since it gets stuck and oscillates between some suboptimal solutions. In such a situation, the BB approach can still obtain the desired global D-optimal sensor locations, although at the cost of serious computational burden. Nevertheless, even if the time required for obtaining the optimal solution for a small number of sensors is relatively long, the BB approach determines the suboptimal solution with better quality than the exchange procedure with no more than just a few recursive calls (i.e. within no more than 5 sec. of simulation time in each case).

Unexpectedly, with the increased number of sensors (and the size of a search space) the pruning process becomes more efficient and BB becomes more competitive in the sense of computational effort, being always no worse in the sense of solution quality. This effect can be explained by observing that a higher density of sensors provides a better estimate for the lower bound to the optimal value of the design criterion, which results in the observed speedup of pruning.

6.6. Concluding remarks

We have addressed the problem of selecting optimal observation points in view of accurate parameter estimation for parameter distributed systems, which stand here for dynamical systems governed by partial differential equations. Although it has been approached from various angles since the mid-1970s, there are still few systematic and versatile methods for its solution. In the existing formulations, an optimal sensor placement is thus computed as that which globally maximizes a criterion directly connected with the expected quality of the parameter estimates. But then the key difficulty becomes the large scale of the resulting global optimization problem, since the monitoring networks encountered in process industry or environmental engineering may often consist of several hundreds of stations. Obviously, this makes the exhaustive search on a candidate-by-candidate basis practically intractable and creates a need for techniques which would implement a guided search and have acceptable performance.

We started from the most common formulation, in which the measurement system has a finite number of sensor candidate positions and the aim is to select the best subset of points of desired cardinality. Choosing the best subset translates to maximizing the determinant of the Fisher information matrix associated with the estimated parameters and fits into the framework of nonlinear 0–1 integer programming. The solution of this combinatorial design problem using the branch-and-bound method constitutes a quite natural option, but the main problem when trying to implement it has been the lack of a low-cost procedure to obtain upper bounds to the optimal values of the D-optimality criterion. Our main contribution consists in adapting a specialized multiplicative algorithm for determinant maximization to produce such bounds. The link to plug this algorithm into the proposed scheme was a simplicial decomposition being perfectly suited for large-scale problems which can be encountered here. Consequently, the proposed method can be implemented with great ease and our experience provides evidence that, with this tool, even large-scale design problems can be solved using an off-the-shelf PC.

An alternative approach to select a best n-element subset from among a given N-element set of candidate sites was to employ an exchange algorithm. Typically, algorithms of this type begin with an n-point starting sensor configuration which then sequentially evolves through addition of new elements selected from among vacant sites and deletion of sites at which sensors have provisionally been planned to reside, in an effort to improve the value of the adopted design criterion (Meyer and Nachtsheim, 1995). Accordingly, a one-point exchange procedure was discussed here. Originally, it had been used in (Uciński, 1999) and further developed in (Uciński, 2005; Uciński and Patan, 2002) in a sensor-network setting, based on the concept of replication-free designs set forth by Fedorov (1989). A much more efficient extension of this idea could be to adapt the fast algorithm based on multiple simultaneous exchanges, which was developed by Lam *et al.* (2002). A step in this direction was made by Liu *et al.* (2005), who refined it and applied the resulting 'sort-and-cut' technique to solve an E-optimum sensor selection problem. It is beyond doubt that this approach outperforms the BB technique proposed here as far as the running time is concerned. One should note, however, that exchange algorithms are heuristics to a significant measure and thus they are only capable of finding globally competitive solutions (i.e.,

nearly optimal ones), with an explicit trade of global optimality for speed. The BB approach presented here is superior in the sense that it always produces global maxima and, what is more, does it within tolerable time.

References

Amouroux M. and Babary J.P. (1988): *Sensor and control location problems*, In: Systems & Control Encyclopedia. Theory, Technology, Applications (M.G. Singh, Ed.). — Oxford: Pergamon Press, Vol. 6, pp. 4238–4245.

Andó B., Cammarata G., Fichera A., Graziani S. and Pitrone N. (1999): *A procedure for the optimization of air quality monitoring networks*. — IEEE Trans. Systems, Man. Cybernetics, Part C: Applications and Reviews, Vol. 29, No. 1, pp. 157–163.

Armstrong M. (1998): *Basic Linear Geostatistics*. — Berlin: Springer-Verlag.

Atkinson A.C. and Donev A.N. (1992): *Optimum Experimental Designs*. — Oxford: Clarendon Press.

Banks H.T. (1992): *Computational issues in parameter estimation and feedback control problems for partial differential equation systems*. — Physica D, Vol. 60, pp. 226–238.

Banks H.T. and Kunisch K. (1989): *Estimation Techniques for Distributed Parameter Systems*. — Systems & Control: Foundations & Applications, Boston: Birkhäuser.

Banks H.T., Smith R.C. and Wang Y. (1996): *Smart Material Structures: Modeling, Estimation and Control*. — Research in Applied Mathematics, Paris: Masson.

Bennett A.F. (1992): *Inverse Methods in Physical Oceanography*. — Cambridge Monographs on Mechanics and Applied Mathematics, Cambridge University Press.

Bertsekas D.P. (1999): *Nonlinear Programming*. — Belmont, MA: Athena Scientific.

Boer E.P.J., Hendrix E.M.T. and Rasch D.A.M.K. (2001): *Optimization of monitoring networks for estimation of the semivariance function*, In: Moda 6 Advances in Model-Oriented Design and Analysis (A.C. Atkinson, P. Hackl and W. Müller, Eds.). — Proc. 6th Int. Workshop *Model-Oriented Data Analysis, mODa 6*, Puchberg/Schneeberg, Austria, Heidelberg: Physica-Verlag, pp. 21–28.

Boyd S. and Vandenberghe L. (2004): *Convex Optimization*. — Cambridge University Press.

Caselton W.F., Kan L. and Zidek J.V. (1992): *Quality data networks that minimize entropy*, In: Statistics in the Environmental and Earth Sciences (A. Walden and P. Guttorp, Eds.). — New York: Halsted Press, Chap. 2, pp. 10–38.

Caselton W.F. and Zidek J.V. (1984): *Optimal monitoring network design*. — Statistics & Probability Letters, Vol. 2, pp. 223–227.

Cassandras C.G. and Li W. (2005): *Sensor networks and cooperative control*. — European J. Control, Vol. 11, No. 4–5, pp. 436–463.

Chong C.-Y. and Kumar S.P. (2003): *Sensor networks: Evolution, opportunities, and challenges*. — Proc. IEEE, Vol. 91, No. 8, pp. 1247–1256.

Christofides P.D. (2001): *Nonlinear and Robust Control of PDE Systems: Methods and Applications to Transport-Reaction Processes*. — Systems & Control: Foundations & Applications, Boston: Birkhäuser.

Cressie N.A.C. (1993): *Statistics for Spatial Data*. — New York: John Wiley & Sons.

Curtain R.F. and Zwart H. (1995): *An Introduction to Infinite-Dimensional Linear Systems Theory*. — Texts in Applied Mathematics, New York: Springer-Verlag.

Daley R. (1991): *Atmospheric Data Analysis*. — Cambridge University Press.

El Jai A. and Amouroux M. (1990): *Automatique des systèmes distribués*. — Paris: Hermès.

Emirsajłow Z. (1991): *The Linear Quadratic Control Problem for Infinite Dimensional Systems with Terminal Targets*. — Szczecin: Technical University Publishers.

Ewing R.E. and George J.H. (1984): *Identification and control for distributed parameters in porous media flow*. — Proc. 2nd Int. Conf. Distributed Parameter Systems, Vorau, Austria, Lecture Notes in Control and Information Sciences, Berlin: Springer-Verlag, pp. 145–161.

Fedorov V.V. (1989): *Optimal design with bounded density: Optimization algorithms of the exchange type*. — J. Statistical Planning and Inference, Vol. 22, pp. 1–13.

Fedorov V.V. (1996): *Design of spatial experiments: Model fitting and prediction*. — Technical Report TM-13152, Oak Ridge National Laboratory, Oak Ridge, TN.

Fedorov V.V. and Hackl P. (1997): *Model-Oriented Design of Experiments*. — Lecture Notes in Statistics, New York: Springer-Verlag.

Floudas C.A. (2001): *Mixed integer nonlinear programming, MINLP*, In: Encyclopedia of Optimization (C.A. Floudas and P.M. Pardalos, Eds.). — Dordrecht, Kluwer Academic Publishers, Vol. 3, pp. 401–414.

Gerdts M. (2005): *Solving mixed-integer optimal control problems by branch&bound: A case study from automobile test-driving with gear shift*. — J. Optimization Theory and Applications, Vol. 26, pp. 1–18.

Gevers M. (2005): *Identification for control: From the early achievements to the revival of experiment design*. — European J. Control, Vol. 11, Nos. 4–5, pp. 335–352.

Goodwin G.C. and Payne R.L. (1977): *Dynamic System Identification. Experiment Design and Data Analysis*. — Mathematics in Science and Engineering, New York: Academic Press.

Grabowski P. (1999): *Lecture Notes on Optimal Control Systems*. — Cracow: University of Mining and Metallurgy Publishers.

Hearn D.W., Lawphongpanich S. and Ventura J.A. (1985): *Finiteness in restricted simplicial decomposition*. — Operations Research Letters, Vol. 4, No. 3, pp. 125–130.

Hearn D.W., Lawphongpanich S. and Ventura J.A. (1987): *Restricted simplicial decomposition: Computation and extensions*. — Mathematical Programming Study, Vol. 31, pp. 99–118.

Hjalmarsson H. (2005): *From experiment design to closed-loop control*. — Automatica, Vol. 41, pp. 393–438.

Hogg N.G. (1996): *Oceanographic data for parameter estimation*, In: Modern Approaches to Data Assimilation in Ocean Modeling P. Malanotte-Rizzoli (Ed.). — Elsevier Oceanography, Amsterdam: Elsevier, pp. 57–76.

Holnicki P., Kałuszko A., Kurowski M., Ostrowski R. and Żochowski A. (1986): *An urban-scale computer model for short-term prediction of air pollution*. — Archiwum Automatyki i Telemechaniki, Vol. XXXI, No. 1–2, pp. 51–71.

Isakov V. (1998): *Inverse Problems for Partial Differential Equations*. — Applied Mathematical Sciences, New York: Springer-Verlag.

Jeremić A. and Nehorai A. (1998): *Design of chemical sensor arrays for monitoring disposal sites on the ocean floor.* — IEEE Trans. Oceanic Engineering, Vol. 23, No. 4, pp. 334–343.

Jeremić A. and Nehorai A. (2000): *Landmine detection and localization using chemical sensor array processing.* — IEEE Trans. Signal Processing, Vol. 48, No. 5, pp. 1295–1305.

Kammer D.C. (1990): *Sensor placement for on-orbit modal identification and correlation of large space structures.* — Proc. American Control Conf., San Diego, California, USA, Vol. 3, pp. 2984–2990.

Kammer D.C. (1992): *Effects of noise on sensor placement for on-orbit modal identification of large space structures.* — Trans. ASME, Vol. 114, pp. 436–443.

Kincaid R.K. and Padula S.L. (2002): *D-optimal designs for sensor and actuator locations.* — Computers & Operations Research, Vol. 29, pp. 701–713.

Klamka J. (1991): *Controllability of Dynamical Systems.* — Mathematics and Its Applications, Dordrecht, Kluwer Academic Publishers.

Korbicz J., Uciński D., Pieczyński A. and Marczewska G. (1993): *Knowledge-based fault detection and isolation system for power plant.* — Applied Mathematics and Computer Science, Vol. 3, No. 3, pp. 613–630.

Korbicz J. and Zgurowski M.Z. (1991): *Estimation and Control of Stochastic Distributed-Parameter Systems.* — Warsaw: Państwowe Wydawnictwo Naukowe, (in Polish).

Kowalewski A. (2001): *Optimal Control of Infinite Dimensional Distributed Parameter Systems with Delays.* — Cracow: University of Mining and Metallurgy Publishers, (in Polish).

Kubrusly C.S. (1977): *Distributed parameter system identification: A survey.* — Int. J. Control, Vol. 26, No. 4, pp. 509–535.

Kubrusly C.S. and Malebranche H. (1985): *Sensors and controllers location in distributed systems — A survey.* — Automatica, Vol. 21, No. 2, pp. 117–128.

Kuczewski B. (2006): *Computational Aspects of Discrimination between Models of Dynamic Systems.* — Lectures Notes in Control and Computer Science, Vol. 10, University of Zielona Góra Press. Available at `http://zbc.uz.zgora.pl/Content/3094/kuczewski_phd.pdf`

Lam R.L.H., Welch W.J. and Young S.S. (2002): *Uniform coverage designs for molecule selection.* — Technometrics, Vol. 44, No. 2, pp. 99–109.

Lange K. (1999): *Numerical Analysis for Statisticians.* — New York: Springer-Verlag.

Lasiecka I. (1998): *Active noise control in an acoustic chamber: Mathematical theory.* — Proc. 5th Int. Symp. *Methods and Models in Automation and Robotics*, Międzyzdroje, Poland, Vol. 1, pp. 13–22.

Lasiecka I. and Triggiani R. (2000): *Control Theory for Partial Differential Equations: Continuous and Approximation Theories.* — Vol. I and II of Encyclopedia of Mathematics and Its Applications, Cambridge University Press.

Lee H.W.J., Teo K.L. and Lim A.E.B. (2001): *Sensor scheduling in continuous time.* — Automatica, Vol. 37, pp. 2017–2023.

Lee H.W.J., Teo K.L., Rehbock V. and Jennings L.S. (1999): *Control parametrization enhancing technique for optimal discrete-valued control problems.* — Automatica, Vol. 35, pp. 1401–1407.

Liu C.Q., Ding Y. and Chen Y. (2005): *Optimal coordinate sensor placements for estimating mean and variance components of variation sources.* — IEE Trans., Vol. 37, pp. 877–889.

Ljung L. (1999): *System Identification: Theory for the User.* — Upper Saddle River, NJ: Prentice Hall.

Malanotte-Rizzoli P. (Ed.) (1996): *Modern Approaches to Data Assimilation in Ocean Modeling.* — Elsevier Oceanography, Amsterdam: Elsevier.

Malanowski K., Nahorski Z. and Peszyńska M. (Eds.) (1996): *Modelling and Optimization of Distributed Parameter Systems.* — Int. Federation for Information Processing, Boston: Kluwer Academic Publishers.

Martínez S. and Bullo F. (2006): *Optimal sensor placement and motion coordination for target tracking.* — Automatica, Vol. 42, pp. 661–668.

MathWorks (2000): *Optimization Toolbox for Use with Matlab. User's Guide, Version 2.* — Natick, MA: The MathWorks, Inc.

Meyer R.K. and Nachtsheim C.J. (1995): *The coordinate-exchange algorithm for constructing exact optimal experimental designs.* — Technometrics, Vol. 37, No. 1, pp. 60–69.

Müller W.G. (2001): *Collecting Spatial Data. Optimum Design of Experiments for Random Fields.* — Contributions to Statistics, Heidelberg: Physica-Verlag.

Munack A. (1984): *Optimal sensor allocation for identification of unknown parameters in a bubble-column loop bioreactor*, In: Analysis and Optimization of Systems, Part 2, (A.V. Balakrishnan and M. Thoma (Eds.). — Lecture Notes in Control and Information Sciences, Berlin: Springer-Verlag, Vol. 63, pp. 415–433.

Navon I.M. (1997): *Practical and theoretical aspects of adjoint parameter estimation and identifiability in meteorology and oceanography.* — Dynamics of Atmospheres and Oceans, Vol. 27, pp. 55–79.

Nehorai A., Porat B. and Paldi E. (1995): *Detection and localization of vapor-emitting sources.* — IEEE Trans. Signal Processing, Vol. 43, No. 1, pp. 243–253.

Nocedal J. and Wright S.J. (1999): *Numerical Opimization.* — New York: Springer-Verlag.

Nychka D., Piegorsch W.W. and Cox L.H. (eds) (1998): *Case Studies in Environmental Statistics.* — Lecture Notes in Statistics, Vol. 132, New York: Springer-Verlag.

Nychka D. and Saltzman N. (1998): *Design of air-quality monitoring networks*, In: Case Studies in Environmental Statistics (D. Nychka, W.W. Piegorsch and L.H. Cox, Eds.). — Lecture Notes in Statistics, New York: Springer-Verlag, Vol. 132,pp. 51–76.

Ögren P., Fiorelli E. and Leonard N.E. (2004): *Cooperative control of mobile sensor networks: Adaptive gradient climbing in a distributed environment.* — IEEE Trans. Automatic Control, Vol. 49, No. 8, pp. 1292–1302.

Omatu S. and Matumoto K. (1991a): *Distributed parameter identification by regularization and its application to prediction of air pollution.* — Int. J. Syst. Sci., Vol. 22, No. 10, pp. 2001–2012.

Omatu S. and Matumoto K. (1991b): *Parameter identification for distributed systems and its application to air pollution estimation.* — Int. J. Syst. Sci., Vol. 22, No. 10, pp. 1993–2000.

Omatu S. and Seinfeld J.H. (1989): *Distributed Parameter Systems: Theory and Applications.* — Oxford Mathematical Monographs, New York: Oxford University Press.

Patan M. (2004): *Optimal Observation Strategies for Parameter Estimation of Distributed Systems.* — Lecture Notes in Control and Computer Science, Vol. 5, University of Zielona Góra Press. Available at `http://zbc.uz.zgora.pl/Content/526/Patan-01.pdf`

Patan M. (2006): *Optimal activation policies for continuous scanning observations in parameter estimation of distributed systems.* — Int. J. Syst. Sci., Vol. 37, No. 11, pp. 763–775.

Patan M. and Patan K. (2005): *Optimal observation strategies for model-based fault detection in distributed systems.* — Int. J. Control, Vol. 78, No. 18, pp. 1497–1510.

Patriksson M. (2001): *Simplicial decomposition algorithms*, In: Encyclopedia of Optimization (C.A. Floudas and P.M. Pardalos, Eds.). — Dordrecht, Kluwer Academic Publishers, Vol. 5, pp. 205–212.

Pázman A. (1986): *Foundations of Optimum Experimental Design.* — Mathematics and Its Applications, Dordrecht, D. Reidel Publishing Company.

Phillipson G.A. (1971): *Identification of Distributed Systems.* — Modern Analytic and Computational Methods in Science and Mathematics, New York: Elsevier.

Pierre D.A. (1969): *Optimization Theory with Applications.* — Series in Decision and Control, New York: John Wiley & Sons.

Point N., Vande Wouwer A. and Remy M. (1996): *Practical issues in distributed parameter estimation: Gradient computation and optimal experiment design.* — Control Engineering Practice, Vol. 4, No. 11, pp. 1553–1562.

Polis M.P. (1982): *The distributed system parameter identification problem: A survey of recent results.* — Proc. 3rd IFAC Symp. *Control of Distributed Parameter Systems*, Toulouse, France, pp. 45–58.

Porat B. and Nehorai A. (1996): *Localizing vapor-emitting sources by moving sensors.* — IEEE Trans. Signal Processing, Vol. 44, No. 4, pp. 1018–1021.

Pronzato L. (2003): *Removing non-optimal support points in D-optimum design algorithms.* — Statistics & Probability Letters, Vol. 63, pp. 223–228.

Pukelsheim F. (1993): *Optimal Design of Experiments.* — Probability and Mathematical Statistics, New York: John Wiley & Sons.

Quereshi Z.H., Ng T.S. and Goodwin G.C. (1980): *Optimum experimental design for identification of distributed parameter systems.* — Int. J. Control, Vol. 31, No. 1, pp. 21–29.

Rafajłowicz E. (1981): *Design of experiments for eigenvalue identification in distributed-parameter systems.* — Int. J. Control, Vol. 34, No. 6, pp. 1079–1094.

Rafajłowicz E. (1983): *Optimal experiment design for identification of linear distributed-parameter systems: Frequency domain approach.* — IEEE Trans. Automatic Control, Vol. 28, No. 7, pp. 806–808.

Rafajłowicz E. (1986): *Optimum choice of moving sensor trajectories for distributed parameter system identification.* — Int. J. Control, Vol. 43, No. 5, pp. 1441–1451.

Rao M.M. (1987): *Measure Theory and Integration.* — New York: John Wiley & Sons.

Reinefeld A. (2001): *Heuristic search*, In: Encyclopedia of Optimization (C.A. Floudas and P.M. Pardalos, Eds.). — Dordrecht, Kluwer Academic Publishers, Vol. 2, pp. 409–411.

Russell S.J. and Norvig P. (2003): *Artificial Intelligence: A Modern Approach.* — Upper Saddle River, NJ: Pearson Education International.

Silvey S.D., Titterington D.M. and Torsney B. (1978): *An algorithm for optimal designs on a finite design space*. — Communications in Statistics — Theory and Methods, Vol. 14, pp. 1379–1389.

Sinopoli B., Sharp C., Schenato L., Schaffert S. and Sastry S.S. (2003): *Distributed control applications within sensor networks*. — Proc. IEEE, Vol. 91, No. 8, pp. 1235–1246.

Sokołowski J. and Zolesio J.-P. (1992): *Introduction to Shape Optimization: Shape Sensitivity Analysis*. — Computational Mathematics, Berlin: Springer-Verlag.

Sturm P.J., Almbauer R.A. and Kunz R. (1994): *Air quality study for the city of Graz, Austria*, In: Urban Air Pollution (H. Power, N. Moussiopoulos and C.A. Brebbia, Eds.). — Southampton: Computational Mechanics Publications, Vol. 1, Chap. 2, pp. 43–100.

Sun N.-Z. (1994): *Inverse Problems in Groundwater Modeling*. — Theory and Applications of Transport in Porous Media, Dordrecht, Kluwer Academic Publishers.

Sydow A., Lux T., Mieth P., Schmidt M. and Unger S. (1997): *The DYMOS model system for the analysis and simulation of regional air pollution*, In: Modellierung und Simulation im Umweltbereich (R. Grützner, Ed.). — Wiesbaden: Vieweg-Verlag, pp. 209–219.

Sydow A., Lux T., Rosé H., Rufeger W. and Walter B. (1998): *Conceptual design of the branch-oriented simulation system DYMOS (dynamic models for smog analysis)*. — Trans. Society for Computer Simulation Int., Vol. 15, No. 3, pp. 95–100.

Titterington D.M. (1980): *Aspects of optimal design in dynamic systems*. — Technometrics, Vol. 22, No. 3, pp. 287–299.

Torsney B. (1988): *Computing optimising distributions with applications in design, estimation and image processing*, In: Optimal Design and Analysis of Experiments (Y. Dodge, V.V. Fedorov and H.P. Wynn, Eds.). — Amsterdam: Elsevier, pp. 316–370.

Torsney B. and Mandal S. (2001): *Construction of constrained optimal designs*, In: Optimum Design 2000 (A. Atkinson, B. Bogacka and A. Zhigljavsky, Eds.). — Dordrecht, Kluwer Academic Publishers, Chap. 14, pp. 141–152.

Torsney B. and Mandal S. (2004): *Multiplicative algorithms for constructing optimizing distributions: Further developments*, In: Moda 7, Advances in Model-Oriented Design and Analysis (A. Di Bucchianico, H. Läuter and H.P. Wynn, Eds.). — Proc. 7th Int. Workshop *Model-Oriented Data Analysis, mODa 7*, Heeze, Heidelberg: Physica-Verlag, pp. 163–171.

Uciński D. (1999): *Measurement Optimization for Parameter Estimation in Distributed Systems*. — Technical University of Zielona Góra Press, Available in electronic form at `http://www.issi.uz.zgora.pl/~ucinski/`

Uciński D. (2000): *Optimal sensor location for parameter estimation of distributed processes*. — Int. J. Control, Vol. 73, No. 13, pp. 1235–1248.

Uciński D. (2005): *Optimal Measurement Methods for Distributed-Parameter System Identification*. — Boca Raton, FL: CRC Press.

Uciński D. and Atkinson A.C. (2004): *Experimental design for time-dependent models with correlated observations*. — Studies in Nonlinear Dynamics & Econometrics, Vol. 8, No. 2., Article No. 13

Uciński D. and Bogacka B. (2005): *T-optimum designs for discrimination between two multivariate dynamic models*. — J. Royal Statistical Society: Series B (Statistical Methodology), Vol. 67, pp. 3–18.

Uciński D. and Chen Y. (2005): *Time-optimal path planning of moving sensors for parameter estimation of distributed systems.* — Proc. 44th IEEE Conf. *Decision and Control, and the European Control*, Seville, Spain. CD-ROM

Uciński D. and El Jai A. (1997): *On weak spreadability of distributed-parameter systems and its achievement via linear-quadratic control techniques.* — IMA J. Mathematical Control and Information, Vol. 14, pp. 153–174.

Uciński D. and El Yacoubi S. (1998): *Modelling and simulation of an ecological problem by means of cellular automata.* — Proc. 5th Int. Symp. *Methods and Models in Automation and Robotics, MMAR*, Międzyzdroje, Poland, Vol. 1, pp. 289–293.

Uciński D. and El Yacoubi S. (1999): *Parameter estimation of cellular automata models.* — Proc. 3rd Int. Conf. *Parallel Processing & Applied Mathematics*, Kazimierz Dolny, Poland, pp. 168–176.

Uciński D. and Korbicz J. (1990): *Parameter identification of two-dimensional distributed systems.* — Int. J. Syst. Sci., Vol. 21, No. 2, pp. 2441–2456.

Uciński D. and Korbicz J. (2001): *Optimal sensor allocation for parameter estimation in distributed systems.* — J. Inverse and Ill-Posed Problems, Vol. 9, No. 3, pp. 301–317.

Uciński D. and Patan M. (2002): *Optimal location of discrete scanning sensors for parameter estimation of distributed systems.* — Proc. 15th IFAC World Congress, Barcelona, Spain, CD-ROM.

Uciński D. and Patan M. (2007): *D-optimal design of a monitoring network for parameter estimation of distributed systems.* — J. Global Optimization, (Accepted).

Uspenskii A.B. and Fedorov V.V. (1975): *Computational Aspects of the Least-Squares Method in the Analysis and Design of Regression Experiments.* — Moscow University Press, (in Russian).

van de Wal M. and de Jager B. (2001): *A review of methods for input/output selection.* — Automatica, Vol. 37, pp. 487–510.

van Loon M. (1994): *Numerical smog prediction, I: The physical and chemical model.* — Technical Report NM-R9411, Centrum voor Wiskunde en Informatica, Amsterdam.

van Loon M. (1995): *Numerical smog prediction, II: Grid refinement and its application to the Dutch smog prediction model.* — Technical Report NM-R9523, Centrum voor Wiskunde en Informatica, Amsterdam.

Vande Wouwer A., Point N., Porteman S. and Remy M. (1999): *On a practical criterion for optimal sensor configuration — Application to a fixed-bed reactor.* — Proc. 14th IFAC World Congress, Beijing, China, Vol. I: Modeling, Identification, Signal Processing II, Adaptive Control, pp. 37–42.

Ventura J.A. and Hearn D.W. (1993): *Restricted simplicial decomposition for convex constrained problems.* — Mathematical Programming, Vol. 59, pp. 71–85.

Vogel C.R. (2002): *Computational Methods for Inverse Problems.* — Frontiers in Applied Mathematics, Philadelphia: Society for Industrial and Applied Mathematics.

von Hohenbalken B. (1977): *Simplicial decomposition in nonlinear programming algorithms.* — Mathematical Programming, Vol. 13, pp. 49–68.

Walter É. and Pronzato L. (1997): *Identification of Parametric Models from Experimental Data.* — Communications and Control Engineering, Berlin: Springer-Verlag.

Zwart H. and Bontsema J. (1997): *An application-driven guide through infinite-dimensional systems theory*, In: G. Bastin and M. Gevers (Eds.), European Control Conference 1997: Plenaries and Mini-Courses, Ottignies/Louvain-la-Neuve: CIACO, Belgium, pp. 289–328.

Chapter 7

USING TIME SERIES APPROXIMATION METHODS IN THE MODELLING OF INDUSTRIAL OBJECTS AND PROCESSES

Wiesław MICZULSKI*, Robert SZULIM*

7.1. Introduction

The measurement of data recorded in measurement systems describes the dynamics of control courses of technological processes or the behavior of objects. The data can be used to build models of those objects or industrial processes.

The initial stage of a knowledge discovery process is a data pre-processing stage, which has a significant influence on the quality of the acquired knowledge (Markowski *et al.*, 2005). In Section 7.3, according to the aforementioned example sets of measurement data (Fig. 7.1), algorithms of data pre-processing are presented. Data thus prepared are used in the process of data exploration. At the stage of data exploration, the employed algorithms should include the following elements (Hand *et al.*, 2001):

- the structure of the model retrieved from the data,
- the estimation function determining the quality of the model derived from the data set,
- methods of search and/or optimization, whose goal is, among other things, to search different structures or results of the model parameters according to estimation function optimization.

Generally speaking, in the field of modelling, descriptive and predictive modelling can be considered (Hand *et al.*, 2001). Descriptive models including, among other things, dividing p-dimension space into groups (analysis of concentration, segmentation), can characterize all the data. The goal of predictive modelling using

* Institute of Electrical Metrology
e-mails: {W.Miczulski, R.Szulim}@ime.uz.zgora.pl

classification or approximation methods is to build a model enabling us to predict the value of one or many variables on the basis of the known values of other variables. In a classification method, the predicted result variables are symbolic variables, and in an approximation method, the result variables have numerical meaning.

In many situations, for the purpose of creating models of industrial processes and technological objects, we can use predictive modelling based on an approximation method, whose models are functions represented by three elements: s – structure, a – set of weights (parameters), and X_i – set of attributes (variables), where $X_i \subseteq X$.

Thus, the formal model notation to set the function approximation has the form

$$y^* = \{s, a, X_i\}\,. \tag{7.1}$$

The discovered function formula should approximate as closely as possible the target function set by the values of the attributes in question. This condition stems directly from the need to predict the values of the chosen dependent attribute. The structure of the function is usually determined by the function approximation method used and is assumed *a priori*, while only the values of the parameters characteristic of this structure are determined. In the case of regression models (Brandt, 1997; Koronacki and Ćwik, 2005; Szydłowski, 1978), the structure will be a parameterized algebraic phrase. For neuron modelling methods, it will be a network of neurons of a particular type (Duch *et al.*, 2000). In function approximation there is also used a supporting vector method (Vapnik, 1995). The aforementioned approximation methods are used wherever the accuracy of the identified relationship function is of great importance. When ease of interpretation of the searched for function relationship description is more important, a method of discovering equations (Washio *et al.*, 1999) is used.

In Section 7.3, there are presented examples of the use of regression models in the approximation of time series presented in Fig. 7.1. Also, the use of an evolution algorithm to optimize the parameter values of the assumed model structure is discussed. The problem of building regression models is widely described in the literature (Hand *et al.*, 2001; Koronacki and Ćwik, 2005). To preserve the cohesion of the entire Chapter 7, information on regression models is presented to a limited extent in Section 7.2.

7.2. Regression models

In data exploration, the building of a regression model is based on measurement data, which are randomly distorted. The general relationship between the observed measurements of the predicting and the result variables can be expressed as

$$\boldsymbol{y} = \boldsymbol{X}\boldsymbol{a} + \boldsymbol{e}, \tag{7.2}$$

where $\boldsymbol{y}$ is the n-dimension result value vector, resulting from the observed n-object measurements, $\boldsymbol{X}$ is the $n \times p + 1$-dimension matrix representing p measurements of predicting variables on n objects, $\boldsymbol{a} = (a_0, \ldots, a_p)$ represents the $(p + 1)$ vector of the searched for values of model parameters, and $\boldsymbol{e} = (e(1), \ldots, e(n))$ is the vector containing the differences between the observed and predicted result values, called residuals.

Model parameter value estimations are conducted in such a way as to minimize the inconsistency $\boldsymbol{e}$. The elements e are connected to each other to achieve a single numerical measure which may be minimized. The most commonly used method of connecting the elements $e(i)$ is summing up their squares. Thus the estimation function is the total square error (Hand *et al.*, 2001),

$$\sum_{i=1}^{n} e(i)^2 = \sum_{i=1}^{n} \left[y(i) - \sum_{j=0}^{p} a_j x_j(i) \right]^2 . \tag{7.3}$$

Searching for the values of the parameter vector $\boldsymbol{a}$ requires the minimization of the total square error. This approach is called the least square method and the parameters values of the vector $\boldsymbol{a}$, which minimize the equation (7.3), are calculated according to the formula (Hand *et al.*, 2001):

$$\boldsymbol{a} = (\boldsymbol{X}^T \boldsymbol{X})^{-1} \boldsymbol{X}^T \boldsymbol{y}. \tag{7.4}$$

In linear regression, the parameters a_j are called regression coefficients. A general regression model can be shown as the dependence

$$\boldsymbol{y}^* = \boldsymbol{X}\boldsymbol{a}. \tag{7.5}$$

Geometrically, the regression model (7.5), called repeated regression, describes a p-dimension hyper-plane embedded in a $(p+1)$-dimension space with the inclination designated by the value of a_j and an intersection at a_0.

Models of this kind of linear structure occupy a significant place in data analysis. It stems from the easily interpretable structure of the model and from the fact that parameter estimation follows directly from an appropriate estimation function.

In the case of time series describing industrial objects and processes, most often we have at our disposal one result variable y, and one predicting variable x, which is the time t. A regression model (regression line) to estimate the predicting values y^* is described by the dependency

$$y^* = a_0 + a_1 t. \tag{7.6}$$

Such models are the oldest, most important and most widely used of all predicting models. One of the reasons for this is their obvious simplicity.

The values of the linear regression coefficients a_0 and a_1 appearing in (7.6) and calculated from (7.4) are described as follows (Szydłowski, 1978):

$$a_1 = \frac{(p+1)\sum_{j=0}^{p} t_j y_j - \sum_{j=0}^{p} t_j \sum_{j=0}^{p} y_j}{(p+1)\sum_{j=0}^{p} t_j^2 - \left(\sum_{j=0}^{p} t_j \right)^2}, \tag{7.7}$$

$$a_0 = \frac{\sum_{j=0}^{p} t_j^2 \sum_{j=0}^{p} y_j - \sum_{j=0}^{p} t_j \sum_{j=0}^{p} t_j y_j}{(p+1)\sum_{j=0}^{p} t_j^2 - \left(\sum_{j=0}^{p} t_j \right)^2}. \tag{7.8}$$

While the standard deviation of the determined coefficients a_1 and a_0 of linear regression are described as the dependencies (Szydłowski, 1978):

$$\sigma_{a_1} = \sqrt{\frac{p+1}{p-1}\frac{\sum_{j=0}^{p}(y_j - a_1 t_j - a_0)^2}{(p+1)\sum_{j=0}^{p} t_j^2 - \left(\sum_{j=0}^{p} t_j\right)^2}}, \tag{7.9}$$

$$\sigma_{a_0} = \frac{\sigma_{a_1}}{p+1}\sum_{j=0}^{p} t_j^2. \tag{7.10}$$

For nonlinear time series, very often there is a need for building a model consisting of segments described by regression lines (linear approximation). Those segments, depending on the needs, can either be connected one by one with each other or the continuity at the ends of the individual linear segments is not required. With an *a priori* assumed approximation error, time series approximation by linear segments can be accomplished with three algorithms (Keogh *et al.*, 2001b): sliding window, top-down, bottom-up.

The sliding window algorithm works by anchoring the left point of a potential segment at the first data point of a time series, then attempting to approximate the data to the right with increasingly longer segments. If, at some point i, the error for the potential segment is greater than the user-specified threshold, then the subsequence from the anchor to $i-1$ is transformed into a segment. The anchor is moved to the location i, and the process is repeated until the entire time series has been transformed into a piecewise linear approximation. The sliding window algorithm is attractive because of its great simplicity, intuitiveness and particularly the fact that it is an online algorithm.

A top-down algorithm works by considering every possible partitioning of data measurement series and splitting it at the best location. Both subsections are then tested to see if their approximation error is below some user-specified threshold. If it is not, the algorithm recursively continues splitting the subsequence until all the segments have approximation errors below the threshold. This algorithm does not work on-line.

The bottom-up algorithm is a natural complement to the top-down algorithm. The algorithm begins by creating the finest possible approximation of the data sequence, so that $n/2$ segments are used to approximate the n-length time series. Next, the merging cost for each pair of adjacent segments is calculated, and the algorithm begins to iteratively merge the lowest cost pair until a stopping criterion is met. When the pair of the adjacent segments i and $i+1$ are merged, the algorithm needs to perform some bookkeeping. First, the cost of merging a new segment with its right neighbour must be calculated. In addition, the cost of merging the $i-1$-th segment with its new longer neighbour must be recalculated. This algorithm does not work on-line.

7.3. Examples of the usage of regression models

7.3.1. Exemplary object and process description

As an example of an industrial process we can consider a copper production technological chain in the Copper Works *Głogów*, Poland (Szulim, 2004). A very important object of that chain is an electrical furnace which is responsible for copper slag reduction from a level of 15% to about 0.5%. The furnace works on a cyclic basis. Every production period consists of three phases: loading, reduction, tapping. During the loading, the furnace is filled with slag. During the reduction, control tappings are made in order to determine the parameters of the alloy inside the furnace. After achieving certain parameters, the furnace is ready to be unloaded. This is an example of a process controlled by competent operators who have to take decisions under the conditions of uncertainty of information about the process. It is impossible to measure the important parameters of the process in a continuous way because of the difficult measurement conditions. Furnace operators are supported by SCADA (*Supervisory Control and Data Acquisition*) computer monitoring systems. These kinds of systems do not have the ability to advise how to control the process to achieve a production target in a particular time and for particular conditions. This kind of task can be formulated for an expert system which includes a knowledge base built using the knowledge discovered from the measurement data.

Data registered in the SCADA measurement system can be divided into a static and a dynamic part. The static part represents the measurement data of the cycle input and output parameter values, such as the amount and features of the cycle ingredients. The control process is described by the values of the measurement data of particular physical quantities stored once every specified period of time, e.g. one minute. Those quantities can be pressure courses, temperatures, electrical quantities, e.g. a time course of real electrical power supplied to the furnace (Fig. 7.1(a)). The model built with the use of the measurement data can constitute the basis for building a knowledge base in the form of a database of examples of industrial process control time series. For finding similar examples in the database, a case based reasoning paradigm can be used. Such an approach calls for devising a special way to represent the examples in the database, building an appropriate similarity measure and tuning the parameters of the system.

An example of an object can be a generator producing the Radio Standard Frequency (RSF) signal of 225 kHz. This signal is simultaneously used as a carrier frequency for Program 1 long wave broadcasts from Polish Radio S.A. The value of this frequency is controlled by the Laboratory of Time and Frequency of the Central Office of Measures (Poland). A block diagram of the measurement system realizing the task mentioned above is presented in Fig. 7.2. The quantity measured by the timer is the phase time that is a momentary RSF signal phase determined with respect to the reference signal and expressed in terms of time units. So far, the registration of the phase time measurement values has been realized by means of a paper tape recorder. Example results of the phase time measurement for a 24-hour period are presented in Fig. 7.1(b). They are the basis of "manual" calculations of two indicators of the radio standard frequency deviation from the nominal value of (Miczulski and Czubla, 2006):

– 24-hour RSF frequency average relative deviation from the nominal value (w_{ds}), calculated from the formula (7.11),

– short-term (lasting longer than half an hour) relative RSF frequency deviations from the nominal value (w_k), calculated from the formula (7.12).

$$w_{ds} = -\frac{tf(t_2) - tf(t_1) - \sum_{i=1}^{n} \Delta tf_{si}(t) - \sum_{j=1}^{m} \Delta tf_{kj}(t)}{t_2 - t_1 - \sum_{j=1}^{m} \Delta t_{kj}}, \tag{7.11}$$

$$w_k = \frac{-\Delta tf_z(t)}{t_{z2} - t_{z1}} = -\frac{tf_z(t_{z2}) - tf_z(t_{z1}) - \Delta tf_{zs}(t)}{t_{z2} - t_{z1}}, \tag{7.12}$$

where $tf(t_1)$, $tf(t_2)$ represents the phase time at the beginning and end of a 24-hour period, $\Delta tf_{si}(t) = tf_{si}(t_{s2i}) - tf_{si}(t_{s1i})$ is the phase time jump calculated before and after the i-th jump – considered only when it is clearly observable and of a fixed character, $\Delta tf_{kj}(t) = tf_{kj}(t_{k2j}) - tf_{kj}(t_{k1j})$ is a short-term change of the phase time calculated on the basis of reading the phase time at the beginning and end of the j-th short-term change, and considered only when it is of a fixed character, clearly observable in comparison with an average phase time course during the analyzed period, and not calculated separately as a jump change or a change of the phase time $\Delta tf_z(t)$ considered while calculating w_k. Moreover, $\Delta t_{kj} = t_{k2j} - t_{k1j}$ is the ignored time of a short-term change of the phase time, $tf_z(t_{z1})$, $tf_z(t_{z2})$ represents the phase time at the beginning and end of short-term (lasting more than half an hour) RSF deviation from the frequency nominal value, and $\Delta tf_{zs}(t)$ is the phase time jump.

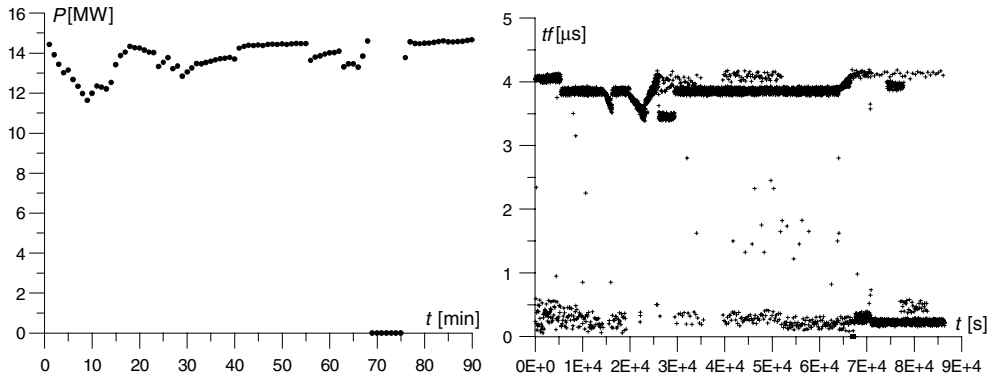


Fig. 7.1. Examples of time series for electrical furnace power supply control (a), phase time measurement results characterizing RSF generator stability (b)

The automation of diagnosing the process of generator work requires connecting (via the General Purpose Interface Bus (GPIB) interface) a digital time meter with a computer (Fig. 7.2). Using the results of the phase time measurement stored in a computer memory, a model to calculate the indicators w_{ds} and w_k is built.

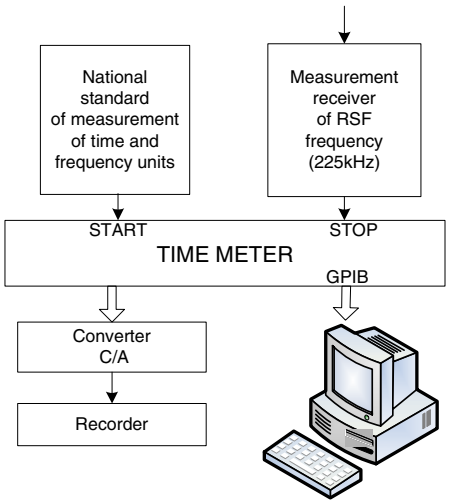


Fig. 7.2. Block diagram of the RSF monitoring module

7.3.2. Knowledge acquisition from measurement data of complex technological process

In many situations, controlling complex technological processes is conducted on the basis of the comparison of separate realizations of the process. Such a comparison may concern searching for similarities between the time series of the registered physical quantities of the current process and archival time series. Measurement data which are the basis for building archival time series are pre-processed. Data pre-processing tasks can include cleaning the data, transforming attributes and selecting relevant attributes.

During similarity estimation, cases should be taken into account when series are characterized by various number of measurements, the shape of the series may be shifted on the time or value axes, or measurement results are burdened with measurement errors and noise (Fig. 7.3).

In real industrial processes, i.e. in the process of copper slag reduction, time series are much more difficult to compare between each other. Examples of such series are shown in Fig. 7.4. These examples illustrate difficulty in comparing and searching for similarities between them in a numerical way. Therefore, it is necessary to use an information reduction process, which may be carried out by many methods (Keogh *et al.*, 2001b; Moczulski and Szulim, 2004). In this case, a piecewise linear approximation using a bottom-up algorithm was employed. In Fig. 7.5 are presented the results of an approximation of time series of real power supplied to an electrical furnace for copper slag reduction cycles (Fig. 7.4(a)). The first course is approximated by six segments, while the second – by five.

A direct comparison of series of lines approximating time series does not yield good results, because there occur situations when the number of lines is different and the lines differ significantly. In order to compare the lines, a special method with a fuzzy description of the lines was elaborated. Every line approximating a time

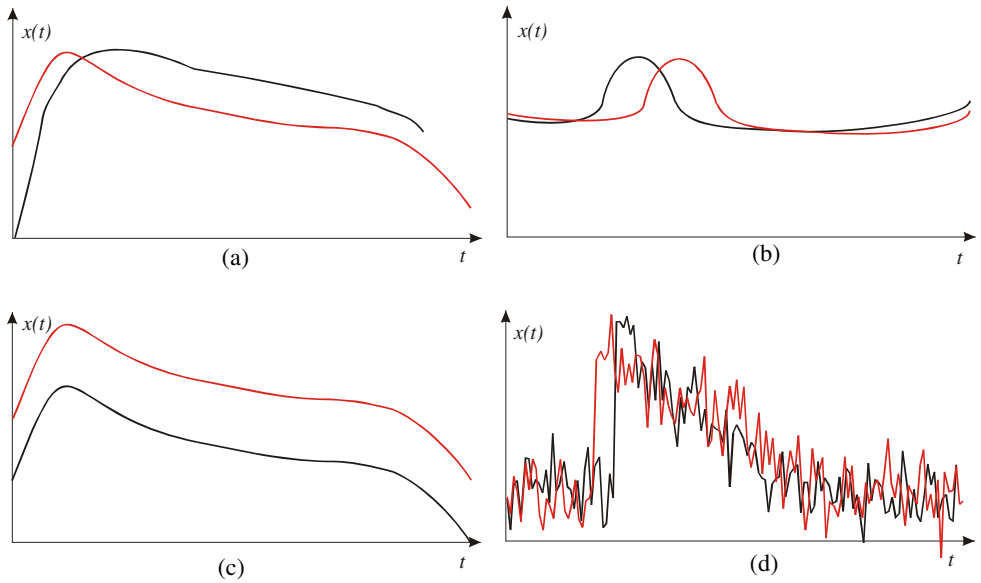


Fig. 7.3. Examples of time series with various numbers of measurements (a), shape of series shifted on time (c), and value axes (b), burdened with noise (d)

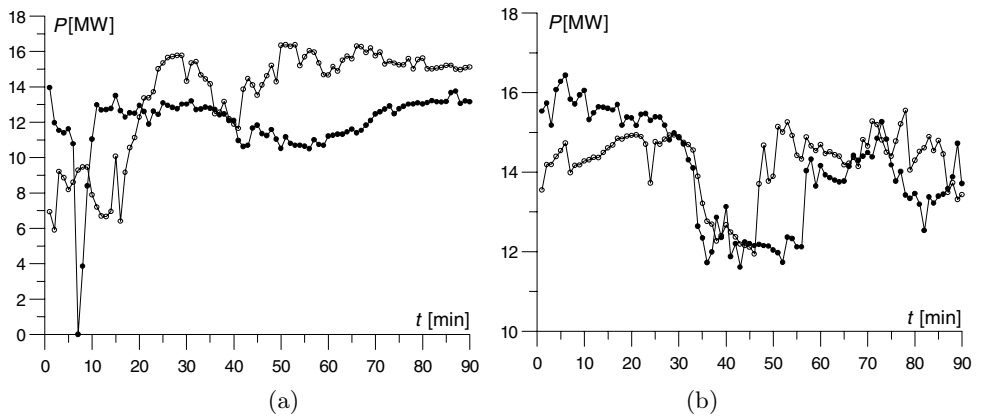


Fig. 7.4. Examples of time series of real power supplied to an electrical furnace in four different cycles of the copper slag reduction process

series is represented by a dynamic fuzzy statement further called an event, which is represented by the following attributes: *Type, Duration*, and *Initial Value*. The *Type* attribute signifies the kind of event, that is, the inclination of the line. The definition of fuzzy sets for this attribute takes also into account the character of the event. It is possible that both positive and negative values of the line inclination (increase and decrease) may occur. The *Duration* attribute describes the time in which the event occurs. The *Initial Value* attribute defines the initial level of a physical quantity (e.g.

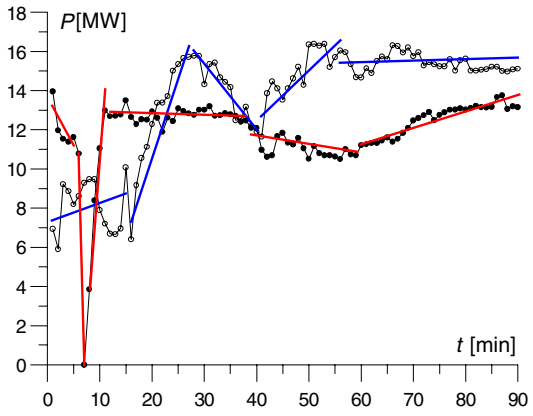


Fig. 7.5. Example of linear approximation of the time series from Fig. 7.4(a)

power) for a line considered as an event. For every attribute, special trapezoid fuzzy sets were defined. In the database, vectors of the values of membership functions of individual fuzzy sets are stored (Szulim, 2004).

Searching for similar examples in a database requires the use of the so-called similarity measure. In many cases it is necessary to design a special similarity measure oriented onto the state of the compared data. Data describing the examples have dynamic and static forms. Two similarity measures were designed. One for static data (data representing process the input and output) and one for control courses described by means of line series. As a control course, only one variable course is considered – the amount of energy supplied to the furnace.

The total similarity of two realizations represented by a list of events singled out in two time series is determined as a weighted average of similarities of pairs of events (Fig. 7.6). The weight is defined on the basis of the duration of a longer line. The similarity of a pair of events is defined as the product of partial similarities of particular attributes describing a given pair of events approximating a series. The similarity measure uses special similarity matrices of fuzzy sets, which define the influence of certain line features on their similarity. It follows from these features that courses approximated by longer lines have a greater influence on the similarity (Szulim, 2004).

The system of representation of examples and the designed similarity measure possess many parameters. They are, among other things the borders of fuzzy sets, similarity matrices of fuzzy sets, and the threshold value of the piecewise linear approximation error. The values of these parameters have a decisive influence on the correctness of the work of an expert system. The determination of these values in an intuitive way does not allow achieving sufficiently good results. It was necessary to elaborate a method of automatic parameter value selection. For this purpose, an evolutional algorithm was used. A system capable of selecting parameter values automatically was built. A special way of parameter representation in a population of chromosomes was designed together with special genetic operators of crossover, mutation and parameter fixing (Moczulski and Szulim, 2004; Szulim, 2004; Szulim and

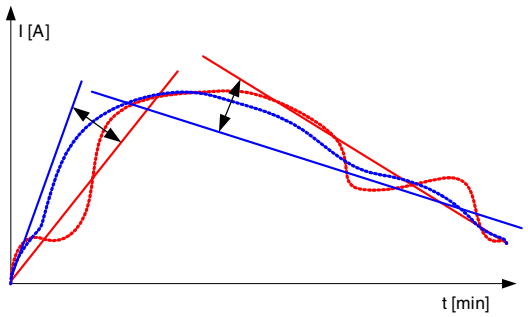


Fig. 7.6. Calculation of similarity of control courses

Moczulski, 2003; 2004). In the process of evolutional tuning, about 3000 chromosomes were processed. An improvement in the quality of the work of the system of representation of examples was achieved. The result of the work of the evolutional algorithm is shown in Fig. 7.7. For two example time series approximations, before employing an evolutional algorithm, a similarity value of 0.5 was obtained (Fig. 7.7(a)). After using evolutional tuning, this value increased to 1 (in a fuzzy sense) – Fig. 7.7(b). The way of the approximation of time series also changed because the values of the error threshold of control course approximation had changed. Here the evolutional algorithm realized a component of a data exploration algorithm concerning searching and optimization methods. The modification of the regression factor values and the number of lines approximating time series followed.

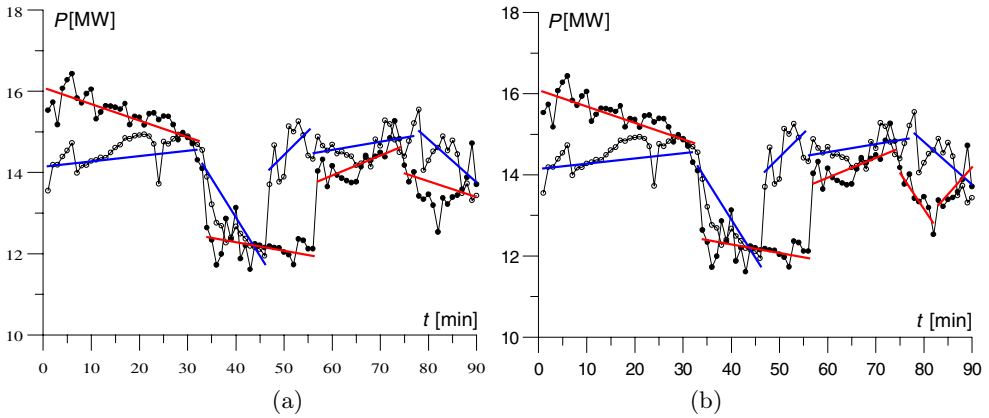


Fig. 7.7. Similarity of two realizations before (a) and after (b) parameter tuning

Due to time-consuming calculations and their great complexity, the system was prepared to carry out calculations in a distributed way, simultaneously on many computers linked up into a network. An application was also designed for remote monitoring of the calculation progress by means of the Internet.

The verification of the correctness of the work of the aforementioned method of obtaining knowledge from archival data to allow conducting a complex technological process consisted in checking the following hypothesis: for a given example in a set of

historical realizations, similar realizations can be found in terms of the input and a control course. For examples similar in terms of the input and control course, output data should also be similar. The hypothesis is similar to the assumptions made by the personnel operating a given object. For incorrect parameter values, the system indicates too many or too few similar process realizations. In order to conduct the verification, a special program module carrying out similarity tests of the recorded cycles according to the described concept was built (Fig. 7.8). The verification was conducted using data from a real object – an electrical furnace working in a copper works.

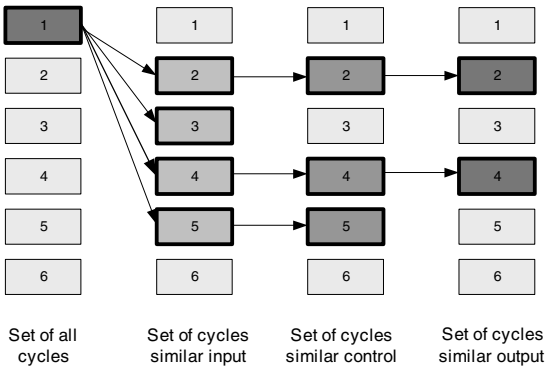


Fig. 7.8. Searching for similar realizations of the process

The knowledge base built in this way can be used to build an expert system. The system can search for examples similar to a process currently being conducted. For the presented example, it is possible to find one or many similar examples for a given similarity threshold (Fig. 7.9). To realize this, sequential browsing of the database records is necessary. Each example is compared with a current one and their similarity is calculated using a similarity measure. The numbers of examples meeting the similarity criteria can be stored in the memory and sent for further processing to concluding procedures.

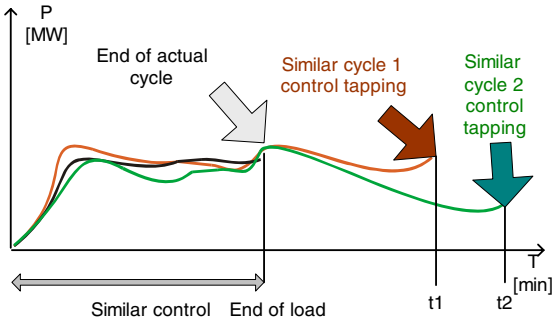


Fig. 7.9. Searching of similar control courses

7.3.3. Diagnostics of a standard radio frequency generator

The way of generating and keeping approximate fixed values of the RSF frequency leads to measurement results of the phase time, which can change within one period $T_{RSF} = 4.44(4)\,\mu$s. Among all measurement results of the phase time tf for one 24-hour period, two groups of results can be singled out:

- a group of results determining the tendency of RSF frequency changes,
- a group of results connected with short-term interferences, usually concerning the quality of the emitted and received RSF signal (Fig. 7.1).

Taking this into account as well as

- the conditions defined during the estimation of jump and short-term changes of the phase time, and
- the specificity of the RSF frequency value regulation,

an algorithm was proposed to acquire diagnostic knowledge connected with the calculation of the w_{ds} and w_k factors (Fig. 7.10(a)). The initial stage of knowledge discovery, called data pre-processing, is connected with loading the computer memory with phase time measurement data recorded in a measurement system for a 24-hour period, as well as data grouping and filtering.

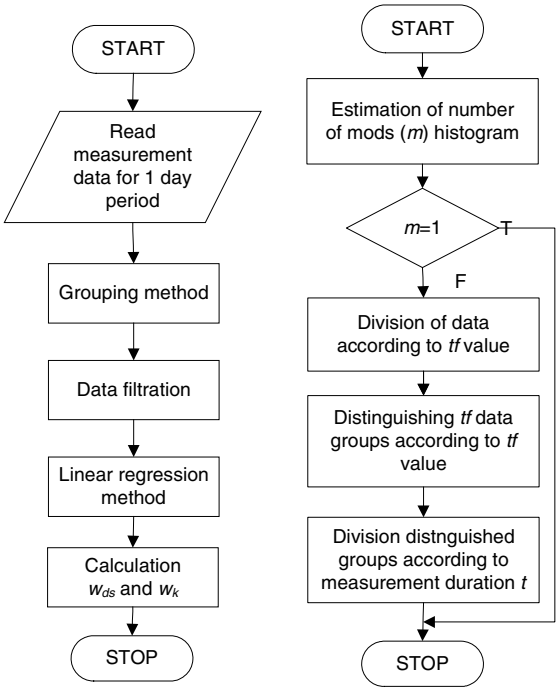


Fig. 7.10. Diagnostic knowledge acquiring algorithm (a), algorithm illustrating grouping method work (b)

In order to estimate the RSF frequency change tendency, a process of measurement data division into groups is done (grouping method – Fig. 7.10(b)). The selection of a proper grouping algorithm is connected with the competent use of the knowledge of the problem described by a set of data. In a general case, the division of data should be conducted in such a way that the data in one group are similar as much as possible to each other and the data from different groups should differ from each other as much as possible. Generally, data division can be conducted according to a fuzzy or a sharp division. Considering specific data as well as the w_{ds} and w_k estimation, the way of sharp grouping was chosen, whose goal is the division of the measurement phase time stored in the vector $\boldsymbol{TF}$ into the m TFG group in such a way that the set of all groups includes all the data (Rutkowski, 2005):

$$\bigcup_{i=1}^{m} TFG_i = \boldsymbol{TF}, \tag{7.13}$$

the groups were disjoint,

$$TFG_i \cap TFG_j = \emptyset, \quad 1 \leq i \neq j \leq m, \tag{7.14}$$

and none of them were empty nor contained a whole set of data,

$$\emptyset \subset TFG_i \subset \boldsymbol{TF}, \quad 1 \leq i \leq m. \tag{7.15}$$

In the grouping method applied, the data division should occur with regard to the phase time values and the measurement time. The basis of algorithm work with regard to the phase time values is a measurement data histogram for one 24-hour period (Fig. 7.11), on the basis of which the number (m) of groups is estimated. In the case of the occurrence of one histogram maximum ($m = 1$), the procedure of grouping method is omitted.

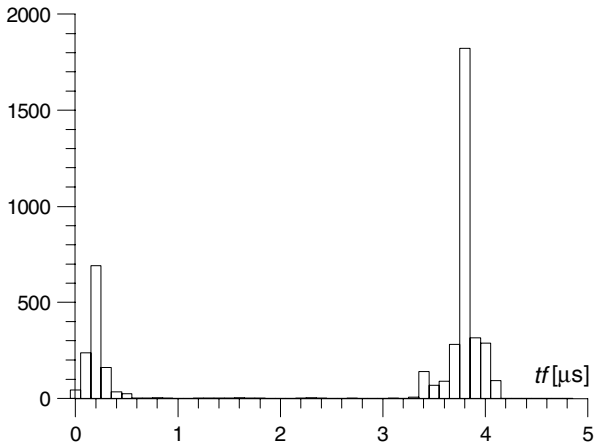


Fig. 7.11. Histogram of the measurement data from Fig. 7.1(b)

The borders of the division into the groups TFG_i of the data stored in the vector $\boldsymbol{TF}$ are defined on the basis of a histogram with an assumed level of the number of occurrences of the phase time measurement results. In the example mentioned, all the data of the phase time measurements results for one 24-hour period were divided into four groups according to the phase time. As a result of grouping procedure work, the division borders were defined as follows:

$$\begin{aligned} &\text{group } TFG_1: && 3.4 \leq tf \leq 4.5,\\ &\text{group } TFG_2: && 0.1 \leq tf \leq 0.4,\\ &\text{group } TFG_3: && 0.4 < tf < 3.4,\\ &\text{group } TFG_4: && 0 < tf < 0.1. \end{aligned}$$

Groups which have phase time measurement results of values equal to the maximum occurrence (based on a histogram) are groups defining the tendency of the RSF frequency changes. In the example considered, these groups are TFG_1 and TFG_2 (Fig. 7.12).

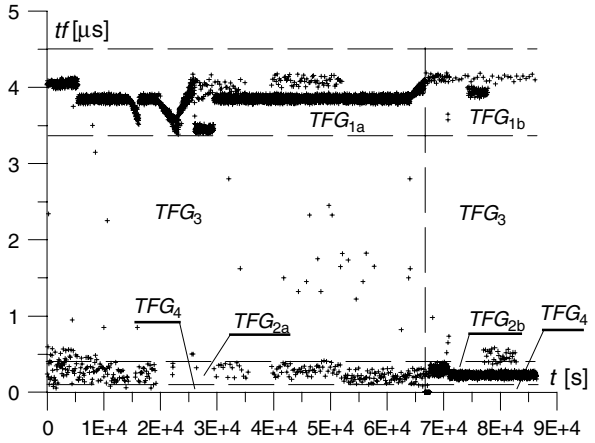


Fig. 7.12. Example set of phase time measurement results for one 24-hour period and its division into groups as a result of grouping method work

In another step of division algorithm work, a division into groups (defining the tendency of the RSF frequency changes) with regard to the measurement time t follows. The numbers of the results of the measurement of the phase time between the singled out groups are compared in a movable time window. The result of the algorithm work is the determination of the time defining the division border of these groups. In this way, in the example mentioned, the division of the TFG_1 group into the groups TFG_{1a} and TFG_{1b}, as well as the TFG_2 group into the groups TFG_{2a} and TFG_{2b} occurred (Fig. 7.12). The division border is the time $t = 67600\,\text{s}$. The groups TFG_{1a} and TFG_{2b} define conclusively the tendency of RSF frequency changes. The remaining groups (TFG_{1b}, TFG_{2a}, TFG_3 and TFG_4) include tf measurement results burdened with errors resulting from short term interferences.

The quality of the discovered knowledge is decided by the quality of the data used in the process of knowledge acquisition. The generally adopted principles of data preparation require data cleaning. According to the adopted procedure of estimation of w_{ds} and w_k, the phase time measurement results qualified to the TFG_3 and TFG_4 groups cannot be removed from the measurement data groups. Phase time values for these measurement results should correspond to the values close to the phase time defining the tendency of the RSF frequency changes in those time intervals. Data filtering is responsible for that.

In the first step of the data filtering procedure, groups of measurement data (TFG_{1a} and TFG_{2b}), which define the tendency of phase time changes for one 24-hour period, are chosen. The selected groups of measurement data are subject to an analysis of time series (Brandt, 1997), in which from the formula (7.16) values of the average movable phase time $srtf_i$ are estimated, for $2k+1$ consecutive measurement points ($k=5$):

$$srtf_i = \frac{1}{2k+1} \sum_{j=i-k}^{i+k} tf_j. \tag{7.16}$$

Further in the procedure of data filtration all the data from the groups TFG_{1b}, TFG_{2a}, TFG_3 and TFG_4 are assigned calculated average movable values from appropriate moments of time. Also the measurement data from the groups TFG_{1a} and TFG_{2b}, which significantly differ from the movable phase time average value, are assigned the average movable phase time calculated values from appropriate moments of time. In that way, the corrected values of the phase time (tf_s) for the analyzed one 24-hour period, which do not include single values, significantly differ from the average movable phase time (Fig. 7.13 – set of points marked (a)). For the obtained set of the phase time corrected values, a time series analysis was applied again to calculate the movable average phase time values (Fig. 7.13 – curve (b)) according to the formula (7.16).

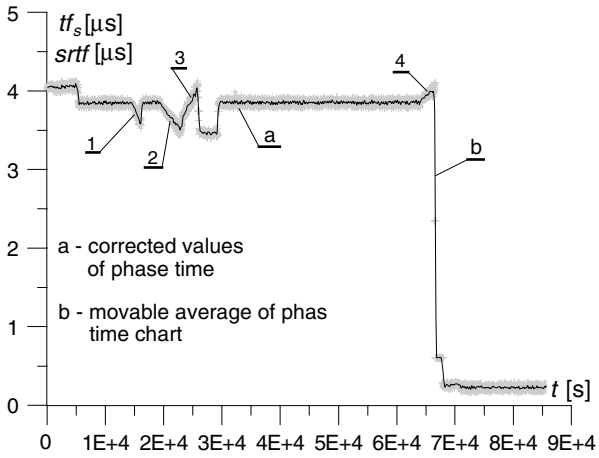


Fig. 7.13. Phase time corrected values and a phase time movable average chart

Table 7.1. Results of w_k calculations

n	w_{kn}	t_{z2} [s]	t_{z1} [s]	Δt_z [min]
1	$1.7 \cdot 10^{-10}$	$1.6 \cdot 10^4$	$1.44 \cdot 10^4$	27
2	$0.9 \cdot 10^{-10}$	$2.28 \cdot 10^4$	$1.95 \cdot 10^4$	55
3	$\mathbf{-1.6 \cdot 10^{-10}}$	$\mathbf{2.58 \cdot 10^4}$	$\mathbf{2.29 \cdot 10^4}$	**48**
4	$-0.5 \cdot 10^{-10}$	$6.63 \cdot 10^4$	$6.42 \cdot 10^4$	35

For data exploration, a linear regression model of time series approximation was applied. For every characteristic fragment of the phase time movable average course (Fig. 7.13 – curve (b)), determined in a previous step of the algorithm and containing n values of t_i and corresponding with them the values of rtf_i, the values of the linear regression factors a_0 and a_1 are calculated according to the formulae (7.7) and (7.8).

The calculated values of linear regression factors permit the determination of specific values of the phase time for the corresponding values of the time t needed for determining from the dependencies (7.11) and (7.12) the factors of the standard radio frequency deviation from the nominal value.

In Fig. 7.13 there are marked four short-term RSF frequency deviations from the nominal value discovered by the algorithm. In Table 7.1, the results of the calculation of four relative short term RSF frequency deviations from the nominal value (w_k) for a given set of the phase time values are presented. According to the adopted principles of the w_k calculation, only the result for $n = 3$ can be treated as a relative RSF frequency deviation from the nominal value. The rest of the obtained results are taken into account while calculating the value of an average relative 24-hour RSF frequency deviation from the nominal value as the omitted short-term changes of the phase time $\Delta tf_k(t)$ in the Δt_k time. The value w_{ds} calculated by the presented algorithm for the considered example is $-0.1 \cdot 10^{-10}$.

For every linear segment approximating the time series of the phase time corrected values (Fig. 7.13), from the dependencies (7.9) and (7.10), standard deviations of the linear regression factors a_0 and a_1 were calculated. The obtained values σ_{a0} and σ_{a1} were the basis for calculating the uncertainty of the determination of the factors w_{ds} and w_k. The calculated uncertainty values at a confidence level of 0.95 are as follows:

$$U_{w_{ds}} = 0.006 \cdot 10^{-10}, \quad U_{w_k} = 0.12 \cdot 10^{-10}.$$

7.4. Summary

Regression models with a linear structure are important in data analysis. They are the most widespread of all predicting models. This stems from the easily interpretable structure of models, and the estimation of the model parameters follows directly from the estimation function. This fact is confirmed by the two quoted examples of the employment of a regression model for modelling and industrial object and process.

In the quoted examples, the time series characterizing the industrial object and process were approximated by linear segments. This was a basis for automatic creation

of a knowledge base about the copper reduction process in an electrical furnace and enabled full automation of the diagnosis of RSF generator work. Simulation research was conducted for data coming from a real industrial object and process. The results confirmed the assumed concept of building models.

The variety of time series and a great complexity of input data presented in both examples required devising special procedures. For an industrial process, fuzzy set borders, fuzzy set similarity matrices and a threshold value of error approximation were defined among other things. The values of these parameters were calculated by an evolution algorithm. In turn, in the process of initial input data preparation for further analysis characterizing generator work, it was necessary to employ grouping methods and an analysis of time series.

References

Brandt S. (1997): *Statistical and Computational Methods in Data Analysis*. — New York: Springer Verlag.

Duch W., Korbicz J., Rutkowski L. and Tadeusiewicz R. (Eds.) (2000): *Neural Networks*. — series on Biocybernetics and Biomedical Engineering, Vol. 6, Warsaw: Akademicka Oficyna Wydawnicza EXIT, (in Polish).

Hand D., Mannila H. and Smyth P. (2001): *Principles of Data Mining*. — Massachusetts Institute of Technology Press.

Keogh E., Chakrabarti K., Pazzani M. and Mehrotra S. (2001a): *Dimensionality reduction for fast similarity search in large time series databases*. — Knowledge and Information Systems – An Int. J., Vol. 3, No. 3, pp. 263–286.

Keogh E., Chu S., Hart D. and Pazzani M. (2001b): *An online algorithm for segmenting time series*. — Proc. IEEE Inter. Conf. *Data Mining*, San Jose, USA, pp. 289–296.

Koronacki J. and Ćwik J. (2005): *Statistical Self Learning Systems*. — Warsaw: Wydawnictwa Naukowo-Techniczne, WNT, (in Polish).

Markowski A., Miczulski W. and Szulim R. (2005): *On quality of measurement data in the process of knowledge acquisition*. — Proc. 14-th IMEKO Symp. *New Technologies in Measurement and Instrumentation*, Gdynia, Poland, pp. 171–174.

Miczulski W. and Czubla A. (2006): *Algorithm of calculation of standard radio frequency deviation from its nominal value*. — Pomiary, Automatyka, Kontrola, PAK, No. 6, pp. 39–41, (in Polish).

Moczulski W. and Szulim R. (2004): *On case-based control of dynamic industrial processes with the use of fuzzy representation*. — Engineering Applications of Artificial Intelligence, EAAI, Vol. 17, pp. 371–381.

Rutkowski L. (2005): *Methods and Techniques of Artificial Intelligence*. — Warsaw: Państwowe Wydawnictwo Naukowe, PWN, (in Polish).

Szulim R. (2004): *A Method of Mining Knowledge to Aid a Control of Complex Industrial Processes*. — Ph.D. dissertation, University of Zielona Góra, Faculty of Electrical Engineering, Computer Science and Telecommunications, (in Polish).

Szulim R. and Moczulski W. (2003): *An evolutionary algorithm for improving fuzzy modeller of industrial processes*. — Proc. Conf. *Methods of Artificial Intelligence, AI-METH*, Gliwice, Poland, CD-ROM.

Szulim R. and Moczulski W. (2004): *A method of mining knowledge to aid control of complex industrial processes.* — Proc. Conf. *Methods of Artificial Intelligence, AI-METH*, Gliwice, Poland, pp. 149–150.

Szydłowski H. (1978.): *Measurement Theory.* — Warsaw: Państwowe Wydawnictwo Naukowe, PWN, (in Polish).

Vapnik V. (1995): *The Nature of Statistical Learning Theory.* — New York: Springer-Verlag.

Washio T., Motoda H. and Yuji N. (1999): *Discovering admissible model equations from observed data based on scale-types and identity constrains.* — Proc. 17-th Int. Joint Conf. *Artificial Intelligence*, Stockholm, Sweden, pp. 772–779.

Chapter 8

ANALYTICAL METHODS AND ARTIFICIAL NEURAL NETWORKS IN FAULT DIAGNOSIS AND MODELLING OF NON-LINEAR SYSTEMS

Józef KORBICZ*, Marcin WITCZAK*, Krzysztof PATAN*
Andrzej JANCZAK*, Marcin MRUGALSKI*

8.1. Introduction

A continuous increase in the complexity, efficiency, and reliability of modern industrial systems necessitates a continuous development in the control and fault diagnosis theory and practice (Blanke *et al.*, 2003; Calado *et al.*, 2006; Chen and Patton, 1999; Isermann, 2006; Korbicz, 2004; Korbicz *et al.*, 2004; Kościelny, 2001; Patton *et al.*, 2006). These requirements extend beyond the normally accepted safety-critical systems of nuclear reactors, chemical plants or aircrafts, to new systems such as autonomous vehicles or fast rail systems. An early detection and maintenance of faults can help avoid system shutdown, breakdowns and even catastrophes involving human fatalities and material damage. A modern control system that is able to tackle such a challenging problem is presented in Fig. 8.1 (Witczak, 2006a). As can be observed, the controlled system is the main part of the scheme, and it is composed of actuators, process dynamics and sensors. Each of these parts is affected by the so-called unknown inputs, which can be perceived as process and measurement noise as well as external disturbances acting on the system. When model-based control and analytical redundancy-based fault diagnosis are utilised (Blanke *et al.*, 2003; Chen and Patton, 1999; Korbicz *et al.*, 2004), then the unknown input can also be extended by model uncertainty, i.e., the mismatch between the model and the system being considered.

The system may also be affected by faults. A fault can generally be defined as an unpermitted deviation of at least one characteristic property or parameter of the

* Institute of Control and Computation Engineering
e-mails: {j.korbicz, m.witczak, k.patan, a.janczak, m.mrugalski}@issi.uz.zgora.pl

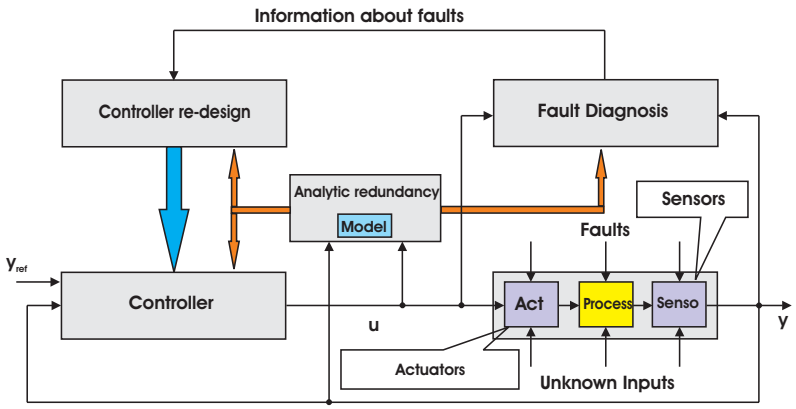


Fig. 8.1. Modern control system

system from the normal condition, e.g., a sensor malfunction. All the unexpected variations that tend to degrade the overall performance of a system can also be interpreted as faults. Contrary to the term *failure*, which suggests a complete breakdown of the system, the term *fault* is used to denote a malfunction rather than a catastrophe. Indeed, *failure* can be defined as a permanent interruption of the system ability to perform a required function under specified operating conditions. This distinction is clearly illustrated in Fig. 8.2. Since a system can be split into three parts (Fig. 8.1),

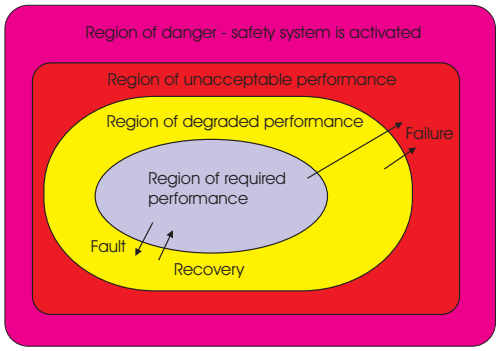


Fig. 8.2. Regions of system performance

i.e., actuators, the process, and sensors, such a decomposition leads directly to three classes of faults:

- *Actuator faults*, which can be viewed as any malfunction of the equipment that actuates the system, e.g., a malfunction of the electro-mechanical actuator for a diesel engine (Blanke *et al.*, 1994);
- *Process faults* (or component faults), which occur when some changes in the system make the dynamic relation invalid, e.g., a leak in a tank in a two-tank system;
- *Sensor faults*, which can be viewed as serious measurements variations.

The role of the fault diagnosis part is to monitor the behaviour of the system and to provide all possible information regarding the abnormal functioning of its components. As a result, the overall task of fault diagnosis consists of three subtasks (Chen and Patton, 1999):

Fault detection: to make a decision regarding the system stage – either that something is wrong or that everything works under the normal conditions;

Fault isolation: to determine the location of the fault, e.g., which sensor or actuator is faulty;

Fault identification: to determine the size and type or nature of the fault.

However, from the practical viewpoint, to pursue a complete fault diagnosis, the following three steps have to be realised (Frank and Ding, 1997):

Residual generation: generation of the signals that reflect the fault. Typically, the residual is defined as a difference between the outputs of the system and its estimate obtained with the mathematical model;

Residual evaluation: logical decision making on the time of occurrence and the location of faults;

Fault identification: determination of the type of a fault, its size and cause.

The knowledge resulting from these steps is then provided to the controller re-design part, which is responsible for changing the control law in such a way as to maintain the required system performance. Thus, the scheme presented in Fig. 8.1 can be perceived as a fault-tolerant one.

If residuals are properly generated, then fault detection becomes a relatively easy task. Since without fault detection it is impossible to perform fault isolation and, consequently, fault identification, all efforts regarding the improvement of residual generation seem to be justified. There have been many developments in model-based fault detection since the beginning of the 1970s, regarding both the theoretical context and the applicability to real systems (see (Chen and Patton, 1999; Korbicz *et al.*, 2002; 2004; Patton *et al.*, 2000) for a survey). Generally, the most popular classical approaches can be split into three categories, i.e., parameter estimation, parity relation and observer-based fault diagnosis. All of them, in one way or another, employ a mathematical system description to generate the residual signal.

Irrespective of the identification metod used, there is always the problem of model uncertainty, i.e., the model-reality mismatch. Thus, the better the model used to represent system behaviour, the better the chance of improving the reliability and performance in diagnosing faults. Indeed, disturbances as well as model uncertainty are inevitable in industrial systems, and hence there exists a pressure creating the need for robustness in fault diagnosis systems. This robustness requirement is usually achieved at the fault detection stage, i.e., the problem is to develop residual generators which should be insensitive (as far as possible) to model uncertainty and real disturbances acting on a system while remaining sensitive to faults. In one way or another, all the above-mentioned approaches can realise this requirement for linear systems.

Taking into account the above conditions, a large amount of knowledge on designing robust fault diagnosis systems has been accumulated through the literature since the beginning of the 1980s. For a comprehensive survey regarding such techniques, the reader is referred to the excellent monographs (Chen and Patton, 1999; Gertler, 1998; Korbicz *et al.*, 2004; Patton *et al.*, 2000).

As can be observed in the literature (Chen and Patton, 1999; Gertler, 1998; Korbicz *et al.*, 2004; Patton *et al.*, 2000), the most common approach to robust fault diagnosis is to use robust observers. This is mainly because of the fact that the theory of robust observers is relatively well developed in the control engineering literature. Challenging design problems arise regularly in modern fault diagnosis systems.

Unfortunately, the classical analytical techniques often cannot provide acceptable solutions to all problems that arise regularly in modern fault diagnosis. Indeed, as can be observed in the literature (Chen and Patton, 1999; Korbicz *et al.*, 2004; Zolghadri *et al.*, 1996), the design complexity of most observers for non-linear systems does not encourage engineers to apply those in practice. Another fact is that the application of observers is limited by the need for non-linear state-space models of the system being considered, which is usually a serious problem in complex industrial systems. This explains why most of the examples considered in the literature are devoted to simulated or laboratory systems, e.g., the well-known two- or three-tank system, the inverted pendulum, etc. (Chen and Patton, 1999; Korbicz *et al.*, 2004; Zolghadri *et al.*, 1996).

The above problems contribute to the rapid development of soft computing-based FDI (Fault Detection and Isolation) (Korbicz *et al.*, 2004; Ruano, 2005). Generally, the most popular soft computing techniques that are used within the FDI framework can be divided into three groups: neural networks, fuzzy logic-based techniques and evolutionary algorithms. There are, of course, many combinations of such approaches, e.g., neuro-fuzzy systems (Korbicz, 2006; Korbicz *et al.*, 2004; Kowal, 2005; Patton *et al.*, 2005). Another popular strategy boils down to integrating analytical and soft computing techniques, e.g., evolutionary algorithms and observers (Witczak and Korbicz, 2004; Witczak *et al.*, 2002) or neuro-fuzzy systems and observers (Uppal *et al.*, 2006).

Taking into account the above discussion, the main objective of this chapter is to present recent developments in modern fault diagnosis with non-linear observers and neural networks. In particular, the chapter is organised as follows: Section 8.2 outlines the problem of observer-based robust fault diagnosis and presents two different observer structures that can be employed for non-linear systems. Section 8.3 presents four alternative neural network-based approaches that can be used to settle the fault diagnosis problem when the mathematical state-space model is not available. The subsequent Section 8.4 presents two applications of the approaches described in the preceding sections. In particular, the first example is devoted to neural network-based modelling of a DC motor while the second one concerns observer-based fault detection of an induction motor. Finally, the last part concludes the chapter.

8.2. Observer-based FDI

The basic idea underlying observer-based (or filter-based, in the stochastic case) approaches to fault detection is to obtain the estimates of certain measured and/or unmeasured signals. Then, in the most usual case, the estimates of the measured signals are compared with their originals, i.e., the difference between the original signal and its estimate is used to form a residual signal $\boldsymbol{r}_k = \boldsymbol{y}_k - \hat{\boldsymbol{y}}_k$ (Fig. 8.3). To tackle this problem, many different observers (or filters) can be employed, e.g., Luenberger observers, Kalman filters, etc. From the above discussion, it is clear that the main objective is the estimation of system outputs while the estimation of the entire state vector $\boldsymbol{x}$ is unnecessary. Since reduced-order observers can be employed, state estimation is significantly facilitated. On the other hand, to provide an additional freedom to achieve the required diagnostic performance, the observer order is usually larger than the possible minimum one.

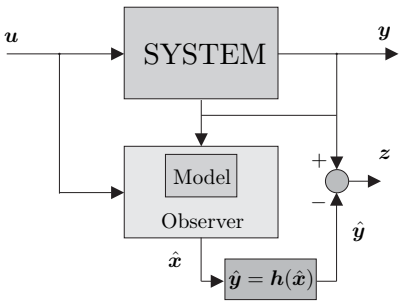


Fig. 8.3. Principle of observer-based residual generation

The admiration for observer-based fault detection schemes is caused by the still increasing popularity of state-space models as well as the wide usage of observers in modern control theory and applications. Due to such conditions, the theory of observers (or filters) seems to be well developed (especially for linear systems). This has made a good background for the development of observer-based FDI schemes.

Irrespective of the linear or non-linear FDI technique being employed, FDI performance will be usually impaired by the lack of robustness to model uncertainty. Indeed, the model-reality mismatch may cause very undesirable situations such as undetected faults or false alarms. This may lead to serious economical losses or even catastrophes.

As can be observed in the literature (Chen and Patton, 1999; Gertler, 1998; Korbicz *et al.*, 2004; Patton *et al.*, 2000), the most common approach to robust fault diagnosis is to use robust observers. This is mainly because of the fact that the theory of robust observers is relatively well developed in the control engineering literature. In particular, the so-called unknown input model uncertainty is mostly preferred. The observer resulting from such an approach is called the Unknown Input Observer (UIO). Although the origins of UIOs can be traced back to the early 1970s (cf. the seminal work of Wang *et al.* (1975)), the problem of designing such observers is still of paramount importance both from the theoretical and practical viewpoints. The

main objective of the subsequent part of this section is to present two unknown input observer design strategies that can be employed for the Lipschitz (Witczak and Korbicz, 2006; Witczak *et al.*, 2006b) and a general (Witczak *et al.*, 2002; 2006c) class of non-linear systems, respectively.

8.2.1. Observers for non-linear Lipschitz systems

Let us consider Lipschitz systems that can be described as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{h}(\boldsymbol{y}_k, \boldsymbol{u}_k) + \boldsymbol{g}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right), \tag{8.1}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{C}\boldsymbol{x}_{k+1}, \tag{8.2}$$

where $\boldsymbol{x}_k \in \mathbb{R}^n$ stands for the state vector, $\boldsymbol{y}_k \in \mathbb{R}^m$ is the output, $\boldsymbol{u}_k \in \mathbb{R}^r$ is the input, and $\boldsymbol{g}\left(\cdot\right)$ and $\boldsymbol{h}(\cdot)$ are non-linear functions. Additionally, $\boldsymbol{g}\left(\cdot\right)$ satisfies

$$\|\boldsymbol{g}\left(\boldsymbol{x}_1, \boldsymbol{u}\right) - \boldsymbol{g}\left(\boldsymbol{x}_2, \boldsymbol{u}\right)\|_2 \leq \gamma \|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2, \ \forall \boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{u}, \tag{8.3}$$

and $\gamma > 0$ stands for the Lipschitz constant.

Let us consider an observer for the system (8.1)–(8.2) described by the following equation:

$$\hat{\boldsymbol{x}}_{k+1} = \boldsymbol{A}\hat{\boldsymbol{x}}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{h}(\boldsymbol{y}_k, \boldsymbol{u}_k) + \boldsymbol{g}\left(\hat{\boldsymbol{x}}_k, \boldsymbol{u}_k\right) + \boldsymbol{K}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k), \tag{8.4}$$

where $\hat{\boldsymbol{x}}_k$ denotes the state estimate and $\boldsymbol{K}$ stands for the gain matrix.

The subsequent part of this section shows three theorems that present three different convergence conditions of (8.4). Following Thau (1973), let us assume that the pair $(\boldsymbol{A}, \boldsymbol{C})$ is observable. Let $\boldsymbol{P} = \boldsymbol{P}^T$, $\boldsymbol{P} > \boldsymbol{0}$ be a solution of the following Lyapunov equation:

$$\boldsymbol{Q} = \boldsymbol{P} - \boldsymbol{A}_0^T \boldsymbol{P} \boldsymbol{A}_0, \quad \boldsymbol{A}_0 = \boldsymbol{A} - \boldsymbol{K}\boldsymbol{C}, \tag{8.5}$$

where $\boldsymbol{A}_0$ is a stable matrix, i.e., $\rho(\boldsymbol{A}_0) < 1$, and $\boldsymbol{Q} = \boldsymbol{Q}^T$, $\boldsymbol{Q} > 0$. Moreover, let $\underline{\sigma}\left(\cdot\right)$ and $\bar{\sigma}\left(\cdot\right)$ stand for the minimum and maximum singular values of its argument, respectively.

Theorem 8.1. (Witczak and Korbicz, 2006) *Let us consider the observer (8.4) for the systems described by (8.1)–(8.2). If the Lipschitz constant γ (cf. (8.3)) satisfies*

$$\gamma < \sqrt{\frac{\underline{\sigma}\left(\boldsymbol{Q} - \boldsymbol{A}_0^T \boldsymbol{P} \boldsymbol{P} \boldsymbol{A}_0\right)}{\bar{\sigma}\left(\boldsymbol{P}\right) + 1}}, \quad \boldsymbol{Q} - \boldsymbol{A}_0^T \boldsymbol{P} \boldsymbol{P} \boldsymbol{A}_0 \succ \boldsymbol{0}, \tag{8.6}$$

then the observer (8.4) is asymptotically convergent.

Unfortunately, (8.6) may merely serve as a method of checking the convergence, but the gain matrix $\boldsymbol{K}$ has to be determined beforehand. This means that the design procedure boils down to selecting various gain matrices $\boldsymbol{K}$, solving the Lyapunov equation (8.5), and then checking the convergence condition (8.6). There is no doubt that this is an ineffective and inconvenient approach.

To tackle such a challenging problem, an effective design procedure was proposed in (Witczak and Korbicz, 2006), which can be written as follows:

Step 1: Obtain γ for (8.1)–(8.2).

Step 2: Solve a set of linear matrix inequalities: (8.7), (8.8), and (8.9).

Step 3: Obtain the gain matrix $\boldsymbol{K} = \boldsymbol{P}^{-1}\boldsymbol{L}$.

$$\begin{bmatrix} \beta\boldsymbol{I} & \boldsymbol{P} \\ \boldsymbol{P} & \beta\boldsymbol{I} \end{bmatrix} \succ \boldsymbol{0}, \quad \beta > 0, \quad \boldsymbol{P} \succ \boldsymbol{0}, \tag{8.7}$$

$$\begin{bmatrix} \boldsymbol{X} & \boldsymbol{A}^T\boldsymbol{P} - \boldsymbol{C}^T\boldsymbol{L}^T \\ \boldsymbol{P}\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} & \boldsymbol{I} \end{bmatrix} \succ \boldsymbol{0}, \tag{8.8}$$

$$\begin{bmatrix} \boldsymbol{P} - \gamma^2(\beta+1)\boldsymbol{I} - \boldsymbol{X} & \boldsymbol{A}^T\boldsymbol{P} - \boldsymbol{C}^T\boldsymbol{L}^T \\ \boldsymbol{P}\boldsymbol{A} - \boldsymbol{L}\boldsymbol{C} & \boldsymbol{P} \end{bmatrix} \succ \boldsymbol{0}. \tag{8.9}$$

The purpose of the subsequent part of this section is to present a straightforward approach for extending the techniques proposed in the preceding sections to discrete-time Lipschitz systems with unknown inputs, which can be described as follows:

$$\boldsymbol{x}_{k+1} = \boldsymbol{A}\boldsymbol{x}_k + \boldsymbol{B}\boldsymbol{u}_k + \boldsymbol{h}(\boldsymbol{y}_k, \boldsymbol{u}_k) + \boldsymbol{g}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) + \boldsymbol{E}\boldsymbol{d}_k, \tag{8.10}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{C}\boldsymbol{x}_{k+1}, \tag{8.11}$$

where $\boldsymbol{d}_k \in \mathbb{R}^q$ is the unknown input, and $\boldsymbol{E}$ is a known unknown input distribution matrix. In order to use the techniques described in the preceding sections for state estimation of the system (8.10)–(8.11), it is necessary to introduce some modifications concerning the unknown input.

Let us assume that

$$\operatorname{rank}(\boldsymbol{C}\boldsymbol{E}) = \operatorname{rank}(\boldsymbol{E}) = q, \tag{8.12}$$

(see (Chen and Patton, 1999, p. 72, Lemma 3.1) for a comprehensive explanation). If the condition (8.12) is satisfied, then it is possible to calculate $\boldsymbol{H} = (\boldsymbol{C}\boldsymbol{E})^+ = \left[(\boldsymbol{C}\boldsymbol{E})^T\boldsymbol{C}\boldsymbol{E}\right]^{-1}(\boldsymbol{C}\boldsymbol{E})^T$, where $(\cdot)^+$ stands for the pseudo-inverse of its argument. By multiplying (8.11) by $\boldsymbol{H}$ and then inserting (8.10), it can be shown that

$$\boldsymbol{x}_{k+1} = \bar{\boldsymbol{A}}\boldsymbol{x}_k + \bar{\boldsymbol{B}}\boldsymbol{u}_k + \bar{\boldsymbol{h}}\left(\boldsymbol{u}_k, \boldsymbol{y}_k\right) + \bar{\boldsymbol{g}}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) + \bar{\boldsymbol{E}}\boldsymbol{y}_{k+1}, \tag{8.13}$$

where

$$\bar{\boldsymbol{A}} = \bar{\boldsymbol{G}}\boldsymbol{A}, \qquad \bar{\boldsymbol{B}} = \bar{\boldsymbol{G}}\boldsymbol{B}, \qquad \bar{\boldsymbol{g}}\left(\cdot\right) = \bar{\boldsymbol{G}}\boldsymbol{g}\left(\cdot\right),$$
$$\bar{\boldsymbol{h}}\left(\cdot\right) = \bar{\boldsymbol{G}}\boldsymbol{h}(\cdot), \qquad \bar{\boldsymbol{G}} = \boldsymbol{I} - \boldsymbol{E}\boldsymbol{H}\boldsymbol{C}, \qquad \bar{\boldsymbol{E}} = \boldsymbol{E}\boldsymbol{H}.$$

Thus, the unknown input observer for (8.10)–(8.11) is given as follows:

$$\hat{\boldsymbol{x}}_{k+1} = \bar{\boldsymbol{A}}\hat{\boldsymbol{x}}_k + \bar{\boldsymbol{B}}\boldsymbol{u}_k + \bar{\boldsymbol{h}}\left(\boldsymbol{u}_k, \boldsymbol{y}_k\right) + \bar{\boldsymbol{g}}\left(\boldsymbol{x}_k, \boldsymbol{u}_k\right) + \bar{\boldsymbol{E}}\boldsymbol{y}_{k+1} + \boldsymbol{K}(\boldsymbol{y}_k - \boldsymbol{C}\hat{\boldsymbol{x}}_k). \tag{8.14}$$

A simple comparison of (8.1) and (8.13) leads to the conclusion that the observer (8.14) can be designed with the above-mentioned three-step procedure, taking into account the fact that (cf. (8.3)):

$$\|\bar{\boldsymbol{g}}\left(\boldsymbol{x}_1, \boldsymbol{u}\right) - \bar{\boldsymbol{g}}\left(\boldsymbol{x}_2, \boldsymbol{u}\right)\|_2 \leq \bar{\gamma}\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|_2, \quad \forall \boldsymbol{x}_1, \boldsymbol{x}_2, \boldsymbol{u}, \tag{8.15}$$

and assuming that the pair $(\bar{\boldsymbol{A}}, \boldsymbol{C})$ is observable.

8.2.2. Extended unknown input observers

Let us consider a non-linear discrete-time system described by

$$\boldsymbol{x}_{k+1} = \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{h}(\boldsymbol{u}_k) + \boldsymbol{E}_k\boldsymbol{d}_k, \tag{8.16}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{C}_{k+1}\boldsymbol{x}_{k+1}. \tag{8.17}$$

Using the similar approach as in Section 8.2.1, it can be shown (Witczak *et al.*, 2006c) that the structure of the so-called Extended Unknown Input Observer (EUIO) is

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}_{k+1/k} + \boldsymbol{K}_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{C}_{k+1}\hat{\boldsymbol{x}}_{k+1/k}), \tag{8.18}$$

where

$$\hat{\boldsymbol{x}}_{k+1/k} = \bar{\boldsymbol{g}}\left(\hat{\boldsymbol{x}}_k\right) + \bar{\boldsymbol{h}}\left(\boldsymbol{u}_k\right) + \bar{\boldsymbol{E}}_k\boldsymbol{y}_{k+1}. \tag{8.19}$$

As a consequence, the algorithm used for state estimation of (8.16)–(8.17) can be given as follows (Witczak *et al.*, 2006c):

$$\hat{\boldsymbol{x}}_{k+1/k} = \bar{\boldsymbol{g}}\left(\hat{\boldsymbol{x}}_k\right) + \bar{\boldsymbol{h}}\left(\boldsymbol{u}_k\right) + \bar{\boldsymbol{E}}_k\boldsymbol{y}_{k+1}, \tag{8.20}$$

$$\boldsymbol{P}_{k+1/k} = \bar{\boldsymbol{A}}_k\boldsymbol{P}_k{\bar{\boldsymbol{A}}_k}^T + \boldsymbol{Q}_k, \tag{8.21}$$

$$\boldsymbol{K}_{k+1} = \boldsymbol{P}_{k+1/k}\boldsymbol{C}_{k+1}^T\left(\boldsymbol{C}_{k+1}\boldsymbol{P}_{k+1/k}\boldsymbol{C}_{k+1}^T + \boldsymbol{R}_{k+1}\right)^{-1}, \tag{8.22}$$

$$\hat{\boldsymbol{x}}_{k+1} = \hat{\boldsymbol{x}}_{k+1/k} + \boldsymbol{K}_{k+1}(\boldsymbol{y}_{k+1} - \boldsymbol{C}_{k+1}\hat{\boldsymbol{x}}_{k+1/k}), \tag{8.23}$$

$$\boldsymbol{P}_{k+1} = \left[\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}_{k+1}\right]\boldsymbol{P}_{k+1/k}, \tag{8.24}$$

where

$$\bar{\boldsymbol{A}}_k = \left.\frac{\partial\bar{\boldsymbol{g}}\left(\boldsymbol{x}_k\right)}{\partial\boldsymbol{x}_k}\right|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_k} = \bar{\boldsymbol{G}}_k\left.\frac{\partial\boldsymbol{g}\left(\boldsymbol{x}_k\right)}{\partial\boldsymbol{x}_k}\right|_{\boldsymbol{x}_k=\hat{\boldsymbol{x}}_k} = \bar{\boldsymbol{G}}_k\boldsymbol{A}_k. \tag{8.25}$$

The main aim is to show that the convergence of the EUIO strongly depends on the instrumental matrices $\boldsymbol{Q}_k$ and $\boldsymbol{R}_k$. Moreover, the fault-free mode is assumed, i.e., $\boldsymbol{f}_k = \boldsymbol{0}$.

Using (8.23), the state estimation error can be given as

$$\boldsymbol{e}_{k+1} = \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1} = \left[\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}_{k+1}\right]\boldsymbol{e}_{k+1/k}, \tag{8.26}$$

and

$$\boldsymbol{e}_{k+1/k} = \boldsymbol{x}_{k+1} - \hat{\boldsymbol{x}}_{k+1/k} = \bar{\boldsymbol{g}}\left(\boldsymbol{x}_k\right) - \bar{\boldsymbol{g}}\left(\hat{\boldsymbol{x}}_k\right) = \boldsymbol{\alpha}_k\bar{\boldsymbol{A}}_k\boldsymbol{e}_k, \tag{8.27}$$

where $\boldsymbol{\alpha}_k = \mathrm{diag}(\alpha_{1,k}, \ldots, \alpha_{n,k})$ is an unknown diagonal matrix. Thus, using (8.27), the equation (8.26) becomes

$$\boldsymbol{e}_{k+1} = \left[\boldsymbol{I} - \boldsymbol{K}_{k+1}\boldsymbol{C}_{k+1}\right]\boldsymbol{\alpha}_k\bar{\boldsymbol{A}}_k\boldsymbol{e}_k. \tag{8.28}$$

It is clear from (8.27) that $\boldsymbol{\alpha}_k$ represents the linearisation error. First, let us define

$$\bar{\alpha}_k = \max_{j=1,\ldots,n}|\alpha_{j,k}|, \quad \underline{\alpha}_k = \min_{j=1,\ldots,n}|\alpha_{j,k}|. \tag{8.29}$$

Theorem 8.2. (Witczak *et al.*, 2006c) *If*

$$\bar{\alpha}_k \leq \left(\underline{\alpha}_k^2 \frac{\underline{\sigma}\left(\bar{\boldsymbol{A}}_k\right)^2 \underline{\sigma}\left(\boldsymbol{C}_{k+1}\right)^2 \underline{\sigma}\left(\bar{\boldsymbol{A}}_k \boldsymbol{P}_k \bar{\boldsymbol{A}}_k{}^T + \boldsymbol{Q}_k\right)}{\bar{\sigma}\left(\boldsymbol{C}_{k+1}\boldsymbol{P}_{k+1/k}\boldsymbol{C}_{k+1}^T + \boldsymbol{R}_{k+1}\right)} + \right.$$

$$\left. + \frac{(1-\zeta)\underline{\sigma}\left(\bar{\boldsymbol{A}}_k \boldsymbol{P}_k \bar{\boldsymbol{A}}_k{}^T + \boldsymbol{Q}_k\right)}{\bar{\sigma}\left(\bar{\boldsymbol{A}}_k\right)^2 \bar{\sigma}\left(\boldsymbol{P}_k\right)} \right)^{\frac{1}{2}}, \tag{8.30}$$

where $0 < \zeta < 1$*, then the proposed extended unknown input observer is locally asymptotically convergent.*

It is clear from (8.30) that the bound of $\bar{\alpha}_k$ *can be maximised by suitable settings of the instrumental matrices* $\boldsymbol{Q}_k$ *and* $\boldsymbol{R}_k$*. This can be realised as follows (Witczak* et al.*, 2006c):*

$$\boldsymbol{Q}_k = (\gamma \boldsymbol{\varepsilon}_k^T \boldsymbol{\varepsilon}_k + \delta_1)\boldsymbol{I}, \quad \boldsymbol{\varepsilon}_k = \boldsymbol{y}_k - \boldsymbol{C}_k \hat{\boldsymbol{x}}_k, \tag{8.31}$$

$$\boldsymbol{R}_{k+1} = \delta_2 \boldsymbol{I}, \tag{8.32}$$

with $\gamma > 0$ *and* $\delta_1 > 0$*,* $\delta_2 > 0$ *large and small enough, respectively.*

8.3. Neural networks in FDI schemes

Artificial Neural Networks (ANNs) have been intensively studied during the last two decades and successfully applied to dynamic system modelling and fault diagnosis (Frank and Köppen-Seliger, 1997; Korbicz, 2006; Korbicz *et al.*, 2004; Köppen-Seliger and Frank, 1999; Narendra and Parthasarathy, 1990; Witczak, 2006a). Neural networks stand for an interesting and valuable alternative to the classical methods, because they can deal with very complex situations which are not sufficiently defined for deterministic algorithms. They are especially useful when there is no mathematical model of a process being considered. In such situations, the classical approaches such as observers or parameter estimation methods cannot be applied. Neural networks provide excellent mathematical tools for dealing with non-linear problems (Haykin, 1999; Korbicz *et al.*, 1994; Nelles, 2001; Norgard *et al.*, 2000; Osowski, 2006; Rutkowski, 2005). They have an important property owing to which any non-linear function can be approximated with an arbitrary accuracy using a neural network with a suitable architecture and weight parameters. For continuous mappings, one hidden layer-based ANN is sufficient but in other cases two hidden layers should be implemented. ANNs are parallel data processing tools capable of learning functional dependencies of the data. This feature is extremely useful for solving different pattern recognition problems.

Another attractive property is the self-learning ability. A neural network can extract the system features from historical training data using the learning algorithm, requiring a little or no *a priori* knowledge about the process. This makes ANNs non-linear modelling tools of a great flexibility. Neural networks are also robust with respect to incorrect or missing data. Protective relaying based on ANNs is not affected by changes in the system operating conditions. Neural networks have also high

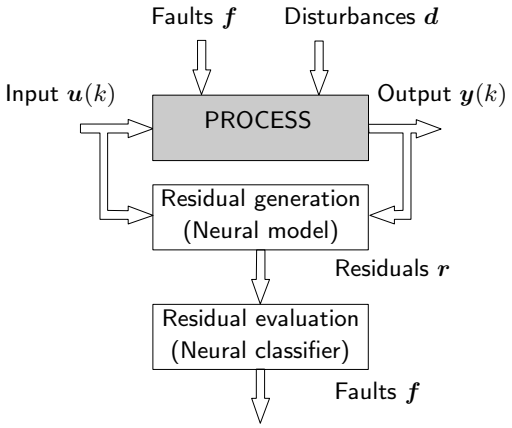


Fig. 8.4. Model-based fault diagnosis using neural networks

computation rates, large input error tolerance and adaptive capability. These features allow applying neural networks effectively to the modelling and identification of complex non-linear dynamic processes and fault diagnosis. Modern methods of FDI for dynamic systems can be split into four broad categories: model-based, robust model-based, knowledge-based and data analysis-based approaches. In the consecutive sections, a possible application of various neural networks in the framework of fault diagnosis is carefully discussed.

8.3.1. Model-based approaches

Model-based approaches generally utilise results from the field of control theory, which rely on parameter or state estimation (Chen and Patton, 1999; Isermann, 2006). The approach is based on the fact that a fault will cause changes in certain physical parameters which in turn will cause changes in some model parameters or states. When using this approach, it is essential to have quite accurate models of a process being considered. Figure 8.4 presents a block scheme of model-based fault diagnosis. As has been mentioned, fault diagnosis procedure consists of two separate stages: residual generation and residual evaluation. The residual generation process is based on a comparison between the output of the system and the output of the model constructed. As a result, the difference, or so-called residual, is expected to be near zero under the normal operating conditions, but on occurrence of a fault, a deviation from zero should appear. Unfortunately, designing mathematical models for complex non-linear systems can be difficult or even impossible.

For the model-based approach, the neural network replaces the analytical model that describes the process under the normal operating conditions (Frank and Köppen-Seliger, 1997; Patan, 2007; Patan *et al.*, 2005). First, the network has to be trained to settle this task. Learning data can be collected directly from the process, if possible, or from a simulation model that should be as realistic as possible. The latter possibility is of special interest for data acquisition in different faulty situations. This is especially

important for the task of testing the residual generator because such data are not generally available from the real process. The training process can be carried out off-line or on-line (it depends on the availability of data).

The possibility to train a network on-line is very attractive, especially in the case of adapting a neural model to mutable environment or time-varying systems. After finishing the training, a neural network is ready for on-line residual generation. In order to be able to capture the dynamic behaviour of the system, the neural network should have dynamic properties, e.g., it should be a recurrent network. Residual evaluation is a decision-making process that transforms quantitative knowledge into qualitative `Yes` or `No` statements. It can also be perceived as a classification problem. The task is to match each pattern of the symptom vector with one of the pre-assigned classes of faults and the fault-free case. This process may be highly facilitated with intelligent decision making. To perform residual evaluation, neural networks can be applied, e.g., feed-forward networks or self-organizing maps.

Multilayer perceptron. Artificial neural networks are constructed with a certain number of single processing units, which are called neurons. A standard neuron model is described by the following equation (Duch *et al.*, 2000; Korbicz *et al.*, 1994; Osowski, 2006):

$$y = F\left(\sum_{n=1}^{N} w_n u_n + u_0\right), \tag{8.33}$$

where $u_n,\ n = 1, 2, \ldots, N$, denotes neuron inputs, u_0 is the threshold, w_n denotes synaptic weight coefficients and $F(\cdot)$ is the non-linear activation function. Sigmoid and hyperbolic tangent functions are very popular and most frequently used. The multi-layer perceptron is a network in which the neurons are grouped into layers where an input layer, one or more hidden layers and an output layer can be distinguished. The main task of the input units (black squares) is preliminary input data processing $\boldsymbol{u} = [u_1, u_2, \ldots, u_N]^T$ and passing them onto the elements of the hidden layer. Data processing can comprise, e.g., scaling, filtering or signal normalisation. Fundamental neural data processing is carried out in hidden and output layers. It is necessary to notice that links between the neurons are designed in such a way that each element of the subsequent layer is connected with each element of the previous layer. These connections are assigned with suitable weight coefficients, which are determined, for each separate case, depending on the task the network should solve. The output layer generates the network response vector $\boldsymbol{y}$.

Recurrent networks. Feed-forward networks can only represent static mappings, and therefore they need past inputs and outputs of the modelled process. This can be performed introducing suitable delays. As a result, a neural network with time delay lines can be elaborated (Narendra and Parthasarathy, 1990; Norgard *et al.*, 2000). Unfortunately, this kind of networks has some disadvantages. First of all, it is required to know the exact order of the process. If the order of the process is known, all necessary inputs and outputs should be fed to the network. In this way, the input space of the network becomes large.

In many practical cases, there is no possibility to learn the order of the system, and the number of suitable delays has to be selected experimentally. Another disadvantage

is the limited past history horizon, thereby preventing the modelling of arbitrarily long time dependencies between the inputs and the desired outputs. Moreover, the trained network has strictly static, not dynamic, characteristics. More natural, dynamic behaviour is assured by recurrent networks (Gupta *et al.*, 2003; Haykin, 1999).

Recurrent neural networks are characterised by considerably better properties, looking from the point of view of their application to control theory. As a result of feedbacks introduced to network structures, it is possible to accumulate the information and use it later. From a possible feedback location point of view, recurrent networks can be divided as follows (Tsoi and Back, 1994):

- *local recurrent networks* – there are feedbacks only inside neuron models. These networks have a structure similar to static feed-forward ones, but consist of dynamic neuron models.
- *global recurrent networks* – there are feedbacks allowed between neurons of different layers or between neurons of the same layer.

Globally recurrent networks, like the fully recurrent network of the Williams-Zipser (1989) type and Elman/Jordan partially recurrent networks (Elman, 1990; Jordan, 1986), are the most widely known recurrent structures elaborated to date and also well documented. Generally, globally recurrent neural networks, in spite of their usefulness in control theory, have some disadvantages. These architectures suffer from a lack of stability; for a given set of initial values, the activations of linear output neurons may grow unlimited. Training algorithms are also complicated and time consuming causing slow convergence of the training. Moreover, the number of states (model order) cannot be selected independently from the number of hidden neurons, and then serious problems can occur while selecting the structure of the network in order to achieve both proper approximation abilities and a proper order of the model (Nelles, 2001).

Model with the IIR filter. The disadvantages of globally recurrent networks can be partially avoided by using locally recurrent networks. The fundamental unit of such networks is a dynamic neuron model. It can be obtained by incorporating a linear dynamic system into the classical neuron model. The dynamics are introduced into neuron in such a way that the neuron activation depends on its internal states. This is done by introducing the Infinite Impulse Response (IIR) filter into the neuron structure (Patan and Parisini, 2005). In such models, one can distinguish three main parts: the weight sumator, the filter block and the activation block. The filter is placed between the weighted sumator and the activation function. The behaviour of the dynamic neuron model being considered is described by the following set of equations:

$$\begin{aligned} x(k) &= \sum_{n=1}^{N} w_n u_n(k), \\ \widetilde{y}(k) &= -\sum_{i=1}^{I} a_i \widetilde{y}(k-i) + \sum_{i=0}^{I} b_i x(k-i), \\ y(k) &= F(g\,\widetilde{y}(k) + c), \end{aligned} \tag{8.34}$$

where $w_n,\ n = 1, \ldots, N$ denotes the input weights, $u_n(k),\ n = 1, \ldots, N$ are the neuron inputs, N is the number of the inputs, $\widetilde{y}(k)$ denotes the filter output, $a_i,\ i = 1, \ldots, I$ and $b_i,\ i = 0, \ldots, I$ are feedback and feed-forward filter parameters, respectively, $F(\cdot)$ is a non-linear activation function that produces the neuron output $y(k)$, and g and c are the slope parameter and the bias of the activation function, respectively.

Due to the dynamic characteristics of neurons, a neural network of the feed-forward structure can be designed. Taking into account the fact that this network does not have any recurrent links between the neurons, to adapt the network parameters a training algorithm based on the back-propagation idea can be elaborated. The calculated output is propagated back to the inputs through hidden layers containing dynamic filters. As a result, extended dynamic back propagation is defined (Korbicz *et al.*, 2001). This algorithm can have both the on-line and the off-line forms, and therefore it can be widely used in control theory. The choice of the proper mode is dependent on problem specification.

8.3.2. Robust model-based approach

As was mentioned in Section 8.2, when non-linear state space models are available, fault diagnosis can be realised by using the concept of an unknown input observer. Unfortunately, when the direction of faults is similar to that of an unknown input, then the unknown input decoupling procedure may considerably impair fault sensitivity. If the above-mentioned approach fails, then describing model uncertainty in a different way seems to be a good remedy. One of the possible approaches is to use statistical techniques (Atkinson and Donev, 1992; Walter and Pronzato, 1996) (for an example regarding different approaches, the reader is referred to (Delebecque *et al.*, 2003)) to obtain parameter uncertainty of the model and, consequently, model output uncertainty. Such parameter uncertainty is defined as the parameter confidence region (Atkinson and Donev, 1992; Walter and Pronzato, 1996) containing a set of admissible parameters that are consistent with the measured data. Thus it is evident that parameter uncertainty depends on measurement uncertainty, i.e., noise, disturbances, etc.

The knowledge about parameter uncertainty makes it possible to design the so-called adaptive threshold (Frank *et al.*, 1999). The adaptive threshold, contrary to the fixed one, bounds the residual at a level that is dependent on model uncertainty, and hence it provides a more reliable fault detection.

Contrary to the typical industrial applications of neural networks that are presented in the literature (Chen and Patton, 1999; Karpenko *et al.*, 2003; Korbicz *et al.*, 2004; Mrugalski and Korbicz, 2006), Witczak *et al.* (2006a) defined the task of designing a neural network in such a way as to obtain a model with a possibly small uncertainty. Indeed, the approaches presented in the literature try to obtain a model that is best suited to a particular data set. This may result in a model with a relatively large uncertainty. A degraded performance of fault diagnosis constitutes a direct consequence of using such models.

To tackle this challenging problem for non-linear dynamic systems, the GMDH (Group Method of Data Handling) approach (Ivakhnenko and Mueller, 1995; Korbicz and Mrugalski, 2007) can be effectively adapted (Witczak *et al.*, 2006a). The authors proposed a complete design procedure concerning the application of GMDH neural

networks to robust fault detection. Starting from a set of input-output measurements of the system, it is shown how to estimate the parameters and the corresponding uncertainty of a neuron using the so-called bounded-error approach (Milanese *et al.*, 1996; Walter and Pronzato, 1996). As a result, they obtained a tool that is able to generate an adaptive threshold. The methodology developed for parameter and uncertainty estimation of a neuron makes it possible to formulate an algorithm that allows obtaining a neural network with a relatively small modelling uncertainty. All the hard computations regarding the design of the GMDH neural network are performed off-line, and hence the problem regarding the time-consuming calculations is not of paramount importance. The above-discussed technique will be clearly detailed in Section 8.3.2.

As has been mentioned, the reliability of such fault diagnosis schemes is strongly dependent on model uncertainty, i.e., the mismatch between a neural network and the system being considered. Thus, it is natural to minimise model uncertainty as far as possible. This can be realised with the application of Optimum Experimental Design (OED) theory (Atkinson and Donev, 1992; Uciński, 2005; Walter and Pronzato, 1996). Recently, Witczak and Prętki (2005) developed a D-optimum experimental design strategy that can be used for training single-output neural networks. They also showed how to use the obtained network for robust fault detection with an adaptive threshold. In (Witczak, 2006b), the author showed how to extend this technique to multi-input multi-output neural networks. He also proposed a sequential experimental design algorithm that allows obtaining a one-step-ahead D-optimum input. This algorithm can be perceived as a hybrid one since it can be used for both training and data development.

Robust GMDH neural networks. A successful application of the ANNs to the system identification and fault diagnosis tasks (Witczak, 2006a) depends on a proper selection of the neural network architecture. In the case of the classical ANNs such as Multi-Layer Perceptron (MLP), the problem reduces to the selection of the number of layers and the number of neurons in a particular layer. If the obtained network does not satisfy prespecified requirements, then a new network structure is selected and parameter estimation is repeated once again. The determination of the appropriate structure and parameters of the model in the presented way is a complex task. Furthermore, an arbitrary selection of the ANN structure can be a source of model uncertainty. Thus, it seems desirable to have a tool which can be employed for automatic selection of the ANN structure, based only on the measured data. To overcome this problem, GMDH neural networks (Ivakhnenko and Mueller, 1995; Mrugalski, 2004; Mrugalski and Korbicz, 2005) have been proposed. The synthesis process of the GMDH model is based on iterative processing of a sequence of operations. This process leads to the evolution of the resulting model structure in such a way as to obtain the best quality approximation of the identified system. Thus, the task of designing a neural network is defined in such a way so as to obtain a model with a small uncertainty.

The idea of the GMDH approach relies on replacing the complex neural model by the set of hierarchically connected neurons. The behaviour of each neuron should reflect the behaviour of the system being considered. It follows from the rule of the

GMDH algorithm that the parameters of each neuron are estimated in such a way that their output signals are the best approximation of the real system output. In this situation, the neuron should have the ability to represent the dynamics. One way out of this problem is to use dynamic neurons (Patan and Parisini, 2005). Dynamics in these neurons are realised by introducing a linear dynamic system – an IIR filter. The process of GMDH network synthesis leads to the evolution of the resulting model structure in such a way as to obtain the best quality approximation of the real system. An outline of the GMDH algorithm can be as follows (Mrugalski, 2004; Witczak *et al.*, 2006a):

Step 1: Determine all neurons (estimate their parameter vectors $\boldsymbol{p}_n^{(l)}$ with the training data set $\mathcal{T}$) whose inputs consist of all possible couples of input variables, i.e., $(r-1)r/2$ couples, where r is the dimension of the system input vector.

Step 2: Using a validation data set $\mathcal{V}$, not employed during the parameter estimation phase, select several neurons which are best fitted in terms of the chosen criterion.

Step 3: If the termination condition is fulfilled (either the network fits the data with a desired accuracy or the introduction of new neurons did not induce a significant increase in the approximation abilities of the neural network), then STOP, otherwise use the outputs of the best-fitted neurons (selected in *Step 2*) to form the input vector for the next layer, and then go to *Step 1*.

To obtain the final structure of the network, all unnecessary neurons are removed, leaving only those which are relevant to the computation of the model output. The procedure of removing the unnecessary neurons is the last stage of the synthesis of the GMDH neural network. The appealing feature of the above algorithm is that the techniques for parameter estimation of linear-in-parameter models can be used during the realisation of *Step 1*. This is possible under the standard invertibility assumption of the activation function of a network.

Confidence estimation of GMDH neural networks. Even though the application of the GMDH approach to model structure selection can improve the quality of the model, the resulting structure is not the same as that of the system. It can be shown (Mrugalski, 2004) that the application of the classical evaluation criteria such as the Akaike Information Criterion (AIC) and the Final Prediction Error (FPE) (Ivakhnenko and Mueller, 1995; Mueller and Lemke, 2000) can lead to the selection of inappropriate neurons and, consequently, to unnecessary structural errors.

Apart from the model structure selection stage, inaccuracy in parameter estimates also contributes to modelling uncertainty. Indeed, while applying the least-square method to parameter estimation of neurons, a set of restrictive assumptions has to be satisfied (see, e.g., (Witczak *et al.*, 2006a) for further explanations). An effective remedy to such a challenging problem is to use the so-called Bounded Error Approach (BEA) (Milanese *et al.*, 1996; Witczak *et al.*, 2006a). Let us consider the following system:

$$y(k) = \boldsymbol{r}(k)^T \boldsymbol{p} + \varepsilon(k). \tag{8.35}$$

where $\boldsymbol{r}(k)$ stands the regressor vector, $\boldsymbol{p} \in \mathbb{R}^{n_p}$ denotes the parameter vector, and $\varepsilon(k)$ represents the difference between the original system and the model.

The problem is to obtain the parameter estimate vector $\hat{\boldsymbol{p}}$, as well as the associated parameter uncertainty required to design robust fault detection system. The knowledge regarding the set of admissible parameter values allows obtaining the confidence region of the model output which satisfies

$$\tilde{y}^m(k) \leq y(k) \leq \tilde{y}^M(k), \tag{8.36}$$

where $\tilde{y}^m(k)$ and $\tilde{y}^M(k)$ are the minimum and maximum admissible values of the model output that are consistent with the input-output measurements of the system.

It is assumed that $\varepsilon(k)$ consists of a structural deterministic error caused by the model-reality mismatch, and the stochastic error caused by the measurement noise is bounded as follows:

$$\varepsilon^m(k) \leq \varepsilon(k) \leq \varepsilon^M(k), \tag{8.37}$$

where the bounds $\varepsilon^m(k)$ and $\varepsilon^M(k)$ ($\varepsilon^m(k) \neq \varepsilon^M(k)$) can be estimated (Witczak *et al.*, 2006a).

The idea underlying the bounded-error approach is to obtain a feasible parameter set $\mathbb{P}$ (Milanese *et al.*, 1996) that is consistent with the input-output measurements used for parameter estimation. The resulting $\mathbb{P}$ is described by a polytope defined by a set of vertices $\mathbb{V}$. Thus, the problem of determining the model output uncertainty can be solved as follows:

$$\boldsymbol{r}^T(k)\boldsymbol{p}^m(k) \leq \boldsymbol{r}^T(k)\boldsymbol{p} \leq \boldsymbol{r}^T(k)\boldsymbol{p}^M(k), \tag{8.38}$$

where

$$\boldsymbol{p}^m(k) = \arg\min_{\boldsymbol{p}\in\mathbb{V}} \boldsymbol{r}^T(k)\boldsymbol{p}, \quad \boldsymbol{p}^M(k) = \arg\max_{\boldsymbol{p}\in\mathbb{V}} \boldsymbol{r}^T(k)\boldsymbol{p}. \tag{8.39}$$

As has been mentioned, the neurons in the l-th ($l > 1$) layer are fed with the outputs of the neurons from the $(l-1)$-th layer. In order to modify the above-presented approach for the uncertain regressor case, let us denote an unknown "true" value of the regressor $\boldsymbol{r}_n(k)$ by a difference between the measured value of the regressor $\boldsymbol{r}(k)$ and the error in the regressor $\boldsymbol{e}(k)$:

$$\boldsymbol{r}_n(k) = \boldsymbol{r}(k) - \boldsymbol{e}(k), \tag{8.40}$$

where it is assumed that the error $\boldsymbol{e}(k)$ is bounded as

$$e_i^m(k) \leq e_i(k) \leq e_i^M(k), \quad i = 1, \ldots, n_p. \tag{8.41}$$

Using (8.35) and substituting (8.40) into (8.41), one can define the space containing the parameter estimates:

$$\varepsilon^m(k) - \boldsymbol{e}^T(k)\boldsymbol{p} \leq y(k) - \boldsymbol{r}(k)^T\boldsymbol{p} \leq \varepsilon^M(k) - \boldsymbol{e}^T(k)\boldsymbol{p}, \tag{8.42}$$

which makes it possible to adapt the above-described technique to the error-in-regressor case (Witczak *et al.*, 2006a).

The proposed modification of the BEA makes it possible to estimate the parameter vectors of the neurons from the l-th, $l > 1$ layers. Finally, it can be shown that the model output uncertainty has the following form:

$$\tilde{y}^m(k) \leq \boldsymbol{r}_n^T\boldsymbol{p} \leq \tilde{y}^M(k). \tag{8.43}$$

In order to adapt the presented approach to parameter estimation of non-linear neurons with an activation function $\xi(\cdot)$, it is necessary to transform the relation

$$\varepsilon^m(k) \leq y(k) - \xi\left(\left(\boldsymbol{r}(k)\right)^T \boldsymbol{p}\right) \leq \varepsilon^M(k), \tag{8.44}$$

using $\xi^{-1}(\cdot)$, and hence

$$\xi^{-1}\left(y(k) - \varepsilon^M(k)\right) \leq \left(\boldsymbol{r}(k)\right)^T \boldsymbol{p} \leq \xi^{-1}\left(y(k) - \varepsilon^m(k)\right). \tag{8.45}$$

Knowing the model structure and possessing the knowledge regarding its uncertainty, it is possible to design a robust fault detection scheme with an adaptive threshold. The model output uncertainty interval, calculated with the application of the GMDH model, should contain the real system response in the fault-free mode. Therefore, the system output should satisfy

$$\tilde{y}^m(k) + \varepsilon^m(k) \leq y(k) \leq \tilde{y}^M(k) + \varepsilon^M(k). \tag{8.46}$$

This means that robust fault detection boils down to checking if the output of the system satisfies (8.46). Thus, when (8.46) is violated, then a fault symptom occurs.

8.3.3. Knowledge-based approaches

Knowledge-based approaches are generally based on expert or qualitative reasoning (Zhang and Ellis, 1991). Several knowledge-based fault diagnosis approaches have been proposed. These include the rule-based approach, where the diagnostic rule can be formulated from the process structure and unit functions as well as the qualitative simulation-based approach. In the rule-based approach, faults are usually diagnosed by casually tracing symptoms backwards along their propagation paths. Fuzzy reasoning can be used in the rule based approach to handle uncertain information.

In the qualitative simulation-based approach, qualitative models of a process are used to predict the behaviour of the process under the normal operating conditions and various faulty conditions. Fault detection and diagnosis are then performed by comparing the predicted behaviour with the actual observations. The methods that fall into this category can be viewed as fault analysers because their objective is to make a decision about whether or not a fault has occurred in the system based on the set of logical rules that are either pre-programmed by an expert or learned through a training process (Fig. 8.5). When data about process states or operating condition is passed into the fault analyser, it is checked against the rule base stored there and a decision about the operating conditions of the system is carried out.

Neural networks are an excellent tool to design such fault analysers (Köppen-Seliger and Frank, 1999). The well-known feed-forward multi-layered networks are most frequently used. To summarise, to develop knowledge-based diagnostic systems, the knowledge about the process structure, process unit fuctions, and qualitative models of process units under various faulty conditions is required. Therefore, the development of a knowledge-based fault diagnosis system is, generally, computationally demanding.

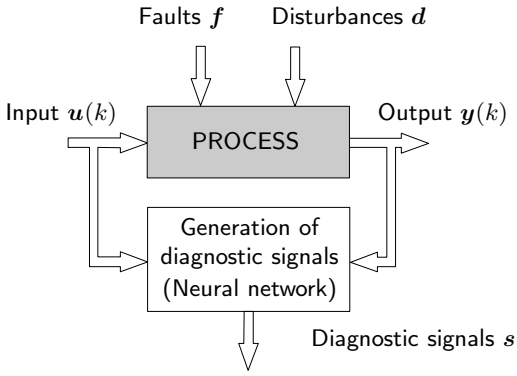


Fig. 8.5. Model-free fault diagnosis using neural networks

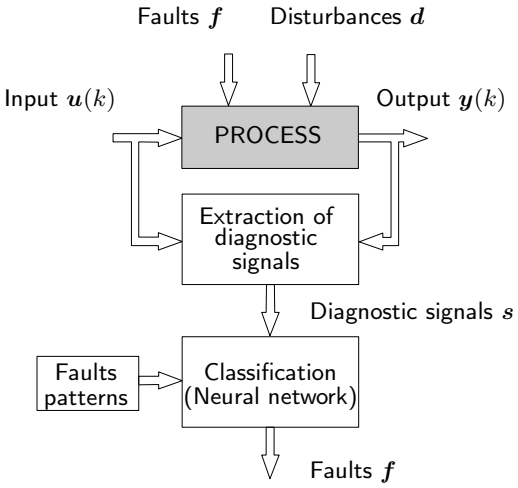


Fig. 8.6. Fault diagnosis as a pattern recognition task

8.3.4. Data analysis-based approaches

In the data analysis-based approaches, process operational data covering various normal and abnormal operations are used to extract diagnostic knowledge. Two main methods can be distinguished: neural network-based and multivariate statistical data analysis-based fault diagnosis. In the neural network-based fault diagnosis, the only knowledge required is the training data which contain faults and their symptoms (Karpenko *et al.*, 2003). The fault symptoms are in the form of variations in process measurements. Through training, the relationships between faults and their symptoms can be discovered and stored as network weights. The trained network can be then used to diagnose faults in such a way that it can associate the observed abnormal conditions with their corresponding faults. In fact, this group of approaches uses neural networks as pattern classifiers (Fig. 8.6).

In multivariate statistical data analysis techniques, fault signatures are extracted from process operational data through some multivariate statistical methods like Principal Component Analysis (PCA), projection to the latent structure or non-linear PCA. It should be mentioned that statistical data analysis such as PCA can be carried out by means of neural network training, e.g., Generalized Hebbian Algorithm (GHA) or Adaptive Principal components EXtraction (APEX) algorithms which utilise a single perceptron network or its modifications (Haykin, 1999).

8.4. Applications

The main objective of this section is to presents two applications of the approaches described in the preceding sections. In particular, the first example is devoted to neural network-based modelling of a DC motor. The second example concerns robust fault diagnosis of an induction motor with the extended unknown input observer described in Section 8.2.2.

8.4.1. Neural network-based modelling of a DC motor

The mathematical model of a separately excited DC motor is composed of two differential equations. The electrical part of the DC motor equations is described by

$$u_a(t) = R_a i_a(t) + L_a \frac{\mathrm{d}i_a(t)}{\mathrm{d}t} + e_a(t), \tag{8.47}$$

where $u_a(t)$ is the motor armature voltage, $i_a(t)$ denotes the armature current, R_a stands for the armature coil resistance, L_a is the armature coil inductance, and $e_a(t)$ is the counter-electromotive force. The mechanical part of the DC motor equations has the following form:

$$J_m \frac{\mathrm{d}\omega(t)}{\mathrm{d}t} = T_m(t) - B_m\omega(t) - T_L(t) - T_f(\omega(t)), \tag{8.48}$$

where J_m is the motor moment of inertia, $\omega(t)$ denotes the angular velocity of the motor, $T_m(t)$ stands for the motor torque, B_m is the viscous friction coefficient, $T_L(t)$ is the load torque, $T_f(\omega(t))$ is the breakaway friction torque. The counter-electromotive force $e_a(t)$ is proportional to the angular velocity of the motor $\omega(t)$:

$$e_a(t) = K_e\omega(t), \tag{8.49}$$

where K_e is the motor voltage constant. The motor torque $T_m(t)$ is proportional to the armature current $i_a(t)$:

$$T_m(t) = K_m i_a(t), \tag{8.50}$$

where K_m is the motor torque constant.

Applying block diagram transformation rules, the model can be transformed into a more convenient form shown in Fig. 8.7, where

$$G_1(s) = \frac{1}{(L_a s + R_a)(J_m s + B_m) + K_e}, \tag{8.51}$$

$$G_2(s) = \frac{L_a s + R_a}{K_m}. \tag{8.52}$$

The friction torque can be considered as a function of angular velocity and it is

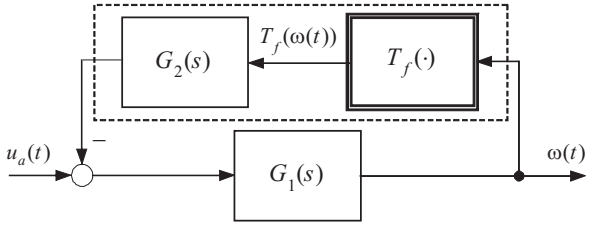


Fig. 8.7. Transformed block diagram of the DC motor

assumed to be the sum of Stribeck, Coulomb, and viscous components (Armstrong and de Wit, 1995). While the viscous friction torque opposes motion and is proportional to the angular velocity, the Coulomb friction torque is constant at any angular velocity. The Stribeck friction is a non-linear component occurring at low angular velocities. The sum of non-linear components of the friction, i.e., the Coulomb and Stribeck frictions, is called the breakaway friction. The rotational friction is a complex physical phenomenon and no exact theoretical model exists, thus numerous models of the friction torque are used in practice, e.g., (Kara and Eker, 2004a; 2004b):

$$T_f(\omega) = \alpha_0 + \alpha \mathrm{sgn}(\omega), \tag{8.53}$$

$$T_f(\omega) = \alpha_0 + \left(\alpha_1 + \alpha_2 \mathrm{e}^{-\alpha_3|\omega|}\right)\mathrm{sgn}(\omega), \tag{8.54}$$

$$T_f(\omega) = \left(\alpha_1 + \alpha_2 \mathrm{e}^{-\alpha_3|\omega|}\right)\mathrm{sgn1}(\omega) + \left(\alpha_4 + \alpha_5 \mathrm{e}^{-\alpha_6|\omega|}\right)\mathrm{sgn2}(\omega), \tag{8.55}$$

$$\mathrm{sgn}(\omega) = \begin{cases} 1 & \text{for } \omega > 0, \\ 0 & \text{for } \omega = 0, \\ -1 & \text{for } \omega < 0, \end{cases}$$

$$\mathrm{sgn1}(\omega) = \begin{cases} 1 & \text{for } \omega \geqslant 0, \\ 0 & \text{for } \omega < 0, \end{cases} \qquad \mathrm{sgn2}(\omega) = \begin{cases} 0 & \text{for } \omega \geqslant 1, \\ -1 & \text{for } \omega < 0. \end{cases}$$

To built a discrete-time neural network model of the DC motor, the structure shown in Fig. 8.7 can be simplified further assuming that $G_2(s) \approx R_a/K_m$. Such a simplification is fully justified taking into account low frequency characteristics of the typical excitation $u_a(t)$. This leads to a block-oriented neural network model of the DC motor that comprises a linear dynamic part, represented by a single linear node with two tap delay lines, and a non-linear static element, represented by a multi-layer perceptron, in the feedback path (Fig. 8.8). The linear dynamic part of the model is described by the following linear difference equation:

$$\hat{\omega}(n) = -\sum_{m=1}^{n_a} \hat{a}_m \hat{\omega}(n-m) + \sum_{m=1}^{n_b} \hat{b}_m u(n-m), \tag{8.56}$$

$$u(n) = u_a(n) - \hat{y}(n), \tag{8.57}$$

where $\hat{a}_1, \ldots, \hat{a}_{n_a}, \hat{b}_1, \ldots, \hat{b}_{n_b}$ are the parameters of the linear dynamical model.

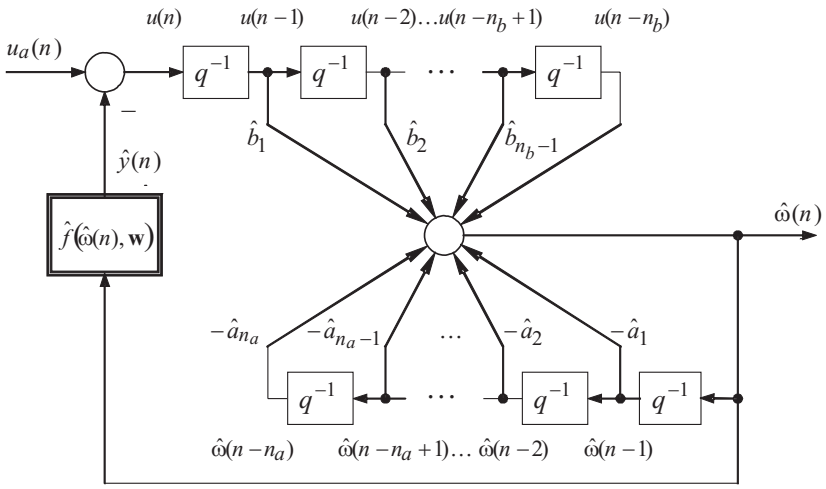


Fig. 8.8. Neural network model of the DC motor

Assuming that the multi-layer perceptron contains one hidden layer consisting of M non-linear nodes, its output $\hat{y}(n)$ can be expressed as

$$\hat{y}(n) = \sum_{j=1}^{M} w_{1j}^{(2)} \varphi\big(x_j(n)\big) + w_{10}^{(2)}, \tag{8.58}$$

$$x_j(n) = w_{j1}^{(1)} \hat{\omega}(n) + w_{j0}^{(1)}, \tag{8.59}$$

where $\varphi(\cdot)$ is the activation function, $w_{11}^{(1)}, \ldots, w_{M1}^{(1)}, w_{10}^{(1)}, \ldots, w_{M0}^{(1)}, w_{11}^{(2)}, \ldots, w_{1M}^{(2)}$, $w_{10}^{(2)}$ are the parameters (weights and biases) of the non-linear element model (Janczak, 2005).

A separately excited DC motor of rated voltage 24 V, rated current 2 A, rated speed 3000 rpm, and rated power 30 W, connected with a stiff shaft with an identical DC machine in the generator mode was used in the example. The learning set and the testing set consisted of 8000 and 4000 input-output pairs, respectively, acquired at the sampling interval of 0.05 s. The neural network model of the non-linear static element was composed of a single hidden layer containing 10 nodes of the hyperbolic tangent activation function. To train the neural network model, the Recursive Prediction Error (RPE) method (Janczak, 2005) was employed and 10 learning cycles were performed. To calculate the gradient of the model output, the sensitivity method was applied. In this method, partial derivatives of the model output with respect to its parameters are obtained by simulation of a set of linear difference equations (Janczak, 2003a; 2003b). The following control input was used in the experiment:

$$u_a(n) = \mathrm{sat}\big(u(n)\big),$$

$$u(n) = 16\sin(2\pi 0.03n + \pi/3) + 16\sin(2\pi 0.11n - \pi/7)r + 16\sin(2\pi 0.17n)$$

$$\mathrm{sat}\big(u(n)\big) = \begin{cases} u(n), & u(n) \leqslant 24, \\ 0, & u(n) > 24. \end{cases}$$

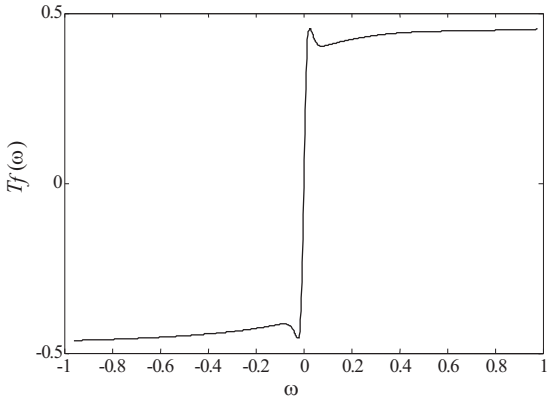


Fig. 8.9. Non-linear element characteristic

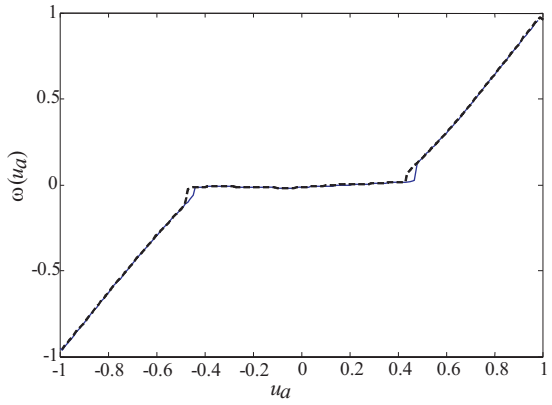


Fig. 8.10. Non-linear characteristic of the model

The non-linear element characteristic and the non-linear characteristic of the overall model are shown in Figs 8.9 and 8.10. The following evaluation of the transfer function $G_1(q^{-1})$ was obtained:

$$G_1(q^{-1}) = \frac{0.0666q^{-1} + 0.0042q^{-2}}{1 - 1.3962q^{-1} + 0.4304q^{-2}}.$$

To verify the modelling accuracy of the block-oriented model, two other models, i.e., a linear Output Error (OE) model and a Non-linear Neural Network Output Error (NNOE) model were identified using the RPE method and the Levenberg-Marquardt method, respectively. Both the OE model and the NNOE model were of the second order and the NNOE model contained one hidden layer composed of ten hyperbolic tangent nodes. The values of the mean square of the prediction error for both the training set and the learning set, given in Tab. 8.1, reveal the highest modeling accuracy of the NNOE model. The values of the mean square of the prediction error for the block-oriented model are only a little higher. Both the NNOE model and the block-oriented model are much more accurate than the OE model.

Table 8.1. Comparison of modelling accuracy. Mean-square of the prediction error

Data	OE	NNOE	Block-oriented model
Training set	1.82×10^{-2}	5.33×10^{-5}	6.68×10^{-5}
Testing set	2.03×10^{-2}	6.57×10^{-5}	7.66×10^{-5}

Architecture optimisation is one way to improve the quality of neural network models. It is a very important issue in neural network modeling as usually there is a large amount of redundant information contained in the weights of a fully connected neural network. A reduction in the amount of weights not only can markedly improve the generalisation properties but also makes the learning easier (Gupta *et al.*, 2003). It is usually performed by removing some insignificant weights from the network so as to retain the functional capability needed to model the system. Note that, for both the NNOE and block-oriented neural network models, the modeling results given in Tab. 8.1 have been obtained without any architecture optimisation. Building a block-oriented neural network model, which corresponds exactly to the structure shown in Fig. 8.7, is another perspective way to increase the model quality. Nevertheless, due to the appearance of the other linear dynamic block, such an approach will entail more complicated rules for gradient calculation even if $G_2(s)$ is assumed to be known.

8.4.2. Observer-based fault detection of an induction motor

The purpose of this section is to show the reliability and effectiveness of the EUIO presented in Section 8.2.2. The numerical example considered here is a fifth-order two-phase non-linear model of an induction motor, which has already been the subject of a large number of various control design applications (see (Boutayeb and Aubry, 1999) and the references therein). The complete discrete-time model in a stator-fixed (a,b) reference frame is

$$x_{1,k+1} = x_{1,k} + h\left(-\gamma x_{1k} + \frac{K}{T_r}x_{3k} + Kpx_{5k}x_{4k} + \frac{1}{\sigma L_s}u_{1k}\right), \tag{8.60}$$

$$x_{2,k+1} = x_{2,k} + h\left(-\gamma x_{2k} - Kpx_{5k}x_{3k} + \frac{K}{T_r}x_{4k} + \frac{1}{\sigma L_s}u_{2k}\right), \tag{8.61}$$

$$x_{3,k+1} = x_{3,k} + h\left(\frac{M}{T_r}x_{1k} - \frac{1}{T_r}x_{3k} - px_{5k}x_{4k}\right), \tag{8.62}$$

$$x_{4,k+1} = x_{4,k} + h\left(\frac{M}{T_r}x_{2k} + px_{5k}x_{3k} - \frac{1}{T_r}x_{4k}\right), \tag{8.63}$$

$$x_{5,k+1} = x_{5,k} + h\left(\frac{pM}{JL_r}(x_{3k}x_{2k} - x_{4k}x_{1k}) - \frac{T_L}{J}\right), \tag{8.64}$$

$$y_{1,k+1} = x_{1,k+1},\ y_{2,k+1} = x_{2,k+1}, \tag{8.65}$$

where $\boldsymbol{x}_k = [x_{1,k}, \ldots, x_{n,k}]^T = [i_{\mathrm{sak}}, i_{\mathrm{sbk}}, \psi_{\mathrm{rak}}, \psi_{\mathrm{rbk}}, \omega_k]^T$ represents the currents, the rotor fluxes, and the angular speed, respectively, while $\boldsymbol{u}_k = [u_{\mathrm{sak}}, u_{\mathrm{sbk}}]^T$ is the stator voltage control vector, p is the number of the pairs of poles, and T_L is the load torque. The rotor time constant T_r and the remaining parameters are defined as

$$T_r = \frac{L_r}{R_r}, \quad \sigma = 1 - \frac{M^2}{L_s L_r}, \quad K = \frac{M}{\sigma L_s L_r}, \quad \gamma = \frac{R_s}{\sigma L_s} + \frac{R_r M^2}{\sigma L_s L_r^2}, \tag{8.66}$$

where R_s, R_r and L_s, L_r are stator and rotor per phase resistances and inductances, respectively, and J is the rotor moment inertia.

The numerical values of the above parameters are as follows: $R_s = 0.18\ \Omega$, $R_r = 0.15\ \Omega$, $M = 0.068$ H, $L_s = 0.0699$ H, $L_r = 0.0699$ H, $J = 0.0586\ \mathrm{kgm}^2$, $T_L = 10$ Nm, $p = 1$, and $h = 0.1$ ms. The input signals are

$$u_{1,k} = 350\cos(0.03k), \quad u_{2,k} = 300\sin(0.03k). \tag{8.67}$$

The initial conditions for the system and the observer are $\boldsymbol{x}_k = \boldsymbol{0}$ and $\hat{\boldsymbol{x}}_k = [200, 200, 50, 50, 300]^T$, and $\boldsymbol{P}_0 = 10^3 \boldsymbol{I}$,

$$\begin{aligned} \boldsymbol{Q}_{k-1} &= 10^{10} \boldsymbol{\varepsilon}_{k-1}^T \boldsymbol{\varepsilon}_{k-1} \boldsymbol{I} + 0.001 \boldsymbol{I}, \\ \boldsymbol{R}_k &= 0.01 \boldsymbol{I}. \end{aligned} \tag{8.68}$$

Let us assume that the unknown input distribution matrix is

$$E = [1.2,\ 0.2,\ 2.4,\ 1,\ -1.6]^T, \tag{8.69}$$

and the corresponding unknown input is simulated by

$$d_k = 3.0\sin(0.5\pi k)\cos(0.03\pi k). \tag{8.70}$$

Thus, the system (8.16)–(8.17) is described using (8.60)–(8.65) and (8.69).

The following fault scenarios were considered:

***Case 1*:** Abrupt fault of the $y_{1,k}$ sensor:

$$f_{1,k} = \begin{cases} 0, & 500 < k < 140, \\ -0.1 y_{1,k}, & \text{otherwise}, \end{cases} \tag{8.71}$$

and $f_{2,k} = 0$.

***Case 2*:** Abrupt fault of the $u_{1,k}$ actuator:

$$f_{2,k} = \begin{cases} 0, & 500 < k < 140, \\ -0.2 u_{1,k}, & \text{otherwise}, \end{cases} \tag{8.72}$$

and $f_{1,k} = 0$.

Thus, the system is now described by

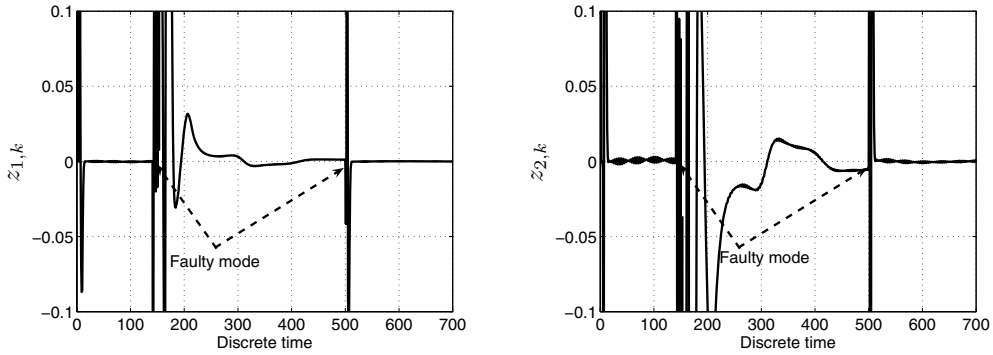
$$\boldsymbol{x}_{k+1} = \boldsymbol{g}\left(\boldsymbol{x}_k\right) + \boldsymbol{h}(\boldsymbol{u}_k) + \boldsymbol{E}_k\boldsymbol{d}_k + \boldsymbol{L}_{1,k}\boldsymbol{f}_k, \tag{8.73}$$

$$\boldsymbol{y}_{k+1} = \boldsymbol{C}_{k+1}\boldsymbol{x}_{k+1} + \boldsymbol{L}_{2,k+1}\boldsymbol{f}_{k+1}, \tag{8.74}$$

with (8.60)–(8.65), (8.69), $\boldsymbol{f}_k = [f_{1,k}, f_{2,k}]^T$, and

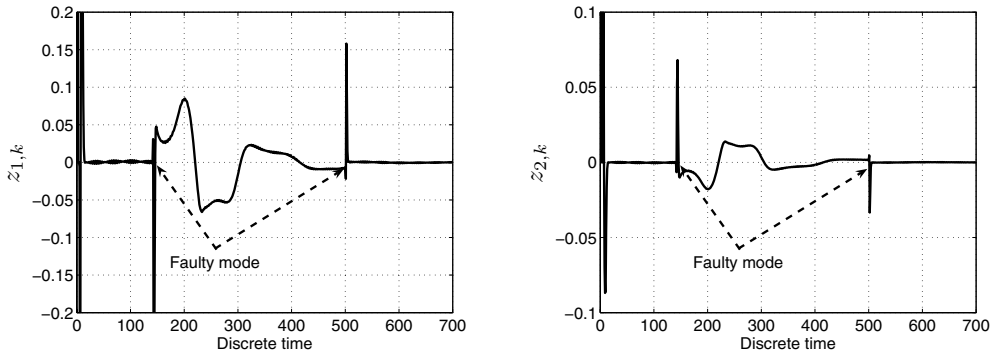
$$\boldsymbol{L}_{1,k} = \begin{bmatrix} \frac{1}{\sigma L_s} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T, \quad \boldsymbol{L}_{2,k} = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}. \tag{8.75}$$

From Figs. 8.11 and 8.12, it can be observed that the residual signal is sensitive to the faults under consideration, which confirms its reliability and abilities of unknown input decoupling. This, together with unknown input decoupling, implies that the process of fault detection becomes a relatively easy task.



(a) (b)

Fig. 8.11. Residuals for a sensor fault



(a) (b)

Fig. 8.12. Residuals for an actuator fault

8.5. Conclusions

From the point of view of engineering, it is clear that providing fast and reliable fault detection and isolation is an integral part of control design, particularly as far as the control of complex industrial systems is considered. Unfortunately, most of such systems exhibit non-linear behaviour, which makes it impossible to use the well-developed techniques for linear systems. If it is assumed that the system is linear, which is not true in general, and even if robust techniques for linear systems are used (e.g., unknown input observers), it is clear that such an approximation may lead to unreliable fault detection and, consequently, an early indication of faults which are developing is rather impossible. Such a situation increases the probability of the occurrence of faults, which can be extremely serious in terms of economic losses, environmental impact, or even human mortality. Indeed, robust techniques are able to tolerate a certain degree of model uncertainty. In other words, they are not robust to everything, i.e., are robust to an arbitrary degree of model uncertainty. This real world development pressure creates the need for new techniques which are able to tackle fault diagnosis of non-linear systems. In spite of the fact that the problem has been attacked from various angles by many authors and a number of relevant results have already been reported in the literature, there is no general framework which can be simply and conveniently applied to maintain fault diagnosis for non-linear systems.

Taking into account the above discussion, the main objective was to present selected solutions to this challenging problem. In particular, recent advances in robust observer- and neural network-based fault diagnosis were presented and carefully discussed. It is also worth noting that the selected techniques were illustrated with practical applications regarding modelling and fault diagnosis of non-linear systems.

References

Armstrong B. and de Wit C.C. (1995): *Friction Modeling and Compensation. The Control Handbook.* — New York: CRC Press.

Atkinson A.C. and Donev A.N. (1992): *Optimum Experimental Designs.* — New York: Oxford University Press.

Blanke M., Bogh S., Jorgensen R.B. and Patton R.J. (1994): *Fault detection for diesel engine actuator – a benchmark for FDI.* — Proc. 2-nd IFAC Symp. *Fault Detection, Supervision and Safety of Technical Processes, SAFEPROCESS*, Vol. 2, Espoo, Finland, pp. 498–506.

Blanke M., Kinnaert M., Lunze J. and Staroswiecki M. (2003): *Diagnosis and Fault-Tolerant Control.* — New York: Springer-Verlag.

Boutayeb M. and Aubry D. (1999): *A strong tracking extended Kalman observer for nonlinear discrete-time systems.* — IEEE Trans. Automatic Control, Vol. 44, No. 8, pp. 1550–1556.

Calado J.M.F., Sa da Costa J.M.G., Bartys M. and Korbicz J. (2006): *FDI approach to the DAMADICS benchmark problem based on qualitative reasoning coupled with fuzzy neural networks.* — Control Engineering Practice, Vol. 14, No. 6, pp. 685–698.

Chen J. and Patton R.J. (1999): *Robust Model Based Fault Diagnosis for Dynamic Systems.* — London: Kluwer Academic Publishers.

Delebecque F., Nikoukah R. and Rubio Scola H. (2003): *Test signal design for failure detection: A linear programming approach*. — Int. J. Appl. Math. and Comp. Sci., Vol. 13, No. 4, pp. 515–526.

Duch W., Korbicz J., Rutkowski L. and Tadeusiewicz R. (Eds.) (2000): *Biocybernetics and Biomedical Engineering 2000. Neural Networks*. — Warsaw: Akademicka Oficyna Wydawnicza, PLJ, Vol. 6, (in Polish).

Elman J.L. (1990): *Finding structure in time*. — Cognitive Science, Vol. 14, pp. 179–211.

Frank P.M. and Ding S.X. (1997): *Survey of robust residual generation and evaluation methods in observer-based fault detection systems*. — J. Process Control, Vol. 7, No. 6, pp. 403–424.

Frank P.M. and Köppen-Seliger B. (1997): *New developments using AI in fault diagnosis*. — Artificial Intelligence, Vol. 10, No. 1, pp. 3–14.

Frank P.M., Schreier G. and Garcia E.A. (1999): *Nonlinear observers for fault detection and isolation*, New Directions in Nonlinear Observer Design (Nijmeijer H. and T. Fossen, Eds.). — Berlin: Springer-Verlag.

Gertler J. (1998): *Fault Detection and Diagnosis in Engineering Systems*. — New York: Marcel Dekker.

Gupta M.M., Jin L. and Homma N. (2003): *Static and Dynamic Neural Networks. From Fundamentals to Advanced Theory*. — New Jersey: Wiley.

Haykin S. (1999): *Neural Networks. A Comprehensive Foundation*. — New Jersey: Prentice-Hall.

Isermann R. (2006): *Fault Diagnosis Systems. An Introduction from Fault Detection to Fault Tolerance*. — New York: Springer-Verlag.

Ivakhnenko A.G. and Mueller J.A. (1995): *Self-organizing of nets of active neurons*. — System Analysis Modelling Simulation, Vol. 20, pp. 93–106.

Janczak A. (2003a): *A comparison of four gradient learning algorithms for neural network Wiener models*. — Int. J. Systems Science, Vol. 34, No. 1, pp. 21–35.

Janczak A. (2003b): *Neural network approach to identification of Hammerstein systems*. — Int. J. Control, Vol. 76, No. 17, pp. 1749–1766.

Janczak A. (2005): *Identification of Nonlinear Systems Using Neural Networks and Polynomial Models. A Block-oriented Approach*. — Lecture Notes in Control and Information Sciences, Vol. 310, Berlin: Springer-Verlag.

Jordan M.I. (1986): *Attractor dynamic and parallelism in a connectionist sequential machine*. — Proc. 8-th Annual Conf. *Cognitive Science Society*, Hillsdale: Erlbaum, pp. 531–546.

Kara T. and Eker I. (2004a): *Nonlinear closed-loop direct identification of a DC motor with load for low speed two-directional operation*. — Electrical Engineering, Vol. 86, No. 2, pp. 87–96.

Kara T. and Eker I. (2004b): *Nonlinear modeling and identification of a DC motor for bidirectional operation with real time experiments*. — Energy Conservation and Management, Vol. 45, No. 7-8, pp. 1087–1106.

Karpenko M., Sepehri N. and Scuse D. (2003): *Diagnosis of process valve actuator faults using multilayer neural network*. — Control Engineering Practice, Vol. 11, pp. 1289–1299.

Köppen-Seliger B. and Frank P.M. (1999): *Fuzzy logic and neural networks in fault detection*, In: Fusion of Neural Networks, Fuzzy Sets, and Genetic Algorithms (L. Jain and N. Martin, Eds.). — New York: CRC Press, pp. 169–209.

Korbicz J. (2004): *Computer designing of diagnostic systems*, In: Engineering of Machine Diagnostics (Żółtowski B. and Cz. Cempel, Eds.). — Radom: Instytut Technologii Eksploatacji, pp. 453–474, (in Polish).

Korbicz J. (2006): *Fault detection using analytical and soft computing methods*. — Bulletin of the Polish Academy of Sciences: Technical Sciences, Vol. 54, No. 1, pp. 75–88.

Korbicz J., Kościelny J.M., Kowalczuk Z. and Cholewa W. (Eds.) (2002): *Processes Diagnosis. Models, Methods of Artificial Intelligence, Applications*. — Warsaw: Wydawnictwo Naukowo Techniczne, WNT, (in Polish).

Korbicz J., Kościelny J.M., Kowalczuk Z. and Cholewa W. (Eds.) (2004): *Fault Diagnosis. Models, Artificial Intelligence, Applications*. — Berlin: Springer-Verlag.

Korbicz J. and Mrugalski M. (2007): *Confidence estimation of GMDH neural networks and its application in fault detection systems*. — Int. J. System Science, (accepted).

Korbicz J., Obuchowicz A. and Uciński D. (1994): *Artificial Neural Networks. Fundations and Applications*. — Warsaw: Akademicka Oficyna Wydawnicza, (in Polish).

Korbicz J., Patan K. and Obuchowicz A. (2001): *Neural network fault detection system for dynamic processes*. — Bulletin of the Polish Academy of Sciences. Technical Sciences., Vol. 49, No. 2, pp. 301–321.

Kościelny J.M. (2001): *Diagnostics of Automatic Industrial Processes*. — Warsaw: Akademicka Oficyna Wydawnicza EXIT, (in Polish).

Kowal M. (2005): *Optimization of Neuro-fuzy Structures in Technical Diagnostics*. — Lecture Notes in Control and Computer Science, Vol. 9, University of Zielona Góra Press.

Milanese M., Norton J., Piet-Lahanier H. and Walter E. (1996): *Bounding Approaches to System Identification*. — New York: Plenum Press,.

Mrugalski M. (2004): *Neural Network Based Modelling of Non-linear Systems in Fault Detection Schemes*. — Ph.D. thesis, Faculty of Electrical Engineering, Computer Science and Telecommunications, University of Zielona Góra, (in Polish).

Mrugalski M. and Korbicz J. (2005): *Robust fault detection via GMDH neural networks*. — Proc. 16-th *IFAC World Congress*, Prague, Czech Republic, CD-ROM.

Mrugalski M. and Korbicz J. (2006): *Application of the MLP neural network to the robust fault detection*. — Proc. 6-th IFAC Symp. *Fault Detection Supervision and Safety of Technical Processes, SAFEPROCESS*, Beijing, China, CD-ROM.

Mueller J.E. and Lemke F. (2000): *Self-organising Data Mining*. — Hamburg: Libri.

Narendra K.S. and Parthasarathy K. (1990): *Identification and control of dynamical systems using neural networks*. — IEEE Trans. Neural Networks, Vol. 1, pp. 12–18.

Nelles O. (2001): *Nonlinear System Identification. From Classical Approaches to Neural Networks and Fuzzy Models*. — Berlin: Springer-Verlag.

Norgard M., Ravn O., Poulsen N. and Hansen L. (2000): *Networks for Modelling and Control of Dynamic Systems*. — London: Springer-Verlag.

Osowski S. (2006): *Artificial Neural Networks for Information Processing*. — Warsaw: Oficyna Wydawnicza Politechniki Warszawskiej, (in Polish).

Patan K. (2007): *Stability analysis and the stabilization of a class of discrete-time dynamic neural networks*. — IEEE Trans. Neural Networks, Vol. 18, (accepted).

Patan K., Korbicz J. and Prętki P. (2005): *Global stability conditions of locally recurrent neural networks*. — Lecture Notes in Computer Science, Vol. 3697, pp. 191–196.

Patan K. and Parisini T. (2005): *Identification of neural dynamic models for fault detection and isolation: The case of a real sugar evaporation process.* — J. Process Control, Vol. 15, pp. 67–79.

Patton R.J., Frank P.M. and Clark R.N. (2000): *Issues of Fault Diagnosis for Dynamic Systems.* — Berlin: Springer-Verlag.

Patton R.J., Korbicz J. and Lesecq S. (Eds.) (2006): *A Benchmark Study of Fault Diagnosis for an Industrial Actuator.* — Control Engineering Practice, Vol. 14, No. 6, pp. 575–717.

Patton R.J., Korbicz J., Witczak M. and Uppal F. (2005): *Combined computational intelligence and analytical methods in fault diagnosis*, In: Intelligent Control Systems using Computational Intelligence Techniques (Rauno A.E. (Ed.). — London: The IEE Press, pp. 349–392.

Ruano A.E. (2005): *Intelligent Control Systems using Computational Intelligence Techniques.* — London: The IEE Press.

Rutkowski L. (2005): *Methods and Techniques of Artificial Intelligence.* — Warsaw: Wydawnictwo Naukowe PWN, (in Polish).

Tsoi A.C. and Back A.D. (1994): *Locally recurrent globally feedforward networks: A critical review of architectures.* — IEEE Trans. Neural Networks, Vol. 5, pp. 229–239.

Uciński D. (2005): *Optimal Measurement Methods for Distributed Parameter System Identification.* — Boca Raton: CRC Press.

Uppal F., Patton R.J. and Witczak M. (2006): *A neuro-fuzzy multiple-model observer approach to robust fault diagnosis based on the DAMADICS benchmark problem.* — Control Engineering Practice, Vol. 14, No. 6, pp. 699–717.

Walter E. and Pronzato L. (1996): *Identification of Parametric Models from Experimental Data.* — London: Springer.

Wang S.H., Davison E.J. and Dorato P. (1975): *Observing the states of systems with unmeasurable disturbances.* — IEEE Trans. Automatic Control, Vol. 20, No. 5, pp. 716–717.

Witczak M. (2006a): *Advances in model-based fault diagnosis with evolutionary algorithms and neural networks.* — Int. J. Appl. Math. and Comp. Sci., Vol. 16, No. 1, pp. 85–99.

Witczak M. (2006b): *Toward the training of feed-forward neural networks with the D-optimum input sequence.* — IEEE Trans. Neural Networks, Vol. 17, No. 2, pp. 357–373.

Witczak M. and Korbicz J. (2004): *Observers and genetic programming in the identification and fault diagnosis of non-linear dynamic systems*, In: Fault Diagnosis. Models, Artificial Intelligence, Applications (Korbicz J., Kościelny J.M., Kowalczuk Z. and Cholewa W., Eds.). — Berlin: Springer-Verlag, pp. 457–509.

Witczak M. and Korbicz J. (2006): *Design of observers for Lipschitz non-linear discrete-time systems.* — Proc. 14-th IFAC Symp. *System Identification, SYSID*, Newcastle, Australia, pp. 985–990, CD-ROM.

Witczak M., Korbicz J., Mrugalski M. and Patton R.J. (2006a): *A GMDH neural network-based approach to robust fault diagnosis: application to the DAMADICS benchmark problem.* — Control Engineering Practice, Vol. 14, No. 6, pp. 671–683.

Witczak M., Korbicz J. and Puig V. (2006b): *An LMI approach to designing observers and unknown input observers for nonlinear systems.* — Proc. 6-th IFAC Symp. *Fault Detection Supervision and Safety of Technical Processes, SAFEPROCESS*, Beijing, China, CD-ROM.

Witczak M., Obuchowicz A. and Korbicz J. (2002): *Genetic programming based approaches to identification and fault diagnosis of non-linear dynamic systems.* — Int. J. Control, Vol. 75, No. 13, pp. 1012–1031.

Witczak M., Pretki P., Korbicz J. and Puig V. (2006c): *Design of an extended unknown input observer.* — Proc. Int. Workshop *Advanced Control Diagnosis, ACD*, Nancy, France, CD-ROM.

Zhang J.P.D.R. and Ellis J.E. (1991): *A self-learning fault diagnosis system.* — Trans. Institute of Measurements and Control, Vol. 13, pp. 29–35.

Zolghadri A., Henry D. and Monsion M. (1996): *Design of nonlinear observers for fault diagnosis. a case study.* — Control Engineering Practice, Vol. 4, No. 11, pp. 1535–1544.

Chapter 9

SOLVING OPTIMIZATION TASKS IN THE CONSTRUCTION OF DIAGNOSTIC SYSTEMS

Andrzej OBUCHOWICZ*, Andrzej PIECZYŃSKI*
Marek KOWAL*, Przemysław PRĘTKI*

9.1. Introduction

The core of the Fault Detection and Isolation (FDI) system is the so-called model-based approach. In the general case, this concept can be implemented using various kinds of models: analytical, knowledge-based and data-based ones (Köppen-Seliger and Frank, 1999), which are used to model a diagnosed system working in normal-operation or faulty conditions. Conventional model-based fault detection techniques make use of analytical or quantitative models (Patton, 1993), mostly in the framework of observers or Kalman filters (Chen and Patton, 1999). The dynamic behaviour of the system is described by differential equations or transfer functions together with the respective parameter values. Unfortunately, the analytical model-based approach is usually restricted to simpler systems described by linear models. When there are no mathematical models of the diagnosed system or the complexity of the dynamic model increases and the task of modeling is hard, an analytical model cannot be applied in the fault diagnosis system nor give satisfactory results. Currently many efforts are made to use knowledge-based or qualitative or data-based models. They represent system behaviour in terms of heuristic or qualitative knowledge (Frank, 1990). The relationship between inputs and outputs may be described by a rule base (Amann and Frank, 1997) or by a set of parameters that have to be determined during an identification stage based on the learning data set. In this case data-based models, such as neural networks (Korbicz *et al.*, 1998; Patan *et al.*, 1999), fuzzy sets (Frank and Köppen-Seliger, 1997; Kiupel and Frank, 1993; Pieczyński, 2003), the

* Institute of Control and Computation Engineering
e-mails: {a.obuchowicz, a.pieczynski, m.kowal, p.pretki}@issi.uz.zgora.pl

genetic approach (Chen and Patton, 1999; Obuchowicz, 2003; Witczak *et al.*, 1999) or their combination (Köppen-Seliger and Frank, 1999; Korbicz *et al.*, 1999), can be considered.

Although there are many techniques for constructing nonanalytical models, in one way or another, they finally boil down to several global optimization problems, like searching for an optimal model structure, the allocation of model parameters etc. They are nonlinear, multi-modal, usually multi-objective, so that conventional "local" optimization methods are insufficient to solve them. In recent years, direct search techniques, which are problem-independent, have been widely used in optimization. Unlike calculus-based methods (gradient descent, etc.), direct search algorithms do not require the use of derivatives. Gradient-descent methods work well when the objective surface is relatively smooth, with few local minima. However, real-world data are often multimodal and contaminated by noise which can further distort the objective surface.

Evolutionary Algorithms (EAs) seem to be particularly attractive direct search methods. EAs are a broad class of stochastic optimization algorithms inspired by some biological processes which allow populations of organisms to adapt to their surrounding environment (Bäck *et al.*, 1997; Goldberg, 1989; Michalewicz, 1996; Obuchowicz, 2003). All natural species survive by adapting themselves to the environment. EA search combines the Darwinian survival of the fittest strategy to eliminate unfit characteristics and uses a random information exchange and mutation, with the exploitation of the knowledge contained in previous populations, to effect the search mechanism with surprising power and speed.

9.2. Optimization tasks in FDI system design

There is rich literature of the soft computing methods approach to FDI systems design. The obtained solutions (Korbicz *et al.*, 1998; Obuchowicz, 1999b; 2003; Obuchowicz and Korbicz, 2002; Witczak *et al.*, 1999; 2002) (Fig. 9.1) show high efficiency of diagnosis systems whose design had been aided by soft computing methods.

Among artificial intelligence methods applied to design fault diagnosis systems, Artificial Neural Networks (ANNs) are very popular, and they are used for building

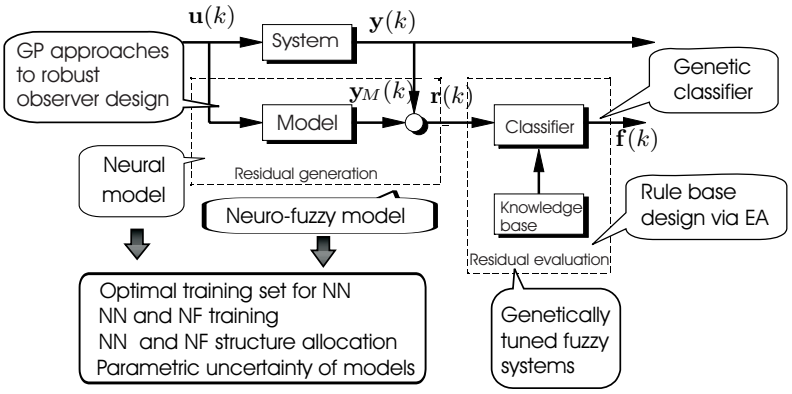


Fig. 9.1. Selected optimization tasks in the FDI system design

neural models as well as neural classifiers (Frank and Köppen-Seliger, 1997; Köppen-Seliger and Frank, 1999; Korbicz *et al.*, 1998). But the construction of the neural model corresponds to two basic optimization problems: the optimization of a neural network architecture and its training process, i.e., searching for the optimal set of network free parameters. Evolutionary algorithms are a very useful tool to solve both problems, especially in the case of dynamic neural networks (Korbicz *et al.*, 1998; Obuchowicz, 1999b; 2000; 2003).

One of the main areas in the process diagnostic field is research concerning an effective use of neuro-fuzzy models (Chen and Patton, 1999; Korbicz *et al.*, 2004). The intensive development of design algorithms and learning methods results in many applications of Neuro-Fuzzy (N-F) networks in different areas of fault diagnosis (Calado *et al.*, 2003; Kowal, 2001; 2005; Kowal and Korbicz, 2002a; 2002b; 2003; Mendes *et al.*, 2002). The attractiveness of neuro-fuzzy methods arises from the fact that they can be employed when there are no phenomenological models available. Since neuro-fuzzy networks constitute an extension of classical neural networks, many algorithms invented for neural networks, e.g., training algorithms can be adapted to neuro-fuzzy networks. The basic structure of a fuzzy system consists of three blocks: the block of fuzzification, the block of inference with the base of knowledge, and the block of defuzzification (Piegat, 2003). On the basis of the structure of neuro-fuzzy networks, it is possible to qualify the form and methods of optimization of the network. From this point of view, the number of partitions is optimized through an examination (Pieczyński, 2003). The shape of the fuzzy set membership curve has major meaning as well. To solve this task, two methods can be applied. The first one is based on the list of standard curves which are defined in literature (Piegat, 2003). The second way uses the general Gauss function (Pieczyński and Obuchowicz, 2004).

The efficiency of fault detection systems in the case of multi-dimensional symptom vectors may be improved by pre-processing which leads to the partitioning of the symptom domain into subdomains (clusters). Among many well-known preprocessing methods, EAs characterize high clustering performance. Let us concentrate on multi-dimensional real data that form a set of the so-called training pairs:

$$\mathcal{T}_d = \big\{ \boldsymbol{p}_q = (\boldsymbol{x}^q, y^q) \in \mathbb{R} \mid q = 1, \ldots, p \big\}. \tag{9.1}$$

The goal is to perform an evolutionary cluster analysis of data in $\mathcal{T}_d$ to get at the end a partitioning of $\mathcal{T}_d$. The number of clusters is not known in advance. To evaluate each off-spring cluster in the population, different local fitness functions may be used. They could be the maximal distance of the training pair of the cluster from the cluster centroid, or a mean variation of all training pairs in the cluster. Based on the local fitness function of the cluster, one can build a global fitness function (Kosiński *et al.*, 1998).

A fuzzy inference system is often used as a universal approximator for problems of multi-dimensional data or as a controller for some industrial applications (Köppen-Seliger and Frank, 1999). The fuzzy modelling approach consists of two kinds of problems, i.e., configuring fuzzy rules and optimizing the shapes of membership functions, which are considered to be combinatorial and numerical optimization problems, respectively. The EA is able to be applied to both of these problems. In many research works, however, the EA is applied only to optimize the configuration of fuzzy rules,

while another optimization algorithm, such as the steepest descent method, is applied to optimize the shapes of membership functions.

The application of artificial intelligence techniques leads to the concept of the fault diagnosis expert system where analytical and heuristic information as well as knowledge processing are combined (Frank and Köppen-Seliger, 1997). The expert system for fault diagnosis consists of a knowledge base which usually includes a rule base. The construction of the rule base is the main problem for knowledge engineers, as it has to implement, usually out of order, incomplete and heuristic knowledge of a human expert. In this case fuzzy techniques seem to be an effective tool to build the knowledge base. Unfortunately, there are many fault diagnosis problems for which the human expert knowledge is insufficient and automatic optimal selection of the rule base is needed. Because of the exponential complexity of the problem of optimal searching there are no possibilities of using a total review method. In this case, techniques of Genetic Algorithms (GAs) and Genetic Programming (GP) (Koza, 1992) may become very effective tools, assuming that decision rules are a set of complexes (Skowroński, 1998). Each complex is a conjunction of selectors and each selector is a disjunction of discrete attribute values. In this case, the population of individuals is built of vectors of selectors. The GA composes the rule base from the sets of attributes and their values. In order to use GP to create the rule base, two sets have to be defined. The first one, the terminal set, contains all possible premises and conclusions, while the second one contains logic operators. Each rule is represented by a structured tree, and GP is used to find the best sets of rules. Contrary to the GA-based approach, where only simple rules (triples) are considered, the GP-based approach makes it possible to use arbitrary complex rules (Koza, 1992).

If physical models are used, the identification problem reduces to the estimation of some parameters. This estimation does not seem to be a difficult problem because these parameters have usually physical interpretations. GP approaches to the modeling of dynamic nonlinear systems include the choice of the gain matrix of the robust nonlinear observer (Witczak *et al.*, 1999), searching for the MIMO NARX model (*Multi Input Multi Output Nonlinear AutoRegresive with eXogenous variable*) (Witczak and Korbicz, 2000), the selection of the state space representation of the system (Witczak *et al.*, 2002), or via Extended Unknown Input Observer (EUIO) design (Witczak *et al.*, 2002) (Fig. 9.2).

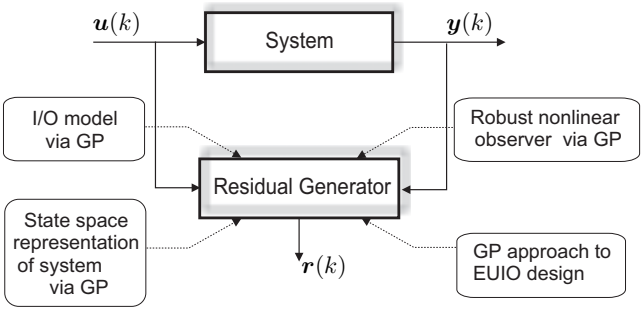


Fig. 9.2. GP implementation considered in this chapter

9.3. Genetic programming approaches to symptom extraction systems

9.3.1. Input/output representation of the system via GP

Knowing that the diagnosed system exhibits nonlinear characteristics, a choice of the nonlinear model set must be made. In this section, an NARX (*Nonlinear AutoRegresive with eXogenous variable*) was selected as the foundation for identification methodology. The MIMO NARX model has the following form:

$$\begin{aligned}\hat{y}_{i,k} = g_i(&\hat{y}_{1,k-1}, \ldots, \hat{y}_{1,k-n_{1,y}}, \ldots, \hat{y}_{m,k-1}, \ldots, \hat{y}_{m,k-n_{m,y}},\\ &u_{1,k-1}, \ldots, u_{1,k-n_{1,u}}, \ldots, u_{r,k-1}, \ldots, u_{r,k-n_{r,u}}, \boldsymbol{p}_i),\\ &i = 1, \ldots, m.\end{aligned} \tag{9.2}$$

Thus the system output is given by

$$\boldsymbol{y}_k = \hat{\boldsymbol{y}}_k + \varepsilon_k, \tag{9.3}$$

where ε_k consists of a structural deterministic error, caused by the model-reality mismatch, and the measurement noise $\boldsymbol{v}_k$. The problem is to determine an unknown function $\boldsymbol{g}(\cdot) = (g_1, \ldots g_m)$ and to estimate the corresponding parameters vector $\boldsymbol{p} = (\boldsymbol{p}_1, \ldots, \boldsymbol{p}_m)$.

One possible solution to this problem is the genetic programming approach. A tree is the main ingredient underlying the GP algorithm. In order to adapt GP to system identification it is necessary to represent the model (9.2) as a tree, or a set of trees. Indeed, as is shown in Fig. 9.3, the MISO NARX model can be easily put in the form of a tree, and hence to build the MIMO model (9.2) it is necessary to use m trees. In such a tree (see Fig. 9.3), two sets can be distinguished, namely, the terminal T and function F sets. The language of the trees in GP is formed by the

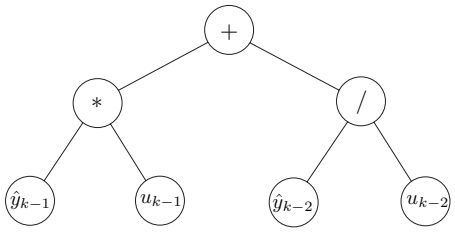


Fig. 9.3. Exemplary GP tree representing the model $\hat{y}_k = \hat{y}_{k-1}u_{k-1} + \hat{y}_{k-2}/u_{k-2}$

user-defined function F set and the terminal T set, which form the nodes of the trees. The functions should be chosen so that they are *a priori* useful in solving the problem, i.e., any knowledge concerning the system under consideration should be included in the function set. This function set is very important and should be universal enough to be capable of representing a wide range of nonlinear systems. The terminals are usually variables or constants. In (Esparcia-Alcazar, 1998), a tree representation is

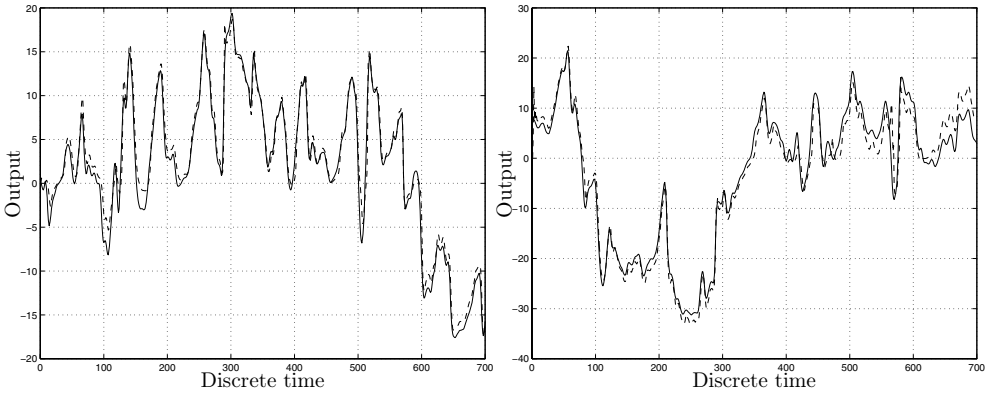


Fig. 9.4. System (solid line) and model (dashed line) output for the identification (left) and validation (right) data sets

extended by the so-called *node gains*. A node gain is a numerical parameter associated with the node, which multiplies its output value.

One of the best known criteria which can be employed to select the model structure and to estimate its parameters is the Akaike Information Criterion (AIC) (Walter and Pronzato, 1997), where the following quality index is minimized:

$$J_{AIC}(M_i) = \frac{1}{2} j\big(M_i(\hat{\boldsymbol{p}}^i)\big) + \frac{1}{n_T} \dim \boldsymbol{p}^i, \tag{9.4}$$

where

$$j\big(M_i(\boldsymbol{p}^i)\big) = \ln \det \sum_{k=1}^{n_T} \boldsymbol{\varepsilon}_k \boldsymbol{\varepsilon}_k^T, \tag{9.5}$$

and $\hat{\boldsymbol{p}}^i = \arg\min_{\boldsymbol{p}^i} j(M_i(\boldsymbol{p}^i))$ are the obtained using the identification data set of n_T pairs of input/output measurements. The GP algorithm was successfully applied to identify the input-output model of the evaporation station at the Lublin Sugar Factor S.A. (Poland) (DAMADICS, 2002). Figure 9.4 illustrates the obtained results (Witczak *et al.*, 2002).

9.3.2. Choice of the gain matrix for the robust nonlinear observer

The solution of the diagnosed object modeling presented in the previous subsection possesses a disadvantage. Usually, the parameters of the obtained GP model have no physical interpretations. In this subsection a nonlinear state observer designing methodology based on the classical approach and a GP technique and proposed by Witczak and co-workers (1999) is presented.

Consider the nonlinear discrete system

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{f}(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k), \\ \boldsymbol{y}_k &= \boldsymbol{h}(\boldsymbol{x}_k, \boldsymbol{v}_k), \end{aligned} \tag{9.6}$$

where $\boldsymbol{u}_k$ is the input, $\boldsymbol{y}_k$ is the output, $\boldsymbol{x}_k$ is the state, $\boldsymbol{w}_k$ and $\boldsymbol{v}_k$ represent the process and measurement noise, and $\boldsymbol{h}(\cdot)$, $\boldsymbol{f}(\cdot)$ are nonlinear functions.

The problem is to estimate the state $\boldsymbol{x}_k$ of the system (9.6), where a set of measured inputs and outputs and the model of the system are given. The classical methods using different kinds of approximation are often applied (Korbicz and Bidyuk, 1993), and can be given as follows:

$$\begin{aligned} \hat{\boldsymbol{x}}_k &= \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}_k \boldsymbol{\varepsilon}_k^-, \\ \boldsymbol{\varepsilon}_k^- &= \boldsymbol{y}_k - \boldsymbol{h}(\hat{\boldsymbol{x}}_k^-, \boldsymbol{v}_k), \end{aligned} \tag{9.7}$$

where $\boldsymbol{\varepsilon}_k^-$ denotes the *a priori* output error, $\hat{\boldsymbol{x}}_k^-$is an *a priori* state estimate, $\hat{\boldsymbol{x}}_k$ is a state estimate and $\boldsymbol{K}_k$ is the gain matrix.

The gain matrix $\boldsymbol{K}_k$ of the observer (9.7) can be searched for by various methods (e.g., the Kalman filter, the Luenberger observer, etc.) which, in a large majority, consist of constant elements and are not robust to model uncertainties. In (Witczak *et al.*, 1999), the gain matrix is composed of certain functions, i.e., each entry of the gain matrix is a function which depends on the *a priori* output error and the system input. Therefore, it can be written as follows:

$$\hat{\boldsymbol{x}}_k = \hat{\boldsymbol{x}}_k^- + \boldsymbol{K}(\varepsilon_k^-, \boldsymbol{u}_k)\varepsilon_k^-. \tag{9.8}$$

Thus the main goal is to obtain an appropriate form of $\boldsymbol{K}(\varepsilon_k^-, \boldsymbol{u}_k)$ based on a set of measured outputs and inputs and the mathematical model of the system. Even if the mathematical model is uncertain and/or the initial state is far from its expected value, it seems possible to obtain $\boldsymbol{K}(\varepsilon_k^-, \boldsymbol{u}_k)$ to ensure the best fitness to the real system. For that purpose, a GP technique is exploited, where the gain matrix is obtained off-line from a randomly created population by means of an evolutionary process.

9.3.3. GP approach to the state-space representation of the system

Let us consider the following class of nonlinear discrete-time systems:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k) + \boldsymbol{w}_k, \\ \boldsymbol{y}_{k+1} &= \mathbf{C}\boldsymbol{x}_{k+1} + \boldsymbol{v}_k. \end{aligned} \tag{9.9}$$

Assume that the function $\boldsymbol{g}(\cdot)$ has the form

$$\boldsymbol{g}(\boldsymbol{x}_k, \boldsymbol{u}_k) = \mathbf{A}(\boldsymbol{x}_k)\boldsymbol{x}_k + \boldsymbol{h}(\boldsymbol{u}_k). \tag{9.10}$$

Thus, the state-space model of the system (9.9) can be expressed as

$$\begin{aligned} \hat{\boldsymbol{x}}_{k+1} &= \mathbf{A}(\hat{\boldsymbol{x}}_k)\hat{\boldsymbol{x}}_k + \boldsymbol{h}(\boldsymbol{u}_k), \\ \hat{\boldsymbol{y}}_{k+1} &= \mathbf{C}\hat{\boldsymbol{x}}_{k+1}. \end{aligned} \tag{9.11}$$

Without loss of generality, it is possible to assume that

$$\mathbf{A}(\hat{\boldsymbol{x}}_k) = \operatorname{diag}(a_{i,i}(\hat{\boldsymbol{x}}_k) \mid i = 1, 2, \ldots, n). \tag{9.12}$$

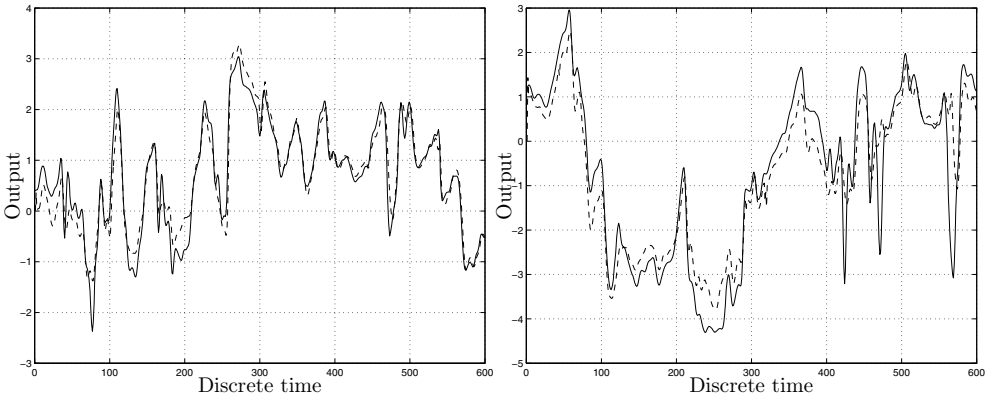


Fig. 9.5. System (solid line) and model (dashed line) output for the identification (left) and validation (right) data sets

The problem reduces to identifying the nonlinear functions $a_{i,i}(\hat{\boldsymbol{x}}_k), h_i(\boldsymbol{u}_k)$ $(i = 1, \ldots, n)$, and the matrix $\mathbf{C}$. Assuming $\max_{i=1,\ldots,n} |a_{i,i}(\hat{\boldsymbol{x}}_k)| < 1$, it can be shown (Witczak *et al.*, 2002) that the model (9.11) is globally asymptotically stable. This implies that $a_{i,i}(\hat{\boldsymbol{x}}_k)$ should have the following structure:

$$a_{i,i}(\hat{\boldsymbol{x}}_k) = \tanh\big(s_{i,i}(\hat{\boldsymbol{x}}_k)\big), \quad i = 1, \ldots, n, \tag{9.13}$$

where $\tanh(\cdot)$ is a hyperbolic tangent function, and $s_{i,i}(\hat{\boldsymbol{x}}_k)$ is a function to be determined.

In order to identify $s_{i,i}(\hat{\boldsymbol{x}}_k), h_i(\boldsymbol{u}_k)$ $(i = 1, \ldots, n)$ and the matrix $\mathbf{C}$, the GP algorithm is applied. The fitness function is defined by (9.4).

The GP algorithm was successfully applied to build a model of the apparatus at the Lublin Sugar Factor S.A. (Poland) (DAMADICS, 2002). Figure 9.5 illustrates obtained results (Witczak *et al.*, 2002).

9.3.4. GP approach to EUIO design

Regardless of the identification method used, there is always the problem of model uncertainty, i.e., the model-reality mismatch. To overcome this problem, many approaches have been proposed (Chen and Patton, 1999; Patton *et al.*, 2000). Undoubtedly, the most common approach is to use robust observers, such as the Unknown Input Observer (UIO) (Chen and Patton, 1999; Patton and Chen, 1997; Patton *et al.*, 2000), which can tolerate a certain degree of model uncertainties, and hence increase the reliability of fault diagnosis. Unfortunately, the design procedure of Nonlinear Unknown Input Observers (NUIOs) (Patton *et al.*, 2000) is usually very complex, even for simple laboratory systems (Zolghardi *et al.*, 1996).

Witczak and co-workers (2002) proposed a modified version of the well-known UIO, which can be applied to linear stochastic systems to form a nonlinear deterministic observer, the so-called extended unknown input observer. Moreover, it is shown that the convergence of the proposed observer is ensured under certain conditions and

the convergence rate can be dramatically increased, when compared to the classical approach, by the application of the genetic programming technique.

Let us consider the class of nonlinear systems which can be modeled by the following equations:

$$\begin{aligned} \boldsymbol{x}_{k+1} &= \boldsymbol{g}(\boldsymbol{x}_k) + \boldsymbol{h}(\boldsymbol{u}_k) + \boldsymbol{L}_{1,k}\boldsymbol{f}_k + \boldsymbol{E}_k\boldsymbol{d}_k, \\ \boldsymbol{y}_{k+1} &= \boldsymbol{C}_{k+1}\boldsymbol{x}_{k+1} + \boldsymbol{L}_{2,k+1}\boldsymbol{f}_{k+1}, \end{aligned} \tag{9.14}$$

where $\boldsymbol{g}(\boldsymbol{x}_k)$ is assumed to be continuously differentiable with respect to $\boldsymbol{x}_k$. This leads to the following structure of the EUIO:

$$\begin{aligned} \hat{\boldsymbol{x}}_{k+1/k} &= \boldsymbol{g}(\hat{\boldsymbol{x}}_k) + \boldsymbol{h}(\boldsymbol{u}_k), \\ \hat{\boldsymbol{x}}_{k+1} &= \hat{\boldsymbol{x}}_{k+1/k} + \boldsymbol{H}_{k+1}\varepsilon_{k+1/k} + \boldsymbol{K}_{1,k+1}\varepsilon_k. \end{aligned} \tag{9.15}$$

Employing the Lyapunov approach to convergence analysis of the EUIO it can be proved that the domain of attraction significantly depends on the covariance matrices $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$ of the process $\boldsymbol{w}_k$ and measurement $\boldsymbol{v}_k$ noise, respectively. Unfortunately, an analytical derivation of the $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$ matrices seems to be an extremely difficult problem. However, it is possible to set the above matrices as $\boldsymbol{Q}_{k-1} = \beta_1\boldsymbol{I}$, $\boldsymbol{R}_k = \beta_2\boldsymbol{I}$, with β_1 and β_2 large enough. On the other hand, it is well-known that the convergence rate of such an EKF-like approach can be increased by an appropriate selection of the covariance matrices $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$, i.e., the more accurate (near "true" values) the covariance matrices, the better the convergence rate. This means that in the deterministic case ($\boldsymbol{w}_k = \boldsymbol{0}$ and $\boldsymbol{v}_k = \boldsymbol{0}$) both matrices should be zero. Unfortunately, such an approach usually leads to the divergence of the observer as well as to other computational problems. To tackle this problem, a compromise between the convergence and the convergence rate should by established. This can be easily done by setting the instrumental matrices as

$$\begin{aligned} \boldsymbol{Q}_{k-1} &= \beta_1\boldsymbol{\varepsilon}_{k-1}^T\boldsymbol{\varepsilon}_{k-1}\boldsymbol{I} + \delta_1\boldsymbol{I}, \\ \boldsymbol{R}_k &= \beta_2\boldsymbol{\varepsilon}_k^T\boldsymbol{\varepsilon}_k\boldsymbol{I} + \delta_2\boldsymbol{I}, \end{aligned} \tag{9.16}$$

with β_1, β_2 large enough, and δ_1, δ_2 small enough. Although this approach is very simple, it is possible to increase the convergence rate further. Indeed, the instrumental matrices can be set as follows:

$$\boldsymbol{Q}_{k-1} = q^2(\boldsymbol{\varepsilon}_{k-1})\boldsymbol{I}, \tag{9.17}$$

$$\boldsymbol{R}_k = r^2(\boldsymbol{\varepsilon}_k)\boldsymbol{I}, \tag{9.18}$$

where $q(\boldsymbol{\varepsilon}_{k-1})$ and $r(\boldsymbol{\varepsilon}_k)$ are nonlinear functions of the output error ε_k (the squares are used to ensure the positive definiteness of $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$). Thus, the problem reduces to identifying the above functions. To tackle this problem, genetic programming can be employed. The unknown functions $q(\varepsilon_{k-1})$ and $r(\varepsilon_k)$ can be expressed as a tree. Thus, in the case of $q(\cdot)$ and $r(\cdot)$, the terminal sets are $\boldsymbol{T} = \{\boldsymbol{\varepsilon}_{k-1}\}$ and $\boldsymbol{T} = \{\boldsymbol{\varepsilon}_k\}$, respectively. In both cases, the function set can be defined as $\boldsymbol{F} = \{+, *, /, \xi_1(\cdot), \ldots, \xi_l(\cdot)\}$,

where $\xi_k(\cdot)$ is a nonlinear univariate function and, consequently the number of populations is $n_p = 2$. Since the terminal and function sets are given, the GP approach can be easily adapted for the identification purpose of $q(\cdot)$ and $r(\cdot)$. First, let us define the performance index including a necessary ingredient of the $\boldsymbol{Q}_{k-1}$ and $\boldsymbol{R}_k$ selection process.

Since the instrumental matrices should be chosen in order to maximize the convergence rate, we have

$$(\boldsymbol{Q}_{k-1}, \boldsymbol{R}_k) = \arg \max_{q(\boldsymbol{\varepsilon}_{k-1}), r(\boldsymbol{\varepsilon}_k)} j_{obs,1}\big(q(\boldsymbol{\varepsilon}_{k-1}), r(\boldsymbol{\varepsilon}_k)\big). \tag{9.19}$$

On the other hand, owing to FDI requirements, it is clear that the output error should be closed to zero in the fault free mode. In this case, one can define another performance index:

$$(\boldsymbol{Q}_{k-1}, \boldsymbol{R}_k) = \arg \min_{q(\varepsilon_{k-1}), r(\varepsilon_k)} j_{obs,2}\big(q(\boldsymbol{\varepsilon}_{k-1}), r(\boldsymbol{\varepsilon}_k)\big), \tag{9.20}$$

where

$$j_{obs,2}\big(q(\boldsymbol{\varepsilon}_{k-1}), r(\boldsymbol{\varepsilon}_k)\big) = \sum_{k=0}^{n_t-1} \boldsymbol{\varepsilon}_k^T \boldsymbol{\varepsilon}_k. \tag{9.21}$$

Therefore, in order to couple (9.19) and (9.20), the following identification criterion is employed:

$$(\boldsymbol{Q}_{k-1}, \boldsymbol{R}_k) = \arg \min_{q(\varepsilon_{k-1}), r(\varepsilon_k)} \frac{j_{obs,1}(q(\boldsymbol{\varepsilon}_{k-1}), r(\boldsymbol{\varepsilon}_k))}{j_{obs,2}(q(\boldsymbol{\varepsilon}_{k-1}), r(\boldsymbol{\varepsilon}_k))} \tag{9.22}$$

Since the identification criterion is established, it is straightforward to use the GP algorithm. The numerical example considered here is a fifth-order two-phase nonlinear model of an induction motor. Moreover, the following three cases concerning the selection of $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$ were considered:

Case 1: Classical approach (constant values), i.e., $\mathbf{Q}_{k-1} = 0.1$, $\mathbf{R}_k = 0.1$.

Case 2: Selection supported by an analytical consideration:

$$\mathbf{Q}_{k-1} = 10^3 \boldsymbol{\varepsilon}_{k-1}^T \boldsymbol{\varepsilon}_{k-1} \mathbf{I} + 0.01\mathbf{I}, \quad \mathbf{R}_k = 10 \boldsymbol{\varepsilon}_k^T \boldsymbol{\varepsilon}_k \mathbf{I} + 0.01\mathbf{I}. \tag{9.23}$$

Case 3: GP-based approach.

In order to obtain the matrices $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$ using the GP-based approach (Case 3), a set of $n_t = 300$ input-output measurements was generated. The simulation results (for all the cases) are shown in Fig. 9.6 (Witczak *et al.*, 2002). It can be seen, that the proposed approach is superior to the classical technique of selecting the instrumental matrices $\mathbf{Q}_{k-1}$ and $\mathbf{R}_k$.

9.4. Optimization tasks in neural models design

Artificial neural networks are one of the most frequently used techniques in designing diagnosis systems. They are effective when there is no analytical model of the

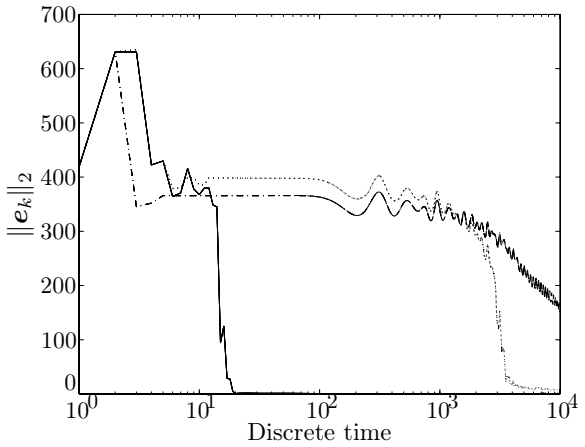


Fig. 9.6. State estimation error norm $\|\boldsymbol{e}_k\|_2$ for Case 1 (dash-dotted line), Case 2 (dotted line) and Case 3 (solid line)

diagnosed system. There are many techniques for constructing static neural models for nonlinear systems (Duch *et al.*, 2000; Korbicz *et al.*, 1994), but their application to dynamic systems modeling requires solving additional problems. Dynamic neural networks are a suitable solution. They can be constructed using feedforward, multilayered networks with additional global (between layers) or local (in the neuron model) feedback connections (Patan and Korbicz, 2000).

Let us assume that $\boldsymbol{y}(k)$ of the form

$$\boldsymbol{y}(k) = f\big(\boldsymbol{u}(k), \boldsymbol{u}(k-1), \ldots, \boldsymbol{u}(k-n), \boldsymbol{y}(k-1), \ldots, \boldsymbol{y}(k-n')\big) \tag{9.24}$$

is a response of a dynamic nonlinear system $f(\cdot)$ to an input signal $\boldsymbol{u}(k)$, and $k \in K$ is the discrete time. Let $\Omega = \{\boldsymbol{u} : K \to \mathbb{R}^n\}$ be the family of all possible maps (infinitely many) from the discrete time domain K to the input signals space $\mathbb{R}^n$. Our goal is to construct a neural model (with the architecture A and the set of free parameters $\boldsymbol{v}$) in the form

$$\begin{aligned}\boldsymbol{y}_{A,\boldsymbol{v}}(k) = f_{A,\boldsymbol{v}}\big(&\boldsymbol{u}(k), \boldsymbol{u}(k-1), \ldots, \boldsymbol{u}(k-n_A), \boldsymbol{y}_{A,\boldsymbol{v}}(k), \boldsymbol{y}_{A,\boldsymbol{v}}(k-1),\\ &\ldots, \boldsymbol{y}_{A,\boldsymbol{v}}(k-n'_A)\big).\end{aligned} \tag{9.25}$$

On the basis of the system description (9.24), the problem of designing the neural model is connected with the minimization of some cost function $\sup_{\boldsymbol{u}(k)\in\Omega} J_T\big(\boldsymbol{y}_{A,\boldsymbol{v}}(k), \boldsymbol{y}(k) \mid k \in K\big)$. Thus, the following pair is searched for:

$$(A^{\mathrm{opt}}, \boldsymbol{v}^{\mathrm{opt}}) = \arg\min\Big(\sup_{\boldsymbol{u}(k)\in\Omega} J_T\big(\boldsymbol{y}_{A,\boldsymbol{v}}(k), \boldsymbol{y}(k) \mid k \in K\big)\Big). \tag{9.26}$$

Practically, the solution $(A^{\mathrm{opt}}, \boldsymbol{v}^{\mathrm{opt}})$ of the problem (9.26) cannot be achieved because of the infinite cardinality of the set Ω. In order to obtain an estimation $(A^*, \boldsymbol{v}^*)$ of the solution, two finite subsets $\Omega_L, \Omega_T \subset \Omega : \Omega_L \cap \Omega_T = \emptyset$ are selected. The set

of training signals Ω_L is used to calculate the best vector $\boldsymbol{v}^*$ for a given model architecture A:

$$\boldsymbol{v}^* = \arg\min_{\boldsymbol{v}\in\mathcal{V}} \Big(\sup_{\boldsymbol{u}(k)\in\Omega_L} J_L\big(\boldsymbol{y}_{A,\boldsymbol{v}}(k), \boldsymbol{y}(k) \mid k \in K\big)\Big), \tag{9.27}$$

where $\mathcal{V}$ is a space of network parameters. Generally, the cost functions for the learning process $J_L\big(\boldsymbol{y}_{A,\boldsymbol{v}}(k), \boldsymbol{y}(k) \mid k \in K\big)$ and the testing process $J_T\big(\boldsymbol{y}_{A,\boldsymbol{v}}(k), \boldsymbol{y}(k) \mid k \in K\big)$ can have different definitions. The set of testing signals Ω_T is used to select a network architecture from all possible architectures $\mathcal{A} = \{A\}$, in order to minimize the following criterion:

$$A^* = \arg\min_{A\in\mathcal{A}} \Big(\sup_{\boldsymbol{u}(k)\in\Omega_T} J_T\big(\boldsymbol{y}_{A,\boldsymbol{v}^*}(k), \boldsymbol{y}(k) \mid k \in K\big)\Big). \tag{9.28}$$

Of course, the solution of both problems (9.27) and (9.28) does not have to be unambiguous. If there are several network architectures which satisfy the assigned criterion, then the model with the minimal number of free parameters is chosen as the solution.

Searching for solutions of the tasks (9.27) and (9.28) is not trivial. The network architecture and the training process strongly influence modeling quality. The main problem is connected with the relation between the learning and generalization abilities, and the finite cardinality of the set of learning signals. If the architecture is too simple, then the obtained network input/output mapping may be unsatisfactory. On the other hand, if the architecture is too complex, then the obtained network mapping strongly depends on the actual set of training signals.

The application of evolutionary algorithms to the design of neural tools has often been described in the literature of the last decade (c.f. (Obuchowicz, 2000; Rutkowska, 2000)). Generally, evolutionary algorithms as global optimization methods can be applied to the following tasks connected with the construction of neural models:

- choosing the set $\boldsymbol{v}$ of free parameters of the network with a given architecture (the training process);
- searching for an optimal neural model architecture; the process of training the tested architectures is performed using other known methods, e.g., the back-propagation algorithm;
- collecting optimal training set for an ANN.

9.4.1. Optimization aspects of collecting the training set for an ANN

During the last several decades many effective algorithms have been proposed to address the problem of estimating the network parameters $\boldsymbol{\theta}^*$ on the basis of the so-called training set (Gupta *et al.*, 2003). However, there is still rather little knowledge regarding collecting the learning set itself. The question how to choose the learning examples arises especially in modeling industrial systems, when the process of collecting measurements is very often costly or/and time consuming. Due to the fact that not every input $\boldsymbol{x} \in \mathbb{R}^{n_u}$ is equally important for the estimation of $\boldsymbol{\theta}$, the problem boils down to the choice of the points from the input space that provide the greatest amount of information about the parameters vectors. At this stage, some concepts

of experimental design theory (Atkinson and Donev, 1992; Uciński, 2004; Witczak, 2006) have proved to be very useful.

Bearing in mind that the primary purpose of the modeling procedure is to obtain a neural model which will be able to imitate the output of the true system as precisely as possible, the so-called G-optimality criterion constitutes the most reasonable choice (Uciński, 2004). Namely, the G-optimality criterion minimizes the variance of the estimated model's output $\Phi(\xi)$ of the following form:

$$\Phi(\xi) = \max_{\boldsymbol{x}\in\mathbf{X}} \boldsymbol{V}\,\boldsymbol{P}^{-1}(\xi)\,\boldsymbol{V}^T, \tag{9.29}$$

where $\boldsymbol{P}(\xi)$ stands for the Fisher Information Matrix (FIM), which in the case of the neural network is given by

$$\boldsymbol{P}(\xi) = \sum_{k=1}^{n_e} \mu_k \Big[\frac{\partial f(\boldsymbol{x}_k,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}\Big] \Big[\frac{\partial f(\boldsymbol{x}_k,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\Big|_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}\Big]^T. \tag{9.30}$$

The experimental design ξ in the above equations stands for a set of training patterns that can be formally defined as

$$\xi = \left\{ \begin{array}{cccc} \boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_{n_e} \\ \mu_1 & \mu_2 & \dots & \mu_{n_e} \end{array} \right\}, \tag{9.31}$$

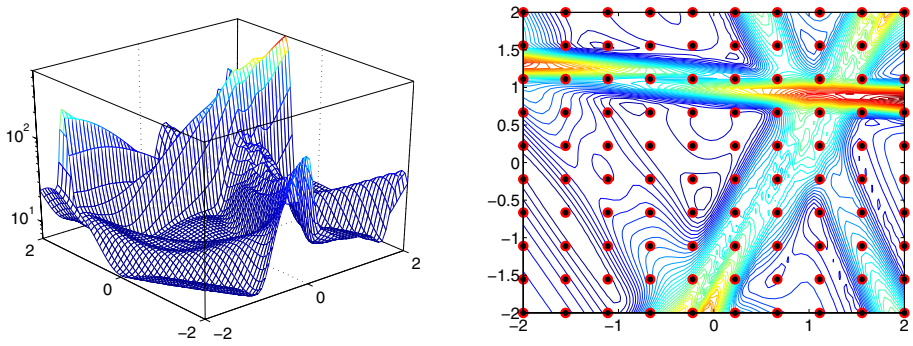
where $\boldsymbol{x}_k$s are said to be the *support points*, and $\mu_1,\dots,\mu_{n_e},\mu_k \in [0,1]$ are called their weights, which satisfy $\sum_{k=1}^{n_e}\mu_k = 1$. The complete algorithm of collecting the G-optimal learning sequence (Witczak, 2006) for the neural model is not free from a severe disadvantage. Namely, its effectiveness to a large extent depends on correctness in choosing the most informative input point $\boldsymbol{x}_k$, which can be found by solving the following optimization problem:

$$\boldsymbol{x}_k = \arg\max_{\boldsymbol{x}\in\mathbf{X}} \boldsymbol{V}^T\,\boldsymbol{P}^{-1}(\xi)\,\boldsymbol{V}. \tag{9.32}$$

The problem (9.32) turns out to be not a trivial one and, due to its multi-modality (see Fig. 9.7), it involves a global optimization technique.

9.4.2. Evolutionary learning of ANNs

Training processes in most neural network implementations are based on the gradient descent method. These algorithms belong to the class of local optimization methods. The advantage of evolutionary training over the back-propagation technique has been revealed by simulation results presented in many works (cf. (Kwaśnicka, 1999)). There are many programs which use evolutionary methods in the neural network training process. For example, the *FlexTool* toolbox is based on genetic algorithms with binary-coding chromosomes, which contain information about all weights of network connections. The advantage of the *Evolver* program is the floating-point representation of an individual. This representation is natural for optimization tasks in a real domain (Rutkowska, 2000).



(a) (b)

Fig. 9.7. (a) – Exemplary variance function for the network with 4 neurons and (b) – its contour along with evenly distributed support points

It is especially in the case of training dynamic neural networks, for which gradient-decent-based methods are limited to a narrow class of networks, that evolutionary algorithms are very attractive. One of the most interesting solutions of dynamic system modeling is the application of a neural network based on dynamic neural models. It is a multilayered feedforward network of processing units, which contain an additional module: the Infinite Impulse Response (IIR) filter. This filter is located between the adder and activation modules. The multilayered architecture of the network allows us to construct the Extended Dynamic Back-Propagation (EDBP) algorithm (Patan and Korbicz, 2000) for the training process of both synaptic connection weights and parameters of IIR filters. The training process of the dynamic neural network seems to be related to the optimization problem with a very rich topology of the network square-error function. The EDBP algorithm usually finds an unsatisfactory local optimum. The effectiveness of the multi-start method is very low, too. Patan and Jesionka (1999) used the genetic algorithm to solve this problem.

The common characteristic of the well-known evolutionary techniques used in the neural network training process is the off-line mode of their processing, i.e., the fitness function is calculated using errors for all training patterns. This fact and the evaluation process of all individuals of the population in each iteration result in a high numerical complexity of the algorithm. The Evolutionary Search with Soft Selection and Forced Direction of Mutation (ESSS-FDM) algorithm (Obuchowicz and Patan, 1997) is applied to on-line dynamic neural network training (Obuchowicz, 1999a).

9.4.3. Optimization of the ANN architecture

From among all known EAs, genetic algorithms seem to be the most natural tool for searching a discrete space of ANN architectures. This fact follows from the classical structure of a chromosome – a string of elements from a discrete set, e.g., a binary set.

The most popular representation of the ANN architecture is a binary string (Obuchowicz and Politowicz, 1997; Obuchowicz,2000). At first, an initial architecture $A_{\max}$

must be chosen. This architecture must be sufficient to realize a desired input-output relation. $A_{\max}$ defines the upper limit of the complexity of the searched architectures. Next, all units of $A_{\max}$ have to be numbered from 1 to N. In this way, the searching space of ANN architectures is limited to the class of all digraphs of N nodes. Any architecture A (a graph) of this class is represented by its connection matrix $\boldsymbol{V}$ of elements equal to 0 or 1. If $V_{ij} = 1$, then there exists a sinaptic connection from the i-th unit to the j-th one, $V_{ij} = 0$ otherwise. A chromosome is constructed by rewriting the matrix $\boldsymbol{V}$ row by row into a bit string of length N^2. Using such a representation of the ANN architecture, a standard GA algorithm can be used. It is easy to see that the above representation can describe an ANN of any architecture: feedforward networks as well as recurrent ones. If the class of the analyzed networks is limited to Multi Layer Perceptron (MLP), then the matrix $\boldsymbol{V}$ contains many elements equal to 0 and cannot be changed during the searching process. Such a limitation complicates genetic operations and requires a lot of memory space in the computer. Thus, passing over these elements in the representation is sensible.

Usually, an ANN has from hundreds to thousands synaptic connections in practical applications, and the binary code representing such an ANN architecture is very long. As a result, standard genetic operations are not effective. The convergence of the genetic process deteriorates as the complexity of the ANN architecture increases. Thus, a simplification of the network architecture representation is needed. One of the possible solutions is a genetic representation of the ANN architecture in the from of the connection matrix $\boldsymbol{V}$. The crossover operator is defined as the exchange of randomly chosen rows or columns between two matrices. In the case of mutation, each bit is turned with some (very small) probability.

The above methods of the genetic representation of the ANN architecture are called *direct encoding*. This term tells us that each bit represents one synaptic connection in the ANN structure. A disadvantage of these methods is slow convergence of the genetic process, or the lack of convergence in the limit of very large architectures. Furthermore, if the initial architecture $A_{\max}$ is very complex, then the result of such a genetic searching process is not as optimal as could be characterized by some compression level. The measure of the efficiency of the method can be the so-called *compression index* of the form

$$\kappa = \frac{\eta^\star}{\eta_{\max}} \times 100\%, \tag{9.33}$$

where $\eta^\star$ is the number of synaptic connections in the resulting architecture, and $\eta_{\max}$ is the maximal number of connections acceptable in a given architecture representation.

An alternative class of genetic representations of ANN architectures is *indirect encoding* (Obuchowicz, 2000). One of the possible solutions is binary encoding of the parameters of an MLP architecture (the number of hidden layers, the number of hidden neurons in each layer, etc.) and the parameters of the BP algorithm used for training this MLP (the training factor, the momentum factor, the desired accuracy, the maximal number of iterations, etc.). A discrete finite set of values is defined for each parameter, and the cardinality of this set depends on the number of bits assigned to a given parameter. In this case the genetic process searches not only for the optimal architecture but for the optimal training process, too.

Another proposition is graph-based encoding. Let the searching space be limited to architectures which contain at most 2^{h+1} units. Then the connection matrix can be represented by a tree of h levels, and each node of this tree possesses four successors or is a leaf. Each leaf is one of the 16 possible matrices (2×2) of binary elements. Four leaves of a given node of the level $h-1$ define a (4×4) matrix, etc. In this way the root of the tree represents the whole connection matrix. The crossover and mutation operators are defined in the same way as in the GP method (Obuchowicz, 2003). The GP algorithm can also be used to design neural models if the mapping realized by a single output unit is represented by an analytical formula which is searched for by GP.

9.5. Parametric uncertainty of neural networks

In order to deliver a complete description of models, especially if a training set is collected by taking a series of measurements by means of industrial sensors, the description of parametric uncertainties seems to be indispensable (Witczak, 2003). It is noticeable that statistical inference about confidence regions and confidence intervals, mainly due to the nonlinearity of neural networks, is not an easy task. Nevertheless, based on the linear approximation (Bates and Watts, 1980; Prętki and Witczak, 2005a), it can be shown that the confidence interval (at, the $100(1-\alpha)$ level) for the predictive output of a neural network can be approximated as follows (Witczak and Prętki, 2005):

$$y_{m,k}^{L} \leq y_{m,k} \leq y_{m,k}^{U} \tag{9.34}$$

where lower and upper adaptive thresholds are given by (Witczak and Prętki, 2005):

$$y_{m,k}^{L} = f(\boldsymbol{x}_k, \hat{\boldsymbol{\theta}}) - \hat{\sigma}\sqrt{\boldsymbol{V}^T \boldsymbol{P}^{-1}(\xi)\boldsymbol{V}}\sqrt{n_p\, F_{(n_p, n_t - n_p; \alpha)}}, \tag{9.35}$$

$$y_{m,k}^{U} = f(\boldsymbol{x}_k, \hat{\boldsymbol{\theta}}) + \hat{\sigma}\sqrt{\boldsymbol{V}^T \boldsymbol{P}^{-1}(\xi)\boldsymbol{V}}\sqrt{n_p\, F_{(n_p, n_t - n_p; \alpha)}}, \tag{9.36}$$

where $F_{(n_p, n_t - n_p; \alpha)}$ denotes the upper α quantile for Fisher's distribution with n_p and $n_t - n_p$ degrees of freedom, and $\hat{\sigma}$ is the standard deviation estimate. The above approach is often used in many practical applications of nonlinear regression (Chryssolouris *et al.*, 1996); nevertheless, one should be cautious about the results obtained in this way. In fact, when using the linearization method the user must be sure that a nonlinear model can be accurately approximated by a linear one in certain operating conditions. This, however, is not necessarily fulfilled, and many examples of such models can be found in (Bates and Watts, 1980). In order to check whether the model under consideration can be substituted by a linear one, one can take advantage of curvature measures of nonlinearity (Bates and Watts, 1980), which allow assessing the credibility of the confidence intervals.

9.5.1. Adequacy of the linear approximation

The methodology presented in the previous sections is based on the convenient assumption, that a neural network can be accurately approximated by a linear model

in certain operating conditions. The linear approximation provides inference regions which are easy to calculate and can be applied to any number of parameters. The approach is often used in many practical applications of nonlinear regression (Chryssolouris *et al.*, 1996). In this section, we introduce a measure which indicates the adequacy of the linear approximation in some operating conditions. Based on the curvature measures of nonlinearity (Bates and Watts, 1980), it is possible to assess the credibility of the confidence intervals (9.35), (9.36). The method, first introduced by Wattes and Bates (Bates and Watts, 1980), is based on measuring how strongly a quadratic approximation of the expectation surface is deviated from the linear one – the so-called *tangent plane*. Formally, the approach uses the second-order Taylor expansion of the expectation surface around the current estimate $\hat{\boldsymbol{\theta}}$:

$$\boldsymbol{\eta}(\boldsymbol{\theta},\boldsymbol{\xi}) = \boldsymbol{\eta}(\hat{\boldsymbol{\theta}},\boldsymbol{\xi}) + \boldsymbol{V}_{\boldsymbol{\xi}}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}) + \frac{1}{2}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}})^T \boldsymbol{A}_{\boldsymbol{\xi}}(\boldsymbol{\theta}-\hat{\boldsymbol{\theta}}), \tag{9.37}$$

where $\boldsymbol{\eta}(\cdot,\boldsymbol{\xi}) : \mathbf{P} \to R^{n_t}$ stands for a mapping from the parameters space to the response space, $\boldsymbol{V}_{\boldsymbol{\xi}} \in \mathbf{R}^{n_t \times n_p}$ is the so-called *velocity matrix*, which in the case of the neural model has the following form

$$\boldsymbol{V}_{\boldsymbol{\xi}} = \Big[\frac{\partial f(\boldsymbol{x}_1,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}, \frac{\partial f(\boldsymbol{x}_2,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}, \ldots, \frac{\partial f(\boldsymbol{x}_n,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}}\Big]^T, \tag{9.38}$$

and $\boldsymbol{A}_{\boldsymbol{\xi}} \in \mathbf{R}^{n_t \times n_p \times n_p}$ stands for the so-called *accelerator array* whose *k-th* face is of the form:

$$\{\boldsymbol{A}_{\boldsymbol{\xi}}\}_k = \left(\frac{\partial^2 f(\boldsymbol{x}_k,\boldsymbol{\theta})}{\partial \boldsymbol{\theta}\, \partial \boldsymbol{\theta}^T}\right)_{\boldsymbol{\theta}=\hat{\boldsymbol{\theta}}}. \tag{9.39}$$

Based on the above structures, it is possible to define the so-called relative curvature in an arbitrary direction $\boldsymbol{d}$ (Bates and Watts, 1980):

$$c_d = \frac{\|\boldsymbol{d}^T \boldsymbol{A}_{\boldsymbol{\xi}} \boldsymbol{d}\|}{\|\boldsymbol{V}_{\boldsymbol{\xi}} \boldsymbol{d}\|^2}. \tag{9.40}$$

Multiplying the numerator of (9.40) only by the first n_p faces one obtains the value of the *parameter effect,* c_P. Subsequently, the numerator of (9.40) composed of $n_p + 1, \ldots, n_t$ faces defines the second measure of nonlinearity – *intrinsic curvature,* c_I. In order to assess the adequacy of the linear approximation, the following averagedvalues of the above measures are recommended (Bates and Watts, 1980):

Root-mean-square curvature of the parameter effect:

$$\bar{c}_P = \frac{\Gamma(n_p/2)}{2\pi^{n_p/2}} \sum_{k=1}^{n_p} \int\limits_{S:\boldsymbol{d}^T\boldsymbol{d}=1} \frac{\|\boldsymbol{d}^T\{\boldsymbol{A}_{\boldsymbol{\xi}}\}_k \boldsymbol{d}\|}{\|\boldsymbol{V}_{\boldsymbol{\xi}}\boldsymbol{d}\|^2} \hat{\sigma}\sqrt{n_p}\, \mathrm{d}S; \tag{9.41}$$

Root-mean-square of the intrinsic curvature:

$$\bar{c}_I = \frac{\Gamma(n_p/2)}{2\pi^{n_p/2}} \sum_{k=n_p+1}^{n_t} \int\limits_{S:\boldsymbol{d}^T\boldsymbol{d}=1} \frac{\|\boldsymbol{d}^T\{\boldsymbol{A}_{\boldsymbol{\xi}}\}_k \boldsymbol{d}\|}{\|\boldsymbol{V}_{\boldsymbol{\xi}}\boldsymbol{d}\|^2} \hat{\sigma}\sqrt{n_p}\, \mathrm{d}S, \tag{9.42}$$

where $\mathrm{d}S$ stands for an element of the surface area on the unit sphere, and $\hat{\sigma}$ is the standard deviation estimate.

When the measures exceed some fixed threshold, i.e., $\bar{c}_I\sqrt{\chi^2_{n_t,\alpha}} < 0.3$ and $\bar{c}_P\sqrt{\chi^2_{n_t,\alpha}} < 0.3$, ($\chi^2_{n_t,\alpha}$ – Chi-square distribution quantile at the level α), the user has to be aware that confidence regions provided by the linear approximation can be extremely misleading (where 0.3 is an empirical selected threshold (Bates and Watts, 1980)).

In order to show how the above technique can be applied in practice, let us consider the following regression model:

$$y_{s,k} = \tanh(p_1 u_k + p_2) + \epsilon_k, \tag{9.43}$$

where ϵ_k stands for a sequence of i.i.d. random variables with the normal distribution $N(0, 0.05)$, and $\boldsymbol{p} = [p_1, p_2]^T = [-0.9, 1.1]^T$. A careful reader can notice that the model (9.43) represents a neuron with a hyperbolic tangent activation function – a unit from which multilayered feedforward neural networks are built. Let us introduce the experimental condition $\boldsymbol{\xi} = [1.7, 0, -0.7]^T$ for which the vector of the system outputs (9.43) $\boldsymbol{Y}_s = [-0.41, 0.8, 0.98]^T$ was obtained. Using the standard Levenberg-Marquard algorithm, a parameter estimate was obtained $\hat{\boldsymbol{p}} = [-0.95, 1.16]^T$ with the following velocity matrix:

$$\boldsymbol{V} = \begin{bmatrix} \boldsymbol{v_1} & \boldsymbol{v_2} \end{bmatrix} = \begin{bmatrix} 0.0396 & 0.0233 \\ 0 & 0.4523 \\ -0.6460 & 0.9229 \end{bmatrix},$$

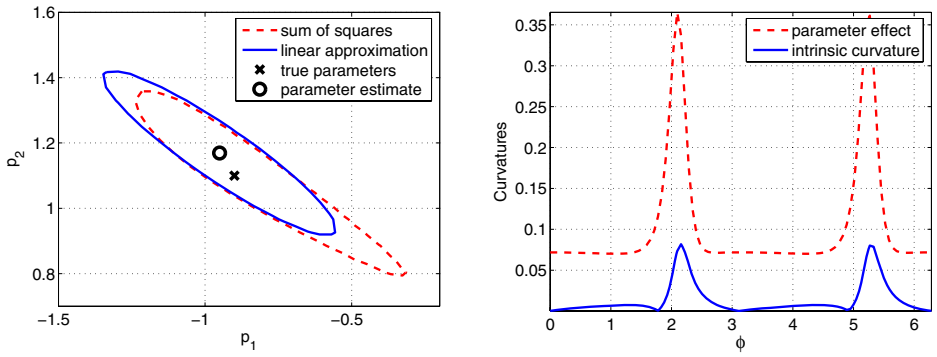
and accelerator vectors

$$\boldsymbol{A} = \begin{bmatrix} \boldsymbol{a_1} & \boldsymbol{a_2} & \boldsymbol{a_3} \end{bmatrix} = \begin{bmatrix} 0.2425 & -0.3629 & 0.5088 \\ -0.0258 & -0.0098 & -0.6727 \\ 0.1460 & 0.0554 & -0.0401 \end{bmatrix}.$$

The values of the both curvatures $c_P^{(1)}, c_I^{(1)}$ are presented in Fig. 9.8 (after transformation to the spherical coordinates $p_1 = \cos(\phi), p_2 = \sin(\phi)$). For this example, the averaged measures of the curvatures (9.41),(9.42) are equal to 0.96 and 0.47 respectively. Since both values exceed an acceptable threshold, we may expect that the confidence ellipsoid determined by the linearization method will be significantly deformed, which may lead to considerably misleading conclusions. In Fig. 9.8 one may compare the contours of the 95% confidence region: the first one obtained with the linear approximation and the exact one related to the sum of squares (9.46).

9.5.2. Evolutionary bands for the expected response

In order to avoid the linearization approach or other extremely computational intensive methods used in regression analysis (Bates and Watts, 1980; Chryssolouris *et al.*, 1996; Donaldson and Schnabel, 1987), let us present an alternative way of



(a) (b)

Fig. 9.8. (a) – Sum of squares contours (dashed line) of confidence regions and the elliptical contour provided by the linear approximation (solid line), (b) – Curvature measures: intrinsic (solid line) and parameter effect (dashed line) of nonlinearity for particular angles in the parameters space

handling confidence intervals. The lower $y^L_{m,k}$ and upper $y^U_{m,k}$ adaptive thresholds for the predicted response of the nonlinear model can be formulated as the following optimization problems (Prętki and Obuchowicz, 2006):

$$y^L_{m,k} = \arg\min_{\boldsymbol{\theta}\in\Theta} f(\boldsymbol{x}_k, \boldsymbol{\theta}), \tag{9.44}$$

$$y^U_{m,k} = \arg\max_{\boldsymbol{\theta}\in\Theta} f(\boldsymbol{x}_k, \boldsymbol{\theta}). \tag{9.45}$$

The feasible parameters set, which for linear models is an ellipsoid (Bates and Watts, 1980), in the case of nonlinear models has much more complex shape, and is defined in the following manner (Bates and Watts, 1980):

$$\Theta = \left\{ \boldsymbol{\theta} \in \mathbb{R}^{n_p} \middle| S(\boldsymbol{\theta}) - S(\hat{\boldsymbol{\theta}}) \leq s^2\, n_p\, F_{(n_p, n_t - n_p; \alpha)} \right\}, \tag{9.46}$$

where

$$S(\boldsymbol{\theta}) = \sum_{k=1}^{n_t} \Big(y_{s,k} - f(\boldsymbol{x}_k, \boldsymbol{\theta}) \Big)^2 \text{ and } s^2 = \frac{S(\hat{\boldsymbol{\theta}})}{n_t - n_p}. \tag{9.47}$$

The nonlinear function $f(\cdot)$ may possess more than one local optimum, thus it seems reasonable to use some of global optimization techniques to solve (9.44) and (9.45). It is worth noticing that the optimization algorithm must be supplied with an additional procedure which allows to deal with nonlinear constraints. Since in this case the matter of obeying the constraints plays a major role, it is not recommend to use such approaches as penalty functions (Michalewicz, 1996). Instead, in Tab.9.1 a very simple and efficient method of projecting a solution into a border of Θ is outlined.

In order to show that the proposed approach provides more accurate adaptive thresholds for residual signals, let us introduce the following experiment: We consider

Table 9.1. Procedure of projecting the solution $\boldsymbol{\theta}'$ into a border of the feasible set Θ

Input data

Eps – absolute accuracy of localization of the border;

$\boldsymbol{\theta}, \boldsymbol{\theta}'$ – parent and individual after mutation, respectively;

$R = s^2\, n_p\, F_{(n_p, n_t - n_p; \alpha)}$ – radius of the feasible set;

Output data

$\boldsymbol{\theta}^*$ – individual projected onto a border;

Algorithm

$\boldsymbol{\theta}^* \leftarrow \boldsymbol{\theta}$

$h \leftarrow ||\boldsymbol{\theta}^* - \boldsymbol{\theta}'||_2$

Repeat

  If $S(\frac{1}{2}(\boldsymbol{\theta}^* + \boldsymbol{\theta}')) - S(\boldsymbol{\theta}) > R$ then

    $\boldsymbol{\theta}' \leftarrow \frac{1}{2}(\boldsymbol{\theta}^* + \boldsymbol{\theta}')$

  else

    $\boldsymbol{\theta}^* \leftarrow \frac{1}{2}(\boldsymbol{\theta}^* + \boldsymbol{\theta}')$

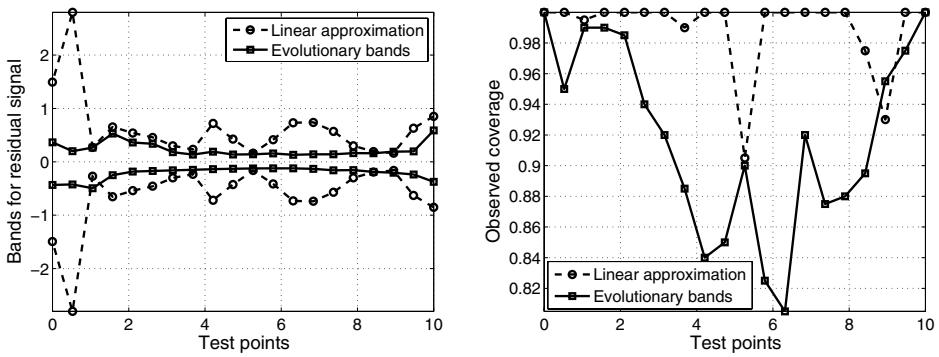
  end if

  $h \leftarrow h/2$

Until $(h > Eps)$

one input, one output neural network consisting of three hidden neurons with a hyperbolic tangent sigmoid transfer function and one linear output unit. Thus, the total number of parameters, including all weights between the connected neurons and their biases, is equal to $n_p = 14$. It must be stressed that in the experiment all parameters $\boldsymbol{\theta}^*$ of the network were chosen randomly. Such a neural model served as a deterministic part of the nonlinear regression problem. The training set $\{x_k, y_k\}_{k=1}^{18}$ was collected by equally dividing an interval $[0, 10]$ – $\{x_k\}$, and disturbing the output of the network by adding the Normal pseudo-random numbers $\xi_k \sim N(0, 0.1^2)$, e.g., $y_k = f(x_k, \boldsymbol{\theta}^*) + \xi_k$. Next, such training set was used to obtain estimates $\hat{\boldsymbol{\theta}}$ for the neuron network with the same structure (for this purpose the well-known Levenberg-Marquardt method was used). For the test purposes we chose twenty points to uniformly cover the interval $[0, 10]$. In Fig. 9.9 (a) comparison of the adaptive thresholds (at the $1 - \alpha = 0.9$ confidence level) for residual signals obtained with the linearization method and evolutionary computation (9.44), (9.45) is presented. In order to check the adequacy of the bands obtained for both methods, the set of two hundred systems responses for each tested point was generated. The percentage of the residuals that lay inside adaptive thresholds can be observed in Fig. 9.9(b).

In the case of nonlinear models, the linearization method may overestimate the value of the bands for expected response. In Fig. 9.9(b) it can be observed that almost for all tested points from the input domain, bands obtained via the linearization



(a) (b)

Fig. 9.9. (a) – $1 - \alpha = 0.9$ bands for the expected responses of the neural model, and (b) their observed coverage

model cover nearly 100 % of the system responses. This may cause serious problems in many engineering applications, i.e., the model based residual generator system of fault detection (Korbicz *et al.*, 2004; Prętki and Witczak, 2005a; 2005b), where sensitivity to faulty conditions plays a key role.

9.6. Neuro-fuzzy model structure and parameters tuning

9.6.1. Number of partition definitions for network inputs

Neuro-fuzzy network structure optimization demands the number of partition definitions for all inputs. In this case the optimization is performed in the discrete space and, therefore, when an optimum number of partitions is searched for we can use genetic algorithms or heuristic search methods. Piegat (2003) applied the A* algorithm, where each node of the search tree represents some neuro-fuzzy model (Fig. 9.10). For heuristic search methods, the following form of the fitness function is proposed:

$$f_{dop} = \frac{1}{N}\sum_{i=1}^{n} e_i^2 + \sqrt{\frac{1}{N}\sum_{i=1}^{n}(x_i - \overline{x})^2} + w\, l_p, \tag{9.48}$$

where $e_i = x_i - y_i$ stands for the modeling error, x_i denotes the model output, y_i signifies the reference signal, $\overline{x}$ is the mean value of the model response, w denotes a weight selected in the interval (0, 1), and l_p is the number of partitions.

This problem is described in detail in (Andrzejewski and Pieczyński, 2005).

9.6.2. Shape of the fuzzy set membership function

The shape of membership functions plays an important role in neuro-fuzzy modeling. In the literature two kinds of membership curves can be found. The first group is characterized by a finite carrier like the triangular or the trapezoidal curve. The

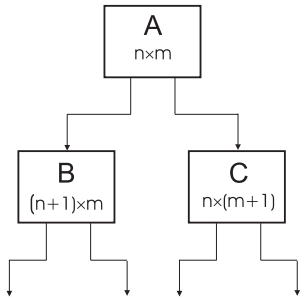


Fig. 9.10. Solutions space searching diagram for the heuristic method for two inputs (notation: A, B, C – nodes of the searched space; n – number of partitions for the input x_1; m – number of partitions for the input x_2)

next one is a group with an infinite carrier like a generalized Gaussian function. The following generalized Gaussian function, which is a compromise between both types of shapes, is proposed:

$$\mu_F(x) = \exp\left(-\left(\frac{x-b}{a}\right)^{\beta}\right), \tag{9.49}$$

where b is a modal value, a is a range factor and β is a characteristic factor.

The factor β is used to change the Gaussian function's shape (Fig. 9.11). For $\beta = 2$, the classical shape of the Gaussian function is obtained. Closely triangular or trapezoidal shapes are obtained for suitable factors, $\beta_1 = 0.5$ or $\beta_2 = 5$.

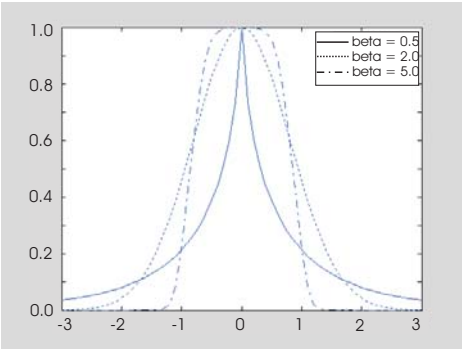


Fig. 9.11. Influence of the β factor on the general Gaussian function shape

The main advantage of the the generalized Gaussian function is the possibility of adapting the function profiles by changing the parameter β.

The optimization process for the neuro-fuzzy model, including the membership function with a finite carrier, is based on the GA, whose chromosome representation

is the following:

$$X = \underbrace{\underbrace{x_1x_2}_{par.\ no\,1}\ \underbrace{x_3x_4}_{par.\ no\,2}\ \underbrace{x_5x_6}_{par.\ no\,3}}_{function\ code\ for\ input\ 1}\ \underbrace{\underbrace{x_7x_8}_{par.\ no\,1}\ \underbrace{x_9x_{10}}_{par.\ no\,2}\ \underbrace{x_{11}x_{12}}_{par.\ no\,3}}_{function\ code\ for\ input\ 2}\ \underbrace{\underbrace{x_{13}x_{14}}_{par.\ no\,1}\ \underbrace{x_{15}x_{16}}_{par.\ no\,2}\ \underbrace{x_{17}x_{18}}_{par.\ no\,3}}_{function\ code\ for\ output}, \quad (9.50)$$

The GA applied uses range reproduction and one-point crossover. On the Table 9.2 shows the code of membership function type. The results of the investigation of the examples are presented in (Pieczyński, 2006).

Table 9.2. Membership function codes

Code	00	01	10	11
Function type	Triangular	Gauss	trapezoidal	sigmoid

The second kind of membership curve is generalized Gauss. The optimization of this curve's parameters concerns also the β parameter. Many approaches to the tuning of the different fuzzy model parameters are based on expert knowledge, gradient methods (Pieczyński, 2002) and evolutionary algorithms. In order to allocate the β value the evolutionary algorithm is used.

The proposed approach is based on the evolutionary search with soft selection algorithm (Obuchowicz, 2003). At the beginning, the population of points is selected from the domain, and next it is iteratively transformed by selection and mutation operations. As a selection operator, the well-known proportional selection (the roulette method) is chosen. The coordinates of, selected parents are mutated by adding normally-distributed random values. In the proposed approach, an individual $\vec{x} \in \mathbb{R}^{n+m}$ contains information of $n + m$ fuzzy sets (n for the input and m for the output) described by two parameters, a and b (9.49). The population consist of 30 individuals, the standard deviation of normally-distributed mutation decreases according to the plan defined earlied. The results of optimization for a example is described in the paper (Pieczyński and Obuchowicz, 2004).

9.6.3. Inference and defuzzification modules

The inference block plays a very important role in the structure of the neuro-fuzzy network. The selection inference system may have essential meaning for the network, because in influences the choice of the rule to be activated. The inference block is connected with the knowledge base. As an inferential system, Mamdani's or Takagi-Sugeno's models are often applied. These models can work on different *t-norm* and *s-norm* operators. In the inference block of the fuzzy model, a few operators are used. The *min* and *max* operators are applied frequently. But in the literature (Piegat, 2003), many operators are known. The discrete optimization method for operators can be exploited. For this task there was used the genetic algorithm. The chromosome consists of 6 genes and is applied to *t-norm* and *s-norm* definition. The chromosome

used given by

$$X = \underbrace{x_1 x_2 x_3}_{t\text{-}norm\ code} \quad \underbrace{x_4 x_5 x_6}_{s\text{-}norm\ code}, \tag{9.51}$$

The last block of the neuro-fuzzy network is the defuzzification block. It is responsible for signal conversion obtained as a result of inference – to form a sharp, accurate signal. From many known ways of sharpener is often accept the singleton method, Centre of Average (CA) or Centre of Sum (CS). In the literature there are known a few different defuzzification methods. Each of them generate differents crisp signal. Therefore the optimization of this parameter is a significant task. The discrete optimization method for the defuzzification task can be exploited again. The chromosome applied in the genetic algorithm consists of 3 genes and it is applied to defuzzification method definition. The chromosome is used given by

$$X = \underbrace{x_1 x_2 x_3}_{defuzzification\ method\ code}, \tag{9.52}$$

These three parameters (the *t-norm* operator, the *s-norm* operator, the defuzzification method) can be defined using the genetic algorithm with the chromosome connecting both chromosomes described earlier. The results of the these parameters' optimization were described in the paper (Pieczyński, 2001; 2006).

9.6.4. Neuro-Fuzzy structure optimization

The procedure of neuro-fuzzy network design consists of the structure selection stage and the parameter estimation stage (Rutkowska, 2002). The pessimistic scenario assumes the construction of the neuro-fuzzy network only on the basis of the available measurements. The main problem is to obtain the required accuracy and transparency of the rule base in such a situation. A lot of different methods have already been developed both for structure selection and parameter estimation of the neuro-fuzzy network, but there is a demand for better, more effective algorithms, and active research is still conducted in this area (Korbicz and Kowal, 2001; Rutkowska, 2000).

Takagi-Sugeno neuro-fuzzy networks can be viewed as multi-model systems which consist of some rules, and each rule defines a single model as the consequent of the rule (Babuška, 1998; Kowal and Korbicz, 2002a; 2002b; 2003). The global neuro-fuzzy system is a set of N_r partial models, where N_r determines the number of fuzzy rules. The output of the global system is calculated as a mixture of partial model outputs. The rule fulfillment is determined by fuzzy sets. In order to ensure the desired accuracy of the neuro-fuzzy system, the membership functions of fuzzy sets must be placed properly in the input space, the number of rules must be appropriate and the parameters of partial models must be chosen to minimize the defined error.

Two main strategies for placing fuzzy sets in the input space can be distinguished: the first one proposes to minimize the output error of the global model (Leith and Leithead, 1999), and the other one is based on partial models that model the local behavior of the system (Abonyi *et al.*, 2002). A typical property of the first approach is to arrange fuzzy sets in the input space in such a way that all partial models are active in the whole domain of input variables. In this case, the accuracy of the global model is guaranteed by the proper mixture of partial model outputs. The alternative approach

does not examine the global accuracy of the model but concentrates on partial models, which should tune in to the local behavior of the system. The problem of rule base declaration reduces to the determination of the number of rules required for a precise description of the problem to be solved.

The simplest method used to determine the number of rules is based on generating a uniformly distributed grid of rules in the input space. The usage of such an approach is limited only to simple systems with a small number of inputs. The approach does not work well for more complicated systems because it generates a combinatorial explosion of rules, which make this method useless. Fuzzy clustering algorithms are another technique which is often used for fuzzy rule generation (Babuška, 1998; Chen *et al.*, 1998; Kowal *et al.*, 2002; Mendes *et al.*, 2002). The idea of this approach is to find natural groups of data in order to apply to each group one fuzzy rule (Babuška, 1998). Generally, clustering algorithms can be divided into two main classes, i.e., hard clustering and fuzzy clustering. It seems to be natural to use fuzzy clustering algorithms in the case of neuro-fuzzy networks. The task of fuzzy clustering is usually reduced to finding the local minimum of the nonlinear cost function, defined by the following expression:

$$J(\boldsymbol{X};\boldsymbol{U},\boldsymbol{V}) = \sum_{i=1}^{c}\sum_{k=1}^{N} \mu_{ik}^{m} D_{ik}^{2}, \tag{9.53}$$

where the matrix $\boldsymbol{U}$ contains the membership degrees of data points from the matrix to the defined clusters $\boldsymbol{X}$, $\boldsymbol{V} = [\boldsymbol{v_1}, \boldsymbol{v_2}, \ldots, \boldsymbol{v_c}], \boldsymbol{v_i} \in R_n$ is a matrix which defines the centers of the clusters, D_{ik} is a metric used to determine the distance between the data points and the cluster centers:

$$D_{ik}^{2} = \|\boldsymbol{x_k} - \boldsymbol{v_i}\|^2 = (\boldsymbol{x_k} - \boldsymbol{v_i})^T \boldsymbol{A} (\boldsymbol{x_k} - \boldsymbol{v_i}), \tag{9.54}$$

and the parameter m takes values from 1 to ∞ and determines the degree of fuzziness of the clusters. The cost function (9.53) can be viewed as a total variance of the data $\boldsymbol{x_k}$ with respect to the cluster centers $\boldsymbol{v_i}$. The matrix A which occurs in the expression (9.54) is used to tune the shape and orientation of the clusters in the space. The fuzzy clustering algorithm which uses such a norm to calculate the distance between data points and cluster centers is called *Fuzzy C-Mean* (FCM) (Bezdek, 1981). However, the number of the found clusters strongly depends on the values of coefficients, which must be defined by the designer at the beginning of the procedure, so the application of the algorithm is difficult. Two clustering algorithms were applied to built the model of the valve which is a part of the industrial installation of the Lublin Sugar Factory (DAMADICS, 2002). The learning procedure of the Takagi-Sugeno neuro-fuzzy network was divided into two phases. In the first step, clustering methods were used to optimize the network structure and prepare the initial values of the parameters. In the second step, the gradient descent method was used to tune all parameters. Two clustering algorithm were used in the first step: the mountain method and the fuzzy C-mean algorithm (Kowal *et al.*, 2002; Mendes *et al.*, 2002). Sample results are shown in Fig. 9.12.

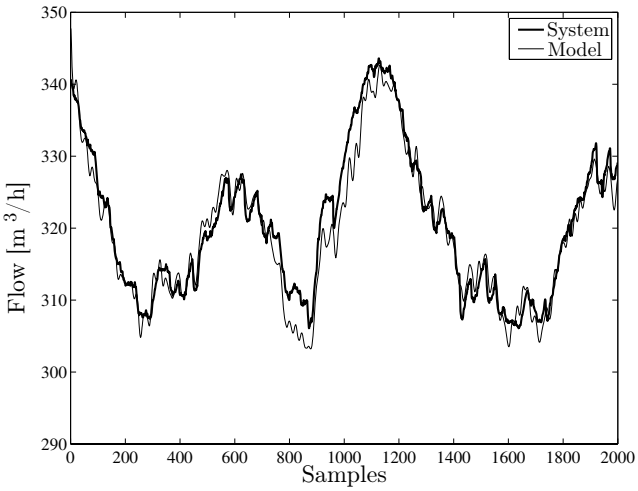


Fig. 9.12. Performance of the TSK neuro-fuzzy model for the valve V_{51} (SSE / $N.^o$ of samples = 6.774)

9.6.5. Neuro-fuzzy parameters tuning

The application of neuro-fuzzy networks in diagnostic areas creates a demand for suitable design procedures which would take into account the specificity of the fault diagnosis task. An important problem from the diagnostic point of view is residual confidence interval minimization because it makes it possible to detect a fault appropriately early. It has to be stressed that the value of the confidence interval for residuals depends directly on the uncertainty of the model which is used to generate the residuals. If the confidence interval is not consistent with model uncertainty, the fault detection system can trigger off a lot of false alarms. It is obvious in such a situation that model uncertainty has to be considered in fault detection threshold calculations (Chen and Patton, 1999; Mrugalski, 2003; Witczak, 2003). It is also important to minimize model uncertainty in order to obtain a reliable fault detection system that would be able to detect a fault fast and at an early stage, so special procedures for neuro-fuzzy model design must be developed.

To overcome the problem, an alternative approach in the form of the Bounded Error Approach (BEA) method can be applied to tune the parameters of the Takagi-Sugeno neuro-fuzzy network and to calculate the admissible set of parameters and the confidence interval for the network output. The method requires only the information about the range of the disturbances which corrupt measurements. The application of the BEA algorithm for computing the confidence interval of the Takagi-Sugeno fuzzy model output requires to establish some assumptions in order to view the model in the form of an LP system (Kowal, 2005; Kowal and Korbicz, 2005a; 2005c). The main assumption based on the fact that the parameters of the membership functions of the fuzzy sets are known. Appropriate selection of the values of these parameters has an essential influence on the uncertainty of the whole fuzzy model. Wrong values of

these parameters can significantly increase model uncertainty, thus the model can be unsuitable for diagnostic tasks.

In the proposed approach the clustering algorithm is used to determine the ellipsoid clusters in the input-output space in order to generate for each found cluster one local linear submodel and to determine the parameters of the fuzzy partitions by cluster projection (Babuška, 1998). Another approach is based on the detection of approximately linear dependencies in the data space using a modified BEA (Kowal, 2005; Kowal and Korbicz, 2005a; 2005b). The algorithm consists in the generation of a single rule for each found linear dependency and allows the parameters of fuzzy partitions.

In order to present the BEA approach for estimating the parameters of the determining Takagi-Sugeno (T-S) network, let us consider the following T-S neuro-fuzzy model:

$$y(k) = \sum_{i=1}^{n} \phi_i(k) y_i(k), \tag{9.55}$$

where $y_i(k)$ is the output of the i-th rule and

$$\phi_i(k) = \frac{\mu_i(k)}{\sum_{j=1}^{n} \mu_j(k)}. \tag{9.56}$$

The model described by the equation (9.55) can be viewed in the form of an LP system:

$$y = \boldsymbol{x}^T(k)\boldsymbol{p}, \tag{9.57}$$

where

$$\boldsymbol{x}(k) = \begin{bmatrix} \phi_1(k)\boldsymbol{r}_1(k) \\ \phi_2(k)\boldsymbol{r}_2(k) \\ \vdots \\ \phi_n(k)\boldsymbol{r}_n(k) \end{bmatrix}, \quad \boldsymbol{p} = \begin{bmatrix} \boldsymbol{p}_1 \\ \boldsymbol{p}_2 \\ \vdots \\ \boldsymbol{p}_n \end{bmatrix}.$$

if the parameters of the fuzzy sets are treated like constant values. The output error is given by the following formulae:

$$\varepsilon(k) = y'(k) - \boldsymbol{x}^T(k)\boldsymbol{p}, \tag{9.58}$$

where $e(k)$ is the error and $y'(k)$ is the output of the system. The error is bounded by means of the following inequalities:

$$\varepsilon^{\min}(k) \leq \varepsilon(k) \leq \varepsilon^{\max}(k). \tag{9.59}$$

thus the admissible set of parameters for N data points is given by the following expression:

$$\mathbf{P} = \left\{ \boldsymbol{p} \in \mathbb{R}^n \,|\, y'(k) - \varepsilon^{\max} \leq \boldsymbol{x}^T(k)\boldsymbol{p} \leq y'(k) - \varepsilon^{\min}, \;\; k = 1, \ldots, N \right\}. \tag{9.60}$$

Each point inside the set $\mathbf{P}$ defines the vector of model parameters and all sets of parameters determine the group of models consistent with the measurements and

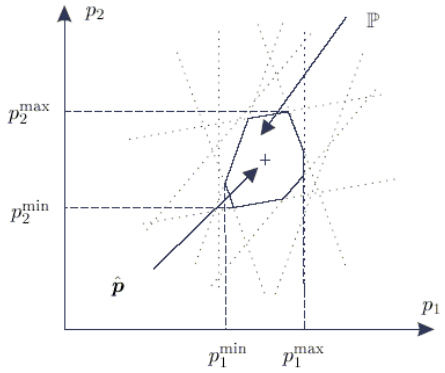


Fig. 9.13. Sample set of parameters $\mathbf{P}$

bounds. This means that, instead of one model, a set of models with different parameters is given and the output signal is represented in the form of an interval which contains all possible model responses. Real applications usually require a single output value, thus one set of parameters must be chosen. The most common approach chooses the geometrical center of the area $\mathbf{P}$ as the set of parameters that is used to calculate the output of the model. This sample procedure is shown in Fig. 9.13. If the maximum and minimum values of the parameters are known,

$$p_i^{\min} = \arg\min_{p\in\mathbf{P}} p_i, \tag{9.61}$$

$$p_i^{\max} = \arg\max_{p\in\mathbf{P}} p_i, \tag{9.62}$$

the estimates of the parameters can be computed using the following formula:

$$p_i = \frac{p_i^{\min} + p_i^{\max}}{2}, \quad i = 1, \ldots, N. \tag{9.63}$$

The minimum and maximum values for the following parameters are determined using the linear programming technique (Milanese *et al.*, 1996). The feasible set of parameters is used also to compute the confidence interval for the output of the system:

$$\boldsymbol{x}^T(k)\boldsymbol{p}^{\min}(k) + \varepsilon^{\min} \le y'(k) \le \boldsymbol{x}^T(k)\boldsymbol{p}^{\max}(k) + \varepsilon^{\max}, \tag{9.64}$$

where

$$\boldsymbol{p}^{\max}(k) = \arg\max_{\boldsymbol{p}\in\mathbf{W}} \boldsymbol{x}^T(k)\boldsymbol{p}, \tag{9.65}$$

$$\boldsymbol{p}^{\min}(k) = \arg\min_{\boldsymbol{p}\in\mathbf{W}} \boldsymbol{x}^T(k)\boldsymbol{p}. \tag{9.66}$$

The confidence interval can be directly applied to calculate the adaptive threshold for the residual signal:

$$e_r(k) = y'(k) - y(k). \tag{9.67}$$

Finally, the adaptive threshold is described by the following inequalities:

$$\boldsymbol{x}^T(k)\boldsymbol{p}^{\min}(k) + \varepsilon^{\min}(k) - y(k) \leq e_r(k) \leq \boldsymbol{x}^T(k)\boldsymbol{p}^{\max}(k) + \varepsilon^{\max}(k) - y(k). \quad (9.68)$$

Unfortunately, the computations required to determine all vertices $\mathbf{W}$ of the convex polyhedron $\mathbf{P}$ are so time and memory consuming that it is hard to employ the classical BEA algorithm for complicated models. In this case the methods that approximate the actual set $\mathbf{P}$ by the area which has a simplified shape should be employed (Milanese *et al.*, 1996).

The method that approximates the set $\mathbf{P}$ by the Outer Bounding Ellipsoid (OBE) has been applied to fault detection in a DC engine (Kowal, 2005; Kowal and Korbicz, 2005b; 2006b). Sample experimental results obtained for the faulty scenario are shown in Figs. 9.14 and 9.15. The presented approach does not take into account the fact that not only the output variable $y(k)$ is uncertain but also all input variables $\boldsymbol{x}(k)$ can be uncertain. Such a situation is common due to the fact that input variables are usually measured so they can be known with a defined accuracy. If this fact is not considered, the threshold computed for the output variable does not reflect real model uncertainty so false alarms can occur. The problem of computing the feasible set of parameters when some or all explanatory variables, as well as the output, are uncertain is usually called the Error-In-Variables (EIV) problem. The study of this problem can be found in (Milanese *et al.*, 1996). The EIV parameter-bounding algorithm can be adapted for use also with the Takagi-Sugeno neuro-fuzzy model (Kowal, 2005; Kowal and Korbicz, 2006a).

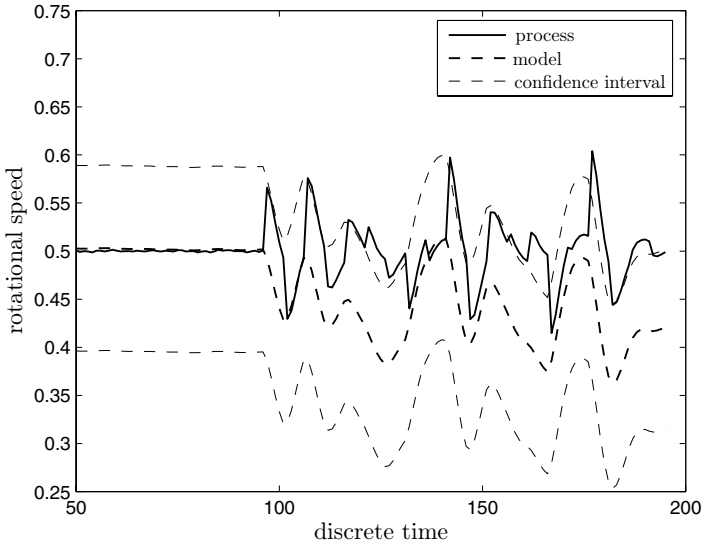


Fig. 9.14. Process and model output: small fault f_1

The future work will concentrate on the extension of the presented approach for the Takagi-Sugeno neuro-fuzzy network with consequences in the form of ARX models and the development of algorithms which assure the minimization of Takagi-Sugeno model uncertainty.

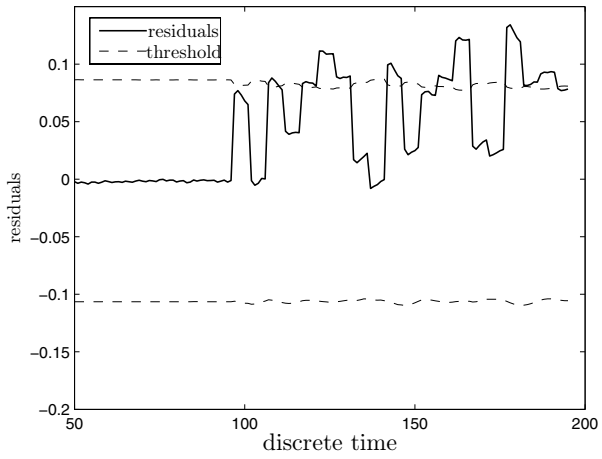


Fig. 9.15. Residuals: small fault f_1

9.7. Conclusions

In the field of fault diagnosis of complex and dynamic systems, one is faced with the problem that no or insufficiently accurate mathematical models of the system are available. In recent years, a rapid development from the well-established but in terms of limited efficiency and applicability traditional methods of model-based fault diagnosis to artificial intelligence methods has been observed. It is clear that FDI system design is related to many optimization problems. They are nonlinear, multi-modal, and usually multi-objective, and because of this the conventional "local" optimization methods fail to solve them, while evolutionary algorithms seem to be very attractive alternative direct search methods, which overcome the limitations of the conventional optimization methods. In this chapter, the state of the art of soft computing approaches to FDI optimal design has been discussed.

References

Abonyi J., Babuška R. and Szeifert F. (2002): *Modified Gath-Geva fuzzy clustering for identification of Takagi-Sugeno fuzzy models.* — IEEE Trans. Systems, Man, and Cybernetics, Part B, Vol. 32, No. 5, pp. 612–621.

Amman P. and Frank P.M. (1997): *Model building in observers for fault diagnosis.* — Proc. 5-th IMACS World Congress *Scientific Computation, Modelling and Applied Mathematics*, Berlin, Germany pp. 24–29.

Andrzejewski M. and Pieczyński A. (2005): *Application of the artificial intelligence to fuzzy model parameters tuning.* — Proc. 12-th *Zittau Fuzzy Colloquium*, Zittau, Germany, pp. 226–230.

Atkinson A.C. and Donev A.N. (1992): *Optimum Experimental Designs.* — New York, USA: Oxford University Press.

Babuška R. (1998): *Fuzzy Modeling for Control.* — London: Kluwer Academic Publisher.

Bäck, T., Fogel, D.B., and Michalewicz, Z. (Eds.) (1997): *Handbook of Evolutionary Computation.* — New York: Institute of Physics Publishing and Oxford University Press.

Bates D.M. and Watts D.G. (1980): *Nonlinear Regression Analysis and Its Application.* — New York: Willey & Sons.

Bezdek D. (1981): *Pattern Recognition with Fuzzy Objective Function Algorithms.* — New York: Plenum Press.

Calado J.M.F., Louro R., Mendes M.J.G.C., Sa da Costa J.M.G. and Kowal M. (2003): *Fault isolation based on HSFNN applied to DAMADICS benchmark problem.* — Proc. 5-th IFAC Symp. *Fault Detection, Supervision and Safety of Technical Processes, SAFEPROCESS*, Washington, DC, USA, pp. 1053–1058.

Chen J. and Patton R.J. (1999): *Robust Model-Based Fault Diagnosis for Dynamic Systems.* — Boston: Kluwer Academic Publishers.

Chen J-Q., Xi Y-G. and Zhang Z-J. (1998): *A clustering algorithm for fuzzy model identification.* — Fuzzy Sets and Systems, Vol. 98, pp. 319–329.

Chryssolouris G., Lee M. and Ramsey A. (1996): *Confidence interval prediction for neural network models.* — IEEE Trans. Neural Networks, Vol. 7, No. 1, pp. 229–232.

DAMADICS (2002): *Website of the RTN DAMADICS: Development and Application of Methods for Actutator Diagnosis in Industrial Control Systems.* — `http://diag.mchtr.pw.edu.pl/damadics`

Donaldson J.R., Schnabel R.B. (1987): *Computational experience with confidence regions and confidence intervals for nonlinear least squares.* — Technometrics, Vol. 29, No. 1, pp. 67–82.

Duch W., Korbicz J., Rutkowski L. and Tadeusiewicz R. (Eds.) (2000): *Biocybernetics and Biomedical Engineering 2000. Neural Networks.* — Warsaw: Akademicka Oficyna Wydawnicza EXIT, Vol. 6, (in Polish)

Esparcia-Alcazar A.I. (1998): *Genetic Programming for Adaptive Digital Signal Processing.* — Ph.D. dissertation, Glasgow University.

Frank P.M. (1998): *Fault diagnosis in dynamic systems using analytical and knowledge-based redudancy – a survey and some new results.* — Automatica, Vol. 26, pp. 459–474.

Frank P.M. and Köppen-Seliger B. (1997): *New developments using AI in fault diagnosis.* — Artificial Intelligence, Vol. 10, No. 1, pp. 3–14.

Goldberg D.E. (1989): *Genetic Algorithms in Search, Optimization and Machine Learning.* — Massachusetts: Addison–Wesley.

Gupta M.M., Jin L. and Homma N. (2003): *Static and Dynamic Neural Networks: From Fundamentals to Advanced Theory.* — New Jersey: John Wiley and Sons.

Kiupel N. and Frank P.M. (1993): *Process supervision with the aid of fuzzy logic.* — Proc. IEEE Int. Conf. *Systems, Man and Cybernetics*, Le Touquet, France, pp. 409–414.

Köppen-Seliger B. and Frank P.M. (1999): *Fuzzy logic and neural networks in fault detection*, In: Fusion of Neural networks, Fuzzy Sets, and Genetic Algorithms (Jain L. and Martin N. (Eds.). — New York: CRC Press, pp. 169–209.

Korbicz J. and Bidyuk P.I. (1993): *State and Parameter Estimation. Digital and Optimal Filtering, Applications.* — Technical University of Zielona Góra Press.

Korbicz J. and Kowal M. (2001): *Neuro-Fuzzy Systems in Processes Diagnostics*, In: Fuzzy Sets and their Applications (Chojcan J. and Łęski J., Eds.). — Gliwice, Poland: Wydawnictwo Politechniki Śląskiej, pp. 333–379, (in Polish)

Korbicz J., Kościelny J.M., Kowalczuk Z. and Cholewa W. (Eds.) (2004): *Fault Diagnosis. Models, Artificial Intelligence, Applications.* — Berlin, Heidelberg: Springer-Verlag.

Korbicz J., Obuchowicz A. and Patan K. (1998): *Network of dynamic neurons in fault detection systems.* — Proc. IEEE Int. Conf. *System, Man, Cybernetics*, San Diego, USA, pp. 1862–1867.

Korbicz J., Obuchowicz A. and Uciński D. (1994): *Artificial Neural Networks. Fundations and Applications.* — Warsaw: Akademicka Oficyna Wydawnicza PLJ, (in Polish).

Korbicz J., Patan K., and Obuchowicz A. (1999): *Dynamic neural networks for process modelling in fault detection and isolation systems.* — Int. J. Appl. Math. and Comp. Sci., Vol. 9, No. 3, pp. 510–546.

Kosiński W., Michalewicz Z., Weigl M. and Koleśnik R. (1998): *Genetic algorithms for preprocessing of data for universal approximators*, In: Intelligent Information Systems (Kłopotek M., Michalewicz M. and Raś Z.W., Eds.). — Warsaw: Polish Academy of Sciences Press, pp. 320–331.

Kowal M. (2001): *Application of the time window method and neuro-fuzzy networks to fault diagnostic.* — Proc. 5-th Nat. Conf. *Diagnostics of Industrial Processes, DPP*, Zielona Góra, Poland, pp. 199–204, (in Polish).

Kowal M. (2005): *Optimization of Neuro-Fuzzy Structures in Technical Diagnostics Systems.* — Serie: Lecture Notes in Control and Computer Science, Vol. 9, Technical University of Zielona Góra Press.

Kowal M. and Korbicz J. (2002a): *Fault detection using neuro-fuzzy networks.* — Proc. 14-th Polish Conf. *Automatic Control, KKA*, Zielona Góra, Poland, pp. 595–600, (in Polish).

Kowal M. and Korbicz J. (2002b): *Faul detection using Takagi-Sugeno neuro-fuzzy networks and ARX models.* — Nat. Conf. *Fuzzy Systems*, Cracow, Poland, pp. 198–205.

Kowal M. and Korbicz J. (2003): *Self-organizing Takagi-Sugeno fuzzy model in a fault detection system.* — Proc. 6-th Nat. Conf. *Diagnostics of Industrial Processes, DPP'03*, Władysławowo, Poland, pp. 253–258, (in Polish).

Kowal M. and Korbicz J. (2005a): *Robust fault detection under fuzzy-neural model uncertainty.* — Pomiary, Automatyka, Kontrola, No. 9, pp. 93–95, (in Polish).

Kowal M. and Korbicz J. (2005b): *Neuro-fuzzy structures in FDI system.* — Proc. 16-th *IFAC World Congress*, Prague, Czech Republic, CD-ROM.

Kowal M. and Korbicz J. (2005c): *Fault detection using neuro-fuzzy networks and bounded error approaches.* — Proc. 6-th Int. Conf. *Methods and Models in Automation and Robotics, MMAR*, Międzyzdroje, Poland, CD-ROM.

Kowal M. and Korbicz J. (2006a): *Fault detection under fuzzy model uncertainty.* — Proc. 6-th IFAC Symp. *Fault Detection, Supervision and Safety of Technical Processes, SAFEPROCESS*, Beijing, China, pp. 775–780, CD-ROM.

Kowal M. and Korbicz J. (2006b): *Robust fault detection using neuro-fuzzy models.* — Przegląd Elektrotechniczny, No. 1, pp. 32–36.

Kowal M., Korbicz J., Mendes M.J.G.C. and Calado J.M.F. (2002): *Fault detection using neuro-fuzzy networks.* — Systems Science, Vol. 28, No. 1, pp. 45–57.

Koza J.R. (1992): *Genetic Programming: On the Programming of Computers by Means of Natural Selection.* — Cambridge: The MIT Press.

Kwaśnicka H. (1999): *Evolutionary Computation in Artificial Intelligence.* — Wrocław: Wrocław University of Technology Press, (in Polish).

Leith D.J. and Leithead W.E. (1999): *Analytic framework for blended multiple model systems using local models.* — Int. J. Control, Vol. 72, No. 7/8, pp. 605–619.

Mendes M.J.G.C., Kowal M., Korbicz J. and Sa da Costa J.M.G. (2002): *Neuro-fuzzy structures in FDI system.* — Proc. 15-th *IFAC Triennial World Congress*, Barcelona, Spain, CD-ROM.

Michalewicz Z. (1996): *Genetic Algorithms + Data Structures = Evolution Programs.* — Berlin: Springer-Verlag.

Milanese M., Norton J.P., Piet-Lahanier H. and Walter E. (1996): *Bounding Approaches to Identification.* — New York: Plenum Press.

Mrugalski M. (2003): *Neural Network Based Modelling of Nonlinear Systems in Fault Detection Schemes.* — Ph.D. thesis, University of Zielona Góra, Faculty of Electrical Engineering, Computer Sience and Telecommunications, (in Polish).

Obuchowicz A. (1999a): *Architecture optimization of a network of dynamic neurons using the $A^\star$ algorithm.* — Proc. 7-th European Congress *Intelligent Techniques and Soft Computing, EUFIT'99*, Aachen, Germany, CD-ROM.

Obuchowicz A. (1999b): *Evolutionary search with soft selection in training a network of dynamic neurons*, In: Intelligent Information Systems: Proc. Workshop (M. Kłopotek and M. Michalewicz, Eds.). — Warsaw: Polish Academy of Sciences Press, pp. 214–223.

Obuchowicz A. (2000): *Optimization of neural networks architectures*, In: Biocybernetics and Biomedical Engineering 2000. Neural Networks (Duch W., Korbicz J., Rutkowski L., and Tadeusiewicz R., Eds.). — Warsaw: Akademicka Oficyna Wydawnicza EXIT, pp. 323–368, (in Polish).

Obuchowicz A. (2003): *Evolutionary Algorithms for Global Optimization and Dynamic System Diagnosis.* — Zielona Góra: Lubuskie Towarzystwo Naukowe.

Obuchowicz A. and Korbicz J. (2002): *Evolutionary methods in diagnosis systems design*, In: Process Diagnosis. Models, Methods of Artificial Inteligence, Applications (Korbicz J., Kościelny J.M., Kowalczuk Z. and Cholewa W., Eds.). — Warsaw: Wydawnictwo Naukowo-Techniczne WNT, pp. 279–309, (in Polish).

Obuchowicz A. and Patan K. (1997): *About some modifications of evolutionary search with soft selection algorithm.* — Proc. 2-nd Conf. *Evolutionary Algorithms and Global Optimization*, Rytro, Poland, pp. 193–200.

Obuchowicz A. and Politowicz K. (1997): *Evolutionary algorithms in optimization of a multilayer feedforward neural network architecture.* — Proc. 4-th Int. Symp. *Methods and Models in Automation and Robotics, MMAR*, Międzyzdroje, Poland, Vol. 2, pp. 739–743.

Patan K. and Jesionka M. (1999): *Genetic algorithms approach to network of dynamic neurons training.* — Proc. 3-rd National Conf. *Evolutionary Algorithms and Global Optimization*, Potok Złoty, Poland, pp. 261–268.

Patan K. and Korbicz J. (2000): *Dynamical network and their application to modelling and identification*, In: Biocybernetics and Biomedical Engineeering. Neural Networks (Duch W., Korbicz J., Rutkowski L. and Tadeusiewicz R., Eds.). — Warsaw: Akademicka Oficyna Wydawnicza EXIT, Vol. 6, pp. 389–417, (in Polish).

Patan K., Obuchowicz A. and Korbicz J. (1999): *Cascade network of dynamic neurons in fault detection systems.* — Proc. 5-th *European Control Conf., ECC*, Karlsruhe, Germany, CD-ROM.

Patton R.J. (1993): *Robustness issues in fault-tolerant control.* — Proc. Int. Conf. *Fault Diagnosis, TOOLDIAL*, Toulouse, France, pp. 1081–1117.

Patton R.J. and Chen J. (1997): *Observer-based fault detection and isolation: robustness and applications.* — Control Eng. Practice, Vol. 5, No. 5, pp. 671–682.

Patton R.J., Frank P. and Clark R.N. (Eds.) (2000): *Issues of Fault Diagnosis for Dynamic Systems.* — Berlin: Springer-Verlag.

Pieczyński A. (2001): *Fuzzy modeling of multidimensional non-linear processes – tuning algorithms.* — Proc. 9-th *Zittau Fuzzy Colloquium*, Zittau, Germany, pp. 94–104.

Pieczyński A. (2002): *Parallel and cascade structures of multidimensional fuzzy model.* — System Science, Vol. 28, No. 4, pp. 17–34.

Pieczyński A. (2003): *Knowledge Representation in the Expert Diagnostic System.* — Zielona Góra: Lubuskie Towarzystwo Naukowe, (in Polish).

Pieczyński A. (2006): *Fuzzy model structure and parameters optimization with evolutionary algorithm application.* — Proc. 13-th *Zittau Fuzzy Colloquium*, Zittau, Germany, pp. 34–39.

Pieczyński A. and Obuchowicz A. (2004): *Application of the general gaussian membership function for the fuzzy model parameters tunning.* — Lecture Notes in Artificial Intelligence: Artificial Intelligence and Soft Computing, ICAISC, Vol. 3070, pp. 350–355.

Piegat A. (2003): *Modelling and Fuzzy Systems.* — Warsaw: Akademicka Oficyna Wydawaniczna EXIT, (in Polish).

Prętki P. and Obuchowicz A. (2006): *Evolutionary bands for the expected response in nonlinear regression.* — Prace Naukowe Politechniki Warszawskiej, Elektronika, Vol. 156, pp. 359–364.

Prętki P. and Witczak M. (2005a): *Assessment and minimization of parametric uncertainty for multi-output neural networks – application to fault diagnosis*, In: Recent Developments in Artificial Intelligence Methods, AI-METH. — Gliwice, Poland, pp. 155–160.

Prętki P. and Witczak M. (2005b): *Developing measurement selection strategy for neural network models.* — Lecture Notes in Computer Science: Artificial neural networks: Formal Models and Their Applications, ICANN, Berlin: Springer, Part II, Vol. 3697, pp. 79–84.

Rutkowska D. (2000): *Evolutionary and genetic algorithms*, In: Biocybernetics and Biomedical Engineeering. Neural Networks (Duch W., Korbicz J., Rutkowski L. and Tadeusiewicz R., Eds.). — Warsaw: Akademicka Oficyna Wydawnicza EXIT, Vol. 6, pp. 691–732, (in Polish).

Rutkowska D. (2002): *Neuro-Fuzzy Architectures and Hybrid Learning.* — Heidelberg, New York: Physica-Verlag, Springer-Verlag Company.

Skowroński K. (1998) *Genetic rules induction based on MDL principle*, In: Intelligent Information Systems (M. Kłopotek, M. Michalewicz and Z.W. Raś, Eds.). — Warsaw: Polish Academy of Sciences Press, pp. 356–360.

Walter E. and Pronzato L. (1997): *Identification of Parametric Models from Experimental Data.* — Berlin: Springer-Verlag.

Witczak M. (2003): *Identification and Fault Detection of Non-linear Dynamic Systems.* — Serie: Lecture Notes in Control and Computer Science, Technical University of Zielona Góra Press, Vol. 1.

Witczak M. (2006): *Towards training of feed-forward neural networks with the D-optimum input sequence.* — IEEE Trans. Neural Networks, Vol. 17, No. 2, pp. 357–373.

Witczak M. and Korbicz J. (2000): *Genetic programming based observers for nonlinear systems.* — Proc. 4-th IFAC Symp. *Fault Detection, Supervision and Safety of Technical Processes, SAFEPROCESS*, Budapest, Hungary, pp. 967–972.

Witczak M. and Prętki P. (2005): *Designing neural-network-based fault detection systems with D-optimum experimental conditions.* — Computer Assisted Mechanics and Engineering Sciences, No. 12, pp. 279–291.

Witczak M., Obuchowicz A. and Korbicz J. (1999): *Design of non-linear state observers using genetic programming.* — Proc. 3-rd Nat. Conf. *Evolutionary Algorithms and Global Optimization*, Potok Złoty, Poland, pp. 345–352,

Witczak M., Obuchowicz A. and Korbicz J. (2002): *Genetic programming based approaches to identification and fault diagnosis of nonlinear dynamic systems.* — Int. J. Control, Vol. 75, No. 13, pp. 1012–1031.

Uciński D. (2004): *Optimal Measurements Methods for Distributed Parameter System Identification.* — New York: CRC Press.

Zolghardi A., Henry D. and Monision M. (1996): *Design of nonlinear observers for fault diagnosis. A case study.* — Control. Eng. Practice, Vol. 4, No. 11, pp. 1535–1544.

Chapter 10

LINEAR REPETITIVE PROCESSES AND MULTIDIMENSIONAL SYSTEMS

Krzysztof GAŁKOWSKI*, Wojciech PASZKE*, Bartłomiej SULIKOWSKI*

10.1. Introduction

In contrast to the classic 1D systems, two-dimensional (2D) systems are characterized by two indeterminates. In the classic theory of 1D systems, the independent variable used in the state-space description in most cases denotes time (discrete or continuous), and in 2D systems independent variables can be treated as a vector time, or the first indeterminate denotes time and the second has the space meaning (the coordinate or the number of the current process phase, iteration or the trail). It can be also said that in 2D systems there are 2 independent directions in which information propagates.

During the last years, 2D systems (or, in general, nD) have been found interesting from both theoretical and practical application standpoints. There can be found a number of books and papers regarding the class of systems considered (see, e.g., Du and Xie, 2002; Gałkowski and Wood, 2001; Kaczorek, 1985 and the references therein). In every case when the system considered is not suitable to be modeled using the well-known 1D models, 2D (nD) models are a very strong alternative. nD (2D) systems have been found useful in modeling physical processes in the areas of control, computer science, telecommunications, acoustics, electrical engineering etc. Particular applications include nD filtering (Basu, 2002), nD coding and decoding (Shi and Zhang, 2002), image processing (Bracewell, 1985), and multidimensional signal processing (Dudgeon and Merserau, 1984).

In the theory of 2D systems there arise several obstacles and limitations which are closely connected to the lack of mathematical tools (or their very complicated forms). What one gets in response is the fact that the application of 2D models to the description of the phenomena studied provides some new possibilities that were unavailable when considering 1D models.

* Institute of Control and Computation Engineering
e-mails: {k.galkowski, w.paszke, b.sulikowski}@issi.uz.zgora.pl

A special case of 2D systems are Linear Repetitive Processes (LRPs) (Rogers and Owens, 1992; Rogers *et al.*, 2007). An LRP is defined by a repetitive execution of an action which lasts for a fixed finite duration. During each iteration (or, as it is called in the context of that class of systems, a pass), an output, called the pass profile, is produced and it acts as a forcing function on the next pass profile. Hence there are twofold dynamics in the model of LRP, i.e., those which regard the direction from pass to pass, and those which regard the direction along the pass.

Applications of LRPs include long-wall coal cutting (Rogers and Owens, 1992; Rogers *et al.*, 2007), metal rolling (Rogers and Owens, 1992; Sulikowski *et al.*, 2005; Yamada and Saito, 1996), Iterative Learning Control (ILC) schemes (Amann *et al.*, 1998; Longman, 2000), and iterative algorithms for solving non-linear dynamical optimal control problems based on the maximum principle (Roberts, 2002). Recently, the link between spatially interconnected systems (D'Andrea and Dullerud, 2003) and LRPs has been recognized and it seems that many of the results obtained for LRPs can be adopted for that class of 2D (nD) systems.

The analysis and synthesis of complex multidimensional dynamical systems and, in particular, LRPs belong to "practically" unsolvable problems. However, research problems related to this area have been recently partially stated and solved in (Paszke, 2005; Sulikowski, 2006). Despite the fact that strong theoretical results in this area can be found, due to significant mathematical limitations (mainly because of that there is no division algorithm of multivariable polynomials), they do not form a computationally efficient methodology. However, it has turned out that, based on the recent results of algorithmic theories strengthened by modern computer science, there appears the possibility to develop sufficiently effective approximations, which was the main aim of (Paszke, 2005; Sulikowski, 2006).

In general, one of the very crucial properties of the dynamical system is stability. For 2D systems, the results regarding asymptotic stability are presented here. For LRPs, this chapter deals with two basic types of stability, i.e., asymptotic stability and stability along the pass (Rogers and Owens, 1992). Hence the emphasis here is put on stability investigation (analysis) and stabilization (synthesis) of 2D systems and LRPs. Note that due to the fact that LRPs are a distinct class of 2D systems, the analysis and synthesis tasks require the developing and application of appropriate methods, which, in general, differ either from the methodology provided for the classic 1D systems and/or the results obtained for 2D systems (Du and Xie, 2002; Kaczorek, 1985).

Another set of problems which appear also in the classic 1D systems theory is the integration of the basic analysis/synthesis tasks with stronger requirements. Those introduce additional restrictions to the problem considered and hence can cause additional problems from both theoretical (formulating appropriate conditions) and practical (numerical problems) standpoints. The aforementioned aspects considered in this chapter include topics related to:

- static and dynamic, state and output based control,
- $\mathcal{H}_2$ and/or $\mathcal{H}_\infty$ control,
- addressing the uncertainties which can appear in the models of 2D systems/LRPs.

Note that those supplementary topics have to be treated as ones introducing additional constraints to the basic problems (stability). Frequently, it is necessary to "pay" for solving such extended problems by decreasing the area of possible solutions and, in many cases, increasing the numerical effort to obtain the solution.

It is necessary to understand that the studied issues of analysis/synthesis can cause serious problems from the theoretical and numerical standpoints. Although several conditions regarding the stability of 2D systems/LRPs have been presented, e.g., in (Agathoklis *et al.*, 1993; Rogers and Owens, 1992), there exist serious limitations in the application of those conditions. The obstacles come from the fact that the known conditions deal mainly with the 2D transfer function. Since for a 2D system the poles of a transfer function are curves on the complex plane (i.e., not isolated points as in the 1D case), there arise serious difficulties with stability analysis and stabilization for this class of systems. Due to this, a new efficient methodology for the analysis and synthesis of 2D systems/LRPs is required. One of the possible solutions to these crucial problems comes from the Lyapunov theory, strengthened by the fact that there have recently appeared numerically efficient methods of Linear Matrix Inequalities (LMIs) (Boyd *et al.*, 1994), which are based on interior point methods convex optimization algorithms. For 2D systems, LMIs turn out to provide a very efficient method (the solution provided in the polynomial time) to solve the problem of stability investigation that comes directly from the Lyapunov method. What is more, an easy and natural extension to stabilization can also be provided here (see, e.g., Gałkowski *et al.*, 2003a; Paszke, 2005; Sulikowski, 2006). The drawback of the application of LMIs to the discussed problems is that the conditions defined in terms of LMIs for 2D systems (LRPs) are only sufficient. Nevertheless, a method to lower the conservativeness of those conditions and to finally get closer to necessary and sufficient LMI conditions for the stability of 2D systems was presented recently in (Bliman, 2002).

Note that even if the analysis and/or synthesis of LRPs can be shown to be polynomial time solvable, those tasks can cause serious problems regarding the numerics. It is especially apparent when the systems considered are highly dimensioned. Then even the application of the polynomial time method can be insufficient to provide the solution accurately. This aspect becomes visible when taking into account the fact that those problems are solved using computers. In view of computer aided analysis/synthesis, the following topics appear: storing the data describing the studied problem in the memory, performing computations according to the analysis/synthesis tasks considered and, finally, simulating the system. The second point is the most demanding one. Hence, to provide an adequate computational power for solving analysis/synthesis problems efficiently, computer clusters have been used. It is important to note that due to the fact that the topics considered are highly specialist, there are no appropriate software packages available that would allow solving those kinds of problems directly. Hence the method of how to reformulate analysis/synthesis problems into a form solvable by the existing cluster software can be treated as the original and practical result of this chapter.

In practical applications, the question about controlling 2D systems/LRPs with a required performance appears. There have been published some preliminary results regarding stability aspects, but here the synthesis problem governing additional

properties ("beyond" stability) of the system in the closed loop configuration, e.g., ensuring the prescribed stability margins or the assurance of the prescribed form of the closed loop system are considered. Further extensions include the application of the developed schemes in practice, where natural goals to be achieved can be defined as driving the system considered to the required output (called the reference signal) and disturbance rejection.

Throughout this chapter, the null matrix and the identity matrix with appropriate dimensions are denoted by 0 and I, respectively. Moreover, for any real symmetric matrices X and Y, the notation $X \succeq Y$ (respectively $X \succ Y$) means that the matrix $X - Y$ is positive semi-definite (respectively positive definite). In long matrix expressions, the symbol $(\star)$ replaces terms that are induced by symmetry.

10.2. Models of 2D systems and repetitive processes

The last three decades have shown a very rapid development of 2D systems theory and applications based upon them. In the area of automatic control, the frequently used state-space models are the 2D Roesser model (see Roesser, 1975) and the 2D Fornasini-Marchesini model (see Fornasini and Marchesini, 1978). Note that, similarly to the classic 1D state-space models, there can be distinguished the state and the output equations as well. As has been mentioned, another distinct sub-class of two-dimensional systems are LRPs (for the references see, e.g., Paszke, 2005; Rogers and Owens, 1992; Rogers *et al.*, 2007; Sulikowski, 2006).

The 2D nature of LRPs allows considering either discrete or hybrid processes (discrete-continuous) which are presented in the following sections.

10.2.1. Discrete LRPs

Following (Rogers and Owens, 1992), the state-space model of a discrete linear repetitive process has the following form over $0 \leq p \leq \alpha - 1$, $k \geq 0$:

$$x_{k+1}(p+1) = Ax_{k+1}(p) + B_0 y_k(p) + Bu_{k+1}(p), \tag{10.1}$$

$$y_{k+1}(p) = Cx_{k+1}(p) + D_0 y_k(p) + Du_{k+1}(p), \tag{10.2}$$

where $\alpha < +\infty$ denotes the pass length, $0 \leq p \leq \alpha - 1 \ \in \mathbb{Z}_+$ is the discrete position on the current pass, $k \in \mathbb{Z}_+$ is the current pass number, $x_k(p) \in \mathbb{R}^n$ is the state vector, $y_k(p) \in \mathbb{R}^m$ is the pass profile (output) vector, $u_k(p) \in \mathbb{R}^r$ is the input vector, $A,\ B_0,\ C,\ D_0,\ B,\ D$ are matrices of appropriate dimensions.

To complete the process description, it is necessary to specify the "initial conditions", termed the boundary conditions here, i.e., the state initial vector on each pass and the initial pass profile. The simplest possible form for these is

$$\begin{aligned} x_{k+1}(0) &= d_{k+1}, \quad k \geq 0, \\ y_0(p) &= f(p), \end{aligned} \tag{10.3}$$

where $d_{k+1} \in \mathbb{R}^n$ is the vector and the entries in the vector $f(p) \in \mathbb{R}^m$ are known functions of p.

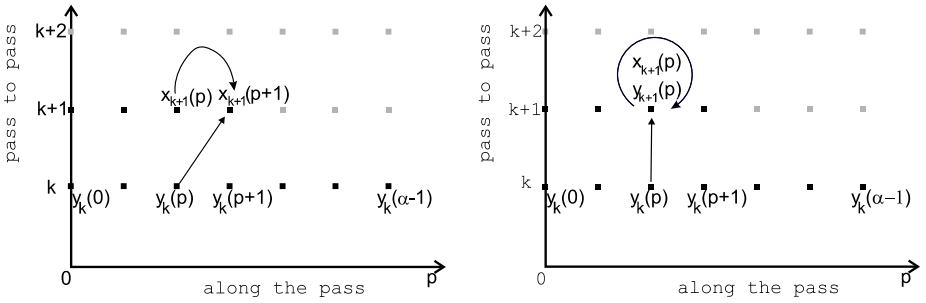


Fig. 10.1. State (a) and pass (b) profile vector updating structure of (10.1)–(10.2)

Figure 10.1 illustrates the updating structure of the state and pass profile vectors in (10.1)–(10.2). The characteristic polynomial for (10.1)–(10.2) is defined as follows:

$$\mathcal{C}_{\text{discreteLRP}} = \det\left(\begin{bmatrix} I - z_1 A & -z_1 B_0 \\ -z_2 C & I - z_2 D_0 \end{bmatrix}\right), \tag{10.4}$$

where $z_1, z_2 \in \mathbb{C}$ are the inverses of z-transform variables in the horizontal and vertical directions, respectively. They can be also considered as unit delay operators in those directions. They are defined as follows:

$$x_k(p) := z_1 x_k(p+1), \quad x_k(p) := z_2 x_{k+1}(p). \tag{10.5}$$

10.2.2. Differential LRPs

The differential LRP has the following form over $0 \le t < \alpha$, $k \ge 0$ (Rogers and Owens, 1992):

$$\dot{x}_{k+1}(t) = A x_{k+1}(t) + B_0 y_k(t) + B u_{k+1}(t), \tag{10.6}$$

$$y_{k+1}(t) = C x_{k+1}(t) + D_0 y_k(t) + D u_{k+1}(t), \tag{10.7}$$

where $0 \le t < \alpha \ \in \mathbb{R}_+ \cup \{0\}$ is the continuous position on the current pass, $k \in \mathbb{Z}_+$ is the current pass number, $x_k(t) \in \mathbb{R}^n$ is the state vector, $y_k(t) \in \mathbb{R}^m$ is the pass profile (output) vector, $u_k(t) \in \mathbb{R}^r$ is the input vector, A, B_0, C, D_0, B, D are matrices of appropriate dimensions.

It is clear that the process is continuous along the pass and discrete from pass to pass.

Again, to complete the process description, it is necessary to specify the "initial conditions", termed the boundary conditions here, i.e., the state initial vector on each pass and the initial pass profile. The simplest possible form for these is

$$\begin{aligned} x_{k+1}(0) &= d_{k+1}, \quad k \ge 0, \\ y_0(t) &= f(t), \end{aligned} \tag{10.8}$$

where $d_{k+1} \in \mathbb{R}^n$ are known vectors and $f(t) \in \mathbb{R}^m$ is the vector valued function, which generates appropriate $f(t) \in \mathbb{R}^m$ for given t.

The characteristic polynomial for (10.6)–(10.7) is defined as follows:

$$\mathcal{C}_{\text{diffLRP}} = \det\left(\begin{bmatrix} sI - A & -B_0 \\ -zC & I - zD_0 \end{bmatrix}\right), \tag{10.9}$$

where $s \in \mathbb{C}$ is the Laplace transform indeterminate and $z \in \mathbb{C}$ comes, as before, from the use of the z-transform in the direction from pass to pass.

There are some research works (see, e.g., Owens and Rogers, 1999) where the definition and influence of dynamic boundary conditions for LRPs (discrete and differential) are considered in detail; however, these results are not given here due to the fact that the sequel of this chapter does not concern those topics.

For the purpose of sequel requirements, define the so-called 2D system plant and the 2D extended input matrix of the studied models of LRPs as follows:

$$\Upsilon = \begin{bmatrix} A & B_0 \\ C & D_0 \end{bmatrix}, \quad \Omega = \begin{bmatrix} B \\ D \end{bmatrix}. \tag{10.10}$$

10.3. Stability conditions

When considering the dynamical system, the very first thing taken into account is its stability. The same thing appears when investigating the properties of 2D systems/LRPs. Then the basic problems are: how to define stability, how to test it (analysis) and, finally, what to do when the system has been affirmed to be unstable (synthesis). In contrast to the classic 1D systems (either discrete or continuous time), for 2D (nD) systems the analysis/synthesis problems are sophisticated and require the application of efficient methods to be solved. The analysis itself is performed to determine if the tested 2D system (an LRP) can be left without any external input (the so-called free evolution of the system), and the synthesis task has the main goal of how to drive the unstable system to stability. The supplementary goal can be defined as ensuring the required performance of the controlled system.

Stability theory (Rogers and Owens, 1992) for LRPs is based on an abstract model of process dynamics in a Banach space (here denoted by E_α) of the form

$$y_{k+1} = L_\alpha y_k + b_{k+1}, \quad k \geq 0. \tag{10.11}$$

In this model, $y_k \in E_\alpha$ denotes the pass profile on the pass k, L_α is a bounded linear operator which maps E_α into itself and $b_{k+1} \in W_\alpha$, where W_α is a linear subspace of E_α. Also, the term $L_\alpha y_k$ describes the contribution of the pass k to the pass $k+1$, and b_{k+1} represents the inputs and other effects which enter on the current pass.

In fact, two distinct forms of stability can be defined in this setting, which are termed asymptotic stability and stability along the pass. The former requires this property with respect to the (finite and fixed) pass length and the latter uniformly, i.e., independently of the pass length. Asymptotic stability guarantees the existence of the so-called limit profile defined as the strong limit as $k \to \infty$ of the sequence

$\{y_k\}_k$ and for the processes under consideration here this limit profile is described by a 1D differential linear systems state-space model with state matrix $A_{lp} := A+B_0(I-D_0)^{-1}C$. Hence it is possible for asymptotic stability to result in a limit profile which is unstable as a 1D differential linear system, e.g., $A=-1,\ B=0,\ B_0=1+\beta,\ C=1,\ D=0,\ D_0=0$, where $\beta>0$ is a real scalar. Stability along the pass prevents this from happening by demanding that the stability property be independent of the pass length, which can be analyzed mathematically by letting $\alpha\to\infty$.

The cases where the limit profile is unstable as a 1D linear system are not acceptable. Hence a stronger concept of stability, i.e., stability along the pass must be used. This stronger stability demands the BIBO property to hold independently of dynamics, i.e., in the direction along the pass (p or t) and from pass to pass (k). Introduce the formal definition of stability along the pass as follows:

Definition 10.1. (Rogers and Owens, 1992) In terms of the abstract model of (10.11), stability along the pass holds provided that there exist real numbers $M_\infty>0$ and $\lambda_\infty\in(0,1)$, which are independent of α such that $\|L_\alpha^k\|\le M_\infty\lambda_\infty^k,\ k\ge 0$.

In terms of characteristic polynomials, stability along the pass can be characterized as follows (Benton, 2000; Rogers and Owens, 1992):

Theorem 10.1.

- *A discrete LRP with the characteristic polynomial defined as (10.4) is stable along the pass if and only if*

$$\mathcal{C}_{\text{discreteLRP}}\neq 0\ \ \forall(z_1,z_2):\ |z_1|\le 1,\ |z_2|\le 1. \tag{10.12}$$

- *A differential LRP with the characteristic polynomial defined as (10.9) is stable along the pass if and only if*

$$\mathcal{C}_{\text{diffLRP}}\neq 0\ \ \forall(s,z):\ \text{Re}(s)\ge 0,\ |z|\le 1. \tag{10.13}$$

The equivalent condition for Theorem 10.1 for stability along the pass of the discrete LRP of (10.1)–(10.2) takes the following form:

Theorem 10.2. (Benton, 2000; Rogers and Owens, 1992) *The discrete LRP of (10.1)–(10.2) is stable along the pass if the following hold: $r(D_0)<1$, $r(A)<1$, and all eigenvalues of the transfer function*

$$G(z)=C(zI-A)^{-1}B_0+D_0, \tag{10.14}$$

$\forall\,|z|=1$, *have moduli strictly smaller than unity.*

Note here that for this result, $z:=z_1^{-1}$ (see (10.4)).

For the differential case of (10.6)–(10.7), the following counterpart of Theorem 10.2 is presented:

Theorem 10.3. (Benton, 2000; Rogers and Owens, 1992) *The differential LRP of (10.6)–(10.7) is stable along the pass if the following hold: $r(D_0)<1$, $Re(\lambda_i(A))<0\ i=1,\dots,n$, where $\lambda_i(\cdot)$ denotes the ith eigenvalue of $(\cdot)$, and all eigenvalues of the transfer function*

$$G(s)=C(sI-A)^{-1}B_0+D_0,$$

$\forall\ s=\imath\omega,\ \omega\ge 0$, *have moduli strictly smaller than unity.*

Remark 10.1. *As has been mentioned, there arises the question of the applicability of the stability conditions presented in this section. It turns out that they are hard to apply in practice or even, in some cases, unapplicable. This is due to the fact that those conditions require dealing with polynomials in two variables (Theorem 10.1), and since there are no sufficient methods for dividing such polynomials those results remain rather theoretical. On the other hand, the conditions given in Theorems 10.2 and 10.3 require checking all possible complex numbers satisfying some constraints. Since there is an infinite amount of such complex numbers, it is straightforward to conclude that those conditions remain to be of theoretical significance only, too.*

Another considerable difficulty arises in defining synthesis (controller design towards stability along the pass) using the above conditions, and this fact also limits their applicability.

10.4. LMI conditions towards stability/stabilization

As has been mentioned, the previously presented stability conditions are necessary and sufficient, but it is also important to remember that they are practically unapplicable. Hence it is necessary to develop new methods which can be used in practice. Here the Lyapunov approach can be very helpful. As it is known, the Lyapunov method provides conditions which take the form of a matrix inequality and hence can be reformulated into the proper LMI condition. Problems formulated as those conditions can be solved using numerically efficient algorithms (based on the interior point method algorithm). However, it is important to understand that the resulting LMI conditions are only sufficient ones.

Below, the LMI conditions for the stability investigation and controller design are provided.

10.4.1. Discrete LRPs

Theorem 10.1 defines the necessary and sufficient conditions for stability along the pass; however, it has to be outlined that the practical applicability of those conditions is really small. To provide the condition which could be used in practice (and in the sequel synthesis), define the following matrices from the state-space model (10.1)–(10.2):

$$\widehat{A}_1 = \begin{bmatrix} A & B_0 \\ 0 & 0 \end{bmatrix}, \quad \widehat{A}_2 = \begin{bmatrix} 0 & 0 \\ C & D_0 \end{bmatrix}. \tag{10.15}$$

Then it is possible to present the following result:

Theorem 10.4. (Gałkowski *et al.*, 2002) *The discrete LRP described by (10.1)–(10.2) is stable along the pass if there exist the matrices $P \succ 0$ and $Q \succ 0$ satisfying the following LMI:*

$$\begin{bmatrix} \widehat{A}_1^T P \widehat{A}_1 + Q - P & \widehat{A}_1^T P \widehat{A}_2 \\ \widehat{A}_2^T P \widehat{A}_1 & \widehat{A}_2^T P \widehat{A}_2 - Q \end{bmatrix} \prec 0. \tag{10.16}$$

Remark 10.2. *Note that the result of Theorem 10.4 provides only the sufficient condition for stability along the pass. Recently, there was published a paper by Bliman*

(Bliman, 2002) regarding a decrease in the conservativeness of the given LMI condition and getting "closer" to the sufficient and necessary condition for the stability of 2D systems. Due to the inherent 2D structure of the LRP, it can be applied to LRPs as well. This new approach relies on sequentially increasing the state vector by the next delayed state vectors till a feasible solution of an appropriate LMI condition is found or the certificate of infeasibility is given. This approach is not considered here due to the fact that it has limited applicability in terms of controller design.

The LMI condition for stability along the pass of discrete LRPs equivalent to (10.16) can be stated as follows:

Theorem 10.5. (Gałkowski *et al.*, 2002) *The discrete LRP described by (10.1)–(10.2) is stable along the pass if there exist the matrices* $Y \succ 0$ *and* $Z \succ 0$ *satisfying the following LMI:*

$$\begin{bmatrix} Z-Y & 0 & Y\widehat{A}_1^T \\ 0 & -Z & Y\widehat{A}_2^T \\ \widehat{A}_1 Y & \widehat{A}_2 Y & -Y \end{bmatrix} \prec 0.$$

Another known approach to stability along the pass investigation is based on using block diagonal decision matrices of appropriate dimensions. The following theorem can be stated:

Theorem 10.6. (Gałkowski *et al.*, 2002) *The discrete LRP described by (10.1)–(10.2) is stable along the pass if there exists the matrix* $W = \operatorname{diag}(W_1, W_2) \succ 0$ *satisfying the following LMI:*

$$\Upsilon^T W \Upsilon - W \prec 0,$$

where Υ *is the so-called plant matrix and was defined in (10.10).*

The synthesis is done similarly as for asymptotic stability. First, define a control law of the following form over $0 \le p \le \alpha - 1$, $k \ge 0$ (Gałkowski *et al.*, 2002):

$$u_{k+1}(p) = K_1 x_{k+1}(p) + K_2 y_k(p) := K \begin{bmatrix} x_{k+1}(p) \\ y_k(p) \end{bmatrix}, \tag{10.17}$$

where K_1 and K_2 are appropriately dimensioned controller matrices to be found. In effect, this control law uses the feedback of the current trial state vector (which is assumed to be available for use), and the "feedforward" of the previous trial pass profile vector. Note that in repetitive processes the term "feedforward" is used to describe the case where (state or pass profile) information from the previous pass (or passes) is used as (part of) the input to a control law applied on the current pass, i.e., to information which is propagated in the pass to pass (k) direction.

This control law has physical meaning for practical applications of discrete LRPs, and the following result uses the LMI setting to give a controller design algorithm which can be easily implemented:

Theorem 10.7. (Gałkowski *et al.*, 2002) *Suppose that the discrete LRP of (10.1)–(10.2) is subject to a control law of the form (10.17). Then the closed loop system is*

stable along the pass if there exist the matrices $Y \succ 0$, $Z \succ 0$, *and* N *such that the following LMI holds:*

$$\begin{bmatrix} Z-Y & 0 & Y\widehat{A}_1^T + N^T\widehat{B}_1^T \\ 0 & -Z & Y\widehat{A}_2^T + N^T\widehat{B}_2^T \\ \widehat{A}_1 Y + \widehat{B}_1 N & \widehat{A}_2 Y + \widehat{B}_2 N & -Y \end{bmatrix} \prec 0, \tag{10.18}$$

where $\widehat{A}_1$, $\widehat{A}_2$ *are defined as in (10.15), and*

$$\widehat{B}_1 = \begin{bmatrix} B \\ 0 \end{bmatrix}, \quad \widehat{B}_2 = \begin{bmatrix} 0 \\ D \end{bmatrix}. \tag{10.19}$$

If (10.18) holds, then stabilizing K *in the control law (10.17) is given by*

$$K = NY^{-1}. \tag{10.20}$$

10.4.2. Differential LRPs

A similar procedure as for discrete LRPs can be performed also for the differential case of (10.6)–(10.7). Here only the most often applied condition for stability along the pass in terms of an LMI is given.

Theorem 10.8. (Gałkowski *et al.*, 2003b) *A differential LRP is stable along the pass if there exist the matrices* $Y \succ 0$ *and* $Z \succ 0$ *satisfying the following LMI:*

$$\begin{bmatrix} YA^T + AY & B_0 Z & YC^T \\ ZB_0^T & -Z & ZD_0^T \\ CY & D_0 Z & -Z \end{bmatrix} \prec 0. \tag{10.21}$$

An LMI condition similar to that given in Theorem 10.7 can be provided here for controller design for the differential LRP (10.6)–(10.7). First, define the control law over $0 \le t < \alpha$, $k \ge 0$:

$$u_{k+1}(t) = K_1 x_{k+1}(t) + K_2 y_k(t), \tag{10.22}$$

where again K_1 and K_2 are appropriately dimensioned controller matrices to be designed. Now, using the LMI stability test and the defined feedback loop (10.22), the following result can be presented:

Theorem 10.9. (Gałkowski *et al.*, 2003b) *A differential LRP is stable along the pass under the control law of (10.22) if there exist the matrices* $Y \succ 0$, $Z \succ 0$, M *and* N *of appropriate dimensions such that the following LMI holds:*

$$\begin{bmatrix} YA^T + AY + N^T B^T + BN & B_0 Z + BM & YC^T + N^T D^T \\ ZB_0^T + M^T B^T & -Z & ZD_0^T + M^T D^T \\ CY + DN & D_0 Z + DM & -Z \end{bmatrix} \prec 0. \tag{10.23}$$

Then the controllers K_1 *and* K_2 *are given by*

$$K_1 = NY^{-1}, \quad K_2 = MZ^{-1}. \tag{10.24}$$

10.5. Robustness analysis

In general, models available for design will only be an approximation to process dynamics. Hence we also deal here with robust control where, as in the 1D linear systems case, we assume that unmodelled dynamics lie within well-defined model classes (or assumptions). Here we consider the following two:

(a) *Norm-bounded uncertainty model*. This model of uncertainty corresponds to a system whose matrices uncertainty are modelled as an additive perturbation to the nominal system matrices. Therefore a system is said to be subjected to norm-bounded parameter uncertainty if the matrices of such a system can be written in the form

$$M = M_0 + \Delta M = M_0 + H\mathcal{F}E, \tag{10.25}$$

where H and E are some known constant matrices with compatible dimensions and M_0 defines the nominal system. $\mathcal{F}$ is an unknown, constant matrix which satisfies

$$\mathcal{F}^T\mathcal{F} \preceq I. \tag{10.26}$$

Note that the above model of uncertainty has been widely adopted in describing parametric uncertainty of 1D uncertain systems (see Khargonekar *et al.*, 1990 and the references therein).

(b) *Polytopic uncertainty model*. This model of uncertainty corresponds to a system whose matrices lie in the polytope of matrices. This means that each system matrix M is only known to lie in a given fix polytope of matrices described by

$$M \in \mathrm{Co}\,(M_1, M_2, \ldots, M_h)\,, \tag{10.27}$$

where Co denotes the convex hull. In particular, for positive $i = 1, 2, \ldots, h$, M can be written as

$$M := \left\{ X : X = \sum_{i=1}^{h} \alpha_i M_i, \quad \alpha_i \geq 0,\ \sum_{i=1}^{h} \alpha_i = 1 \right\}.$$

Uncertainties satisfying any of these are termed to be admissible.

Based on the state space model of differential LRPs, a robust stabilization (using an appropriately specified control law) condition is provided for solving them in terms of the feasibility of some LMIs.

In the case of matrices of Eqn. (10.6), which are considered here, it is assumed that the uncertainty of the differential fraction of an uncertain differential LRP has a polytopic character, i.e., all possible choices for matrices which define the current pass state dynamics can be expressed as

$$\begin{bmatrix} A & B_0 & B \end{bmatrix} \in \mathrm{Co}\left(\begin{bmatrix} A^i & B_0^i & B^i \end{bmatrix} \right), \quad i = 1, 2, \ldots, h\,, \tag{10.28}$$

where

$$\mathrm{Co}\left(\begin{bmatrix} A^i & B^i & B_0^i \end{bmatrix} \right) := \left\{ X : X = \sum_{i=1}^{h} \alpha_i \begin{bmatrix} A^i & B^i & B_0^i \end{bmatrix},\ \alpha_i \geq 0,\ \sum_{i=1}^{h} \alpha_i = 1 \right\}. \tag{10.29}$$

For the current pass profile updating equation we assume a norm-bounded type of uncertainty, i.e.,

$$y_{k+1}(t) = (C + \Delta C)x_{k+1}(t) + (D_0 + \Delta D_0)y_k(t) + (D + \Delta D)u_{k+1}(t),$$

where

$$\begin{bmatrix} \Delta C & \Delta D_0 & \Delta D \end{bmatrix} = H_2 \mathcal{F} \begin{bmatrix} E_1 & E_2 & E_3 \end{bmatrix}, \tag{10.30}$$

and the matrix $\mathcal{F}$ satisfies (10.26). Then the following result holds in the case when a control law of the form (10.22) is applied:

Theorem 10.10. *Suppose that a differential LRP with the uncertainty structure modeled by (10.28)–(10.29) and (10.30) is subjected to a control law of the form of (10.22). Then the resulting closed loop process is stable along the pass for all admissible uncertainties if there exist the matrices $W_1 \succ 0$, $W_2 \succ 0$, N_1 and N_2 of compatible dimensions and a scalar $\epsilon > 0$ such that*

$$\begin{bmatrix} -W_2 + 2\epsilon H_2 H_2^T & (\star) & (\star) & (\star) & (\star) \\ W_1 C^T + N_1^T D^T & W_1 A^{iT} + N_1^T B^{iT} + A^i W_1 + B^i N_1 & (\star) & (\star) & (\star) \\ W_2 D_0^T + N_2^T D^T & W_2 B_0^{iT} + N_2^T B^{iT} & -W_2 & (\star) & (\star) \\ 0 & E_1 W_1 + E_3 N_1 & 0 & -\epsilon I & (\star) \\ 0 & 0 & E_2 W_2 + E_3 N_2 & 0 & -\epsilon I \end{bmatrix} \prec 0.$$

If the above LMI holds, then the controller matrices K_1 and K_2 are given by

$$K_1 = N_1 W_1^{-1}, \quad K_2 = N_2 W_2^{-1}, \tag{10.31}$$

respectively.

Some comments are required for the above result. First, it should be emphasized that the result has been obtained by keeping W_1 constant and independent of the index i. The main drawback associated with this fact is that the Lyapunov matrix W_1 must work for all uncertain matrices (10.29). This condition can introduce a significant degree of conservativeness to the above LMI condition. However, this can be overcome by using parameter-dependent Lyapunov functions.

For discrete processes solutions to the synthesis problem for uncertain LRPs can be found, e.g., in (Paszke, 2005).

10.6. Guaranteed cost control

Many applications will require a controller which not only guarantees stability along the pass but also meets specified performance criteria.

We start by developing the LMI condition which guarantees that the unforced (the control input terms are deleted) process is stable along the pass and also the associated cost function is bounded for all admissible uncertainties. These results are then extended to design a guaranteed cost controller.

It is assumed that the following cost function is associated with the uncertain process with the uncertainty structure defined by (10.26):

$$J=\sum_{k=0}^{\infty}\int_0^{\infty}\left(u_{k+1}^T(t)\Psi u_{k+1}(t)\right)\mathrm{d}t+\sum_{k=0}^{\infty}\int_0^{\infty}\left(\begin{bmatrix}x_{k+1}(t)\\ y_k(t)\end{bmatrix}^T\begin{bmatrix}Q_1 & 0\\ 0 & Q_2\end{bmatrix}\begin{bmatrix}x_{k+1}(t)\\ y_k(t)\end{bmatrix}\right)\mathrm{d}t, \tag{10.32}$$

where $\Psi \succ 0$, $Q_1 \succ 0$ and $Q_2 \succ 0$ are given matrices, is bounded for all admissible uncertainties.

Remark 10.3. *LRPs are defined over the finite pass length α and, in practice, only a finite number of passes, say k^*, will actually be completed. However, it is routine to argue that the signals involved can be extended from $[0,\alpha]$ to the infinite interval in such a way that the projection of the infinite interval solution onto the finite interval is possible. The same is true for the pass-to-pass direction and hence we can work with (10.32).*

10.6.1. Guaranteed cost bound

Here we are interested in finding an upper bound for the corresponding cost function of the unforced process ($u_{k+1}(t) = 0$), hence the first term in (10.32) is removed. The following theorem gives a sufficient condition for stability along the pass with a guaranteed cost.

Theorem 10.11. *An unforced differential LRP is robustly stable if there exist the matrices $P_1 \succ 0$, $P_2 \succ 0$ and a scalar $\epsilon > 0$ such that the following LMI holds:*

$$\begin{bmatrix} -P_2 & P_2C & P_2D_0 & P_2H_2 & P_2H_2\\ C^TP_2 & A^TP_1+P_1A+Q_1+\epsilon E_1^TE_1 & P_1B_0 & P_1H_1 & P_1H_1\\ D_0^TP_2 & B_0^TP_1 & -P_2+Q_2+\epsilon E_2^TE_2 & 0 & 0\\ H_2^TP_2 & H_1^TP_1 & 0 & -\epsilon I & 0\\ H_2^TP_2 & H_1^TP_1 & 0 & 0 & -\epsilon I \end{bmatrix} \prec 0. \tag{10.33}$$

Moreover, in this case the cost function satisfies the following upper bound:

$$J_0 \le \sum_{k=0}^{k^*} x_{k+1}^T(0)P_1x_{k+1}(0) + \int_0^{\alpha} y_0^T(t)P_2y_0(t)\,\mathrm{d}t. \tag{10.34}$$

10.6.2. Guaranteed cost control with a static feedback controller

Applying the control law (10.22) gives the closed loop process state-space model of the form

$$\begin{bmatrix}\dot{x}_{k+1}(t)\\ y_{k+1}(t)\end{bmatrix} = \left(\begin{bmatrix}A+BK_1 & B_0+BK_2\\ C+DK_1 & D_0+DK_2\end{bmatrix}+\begin{bmatrix}\Delta A+\Delta BK_1 & \Delta B_0+\Delta BK_2\\ \Delta C+\Delta DK_1 & \Delta D_0+\Delta DK_2\end{bmatrix}\right)\begin{bmatrix}x_{k+1}(t)\\ y_k(t)\end{bmatrix}, \tag{10.35}$$

and the associated cost function is

$$J=\sum_{k=0}^{\infty}\int_0^{\infty}\left(\begin{bmatrix}x_{k+1}(t)\\ y_k(t)\end{bmatrix}^T\begin{bmatrix}Q_1+K_1^T\Psi K_1 & K_1^T\Psi K_2\\ K_2^T\Psi K_1 & Q_2+K_2^T\Psi K_2\end{bmatrix}\begin{bmatrix}x_{k+1}(t)\\ y_k(t)\end{bmatrix}\right)\mathrm{d}t, \tag{10.36}$$

where $\Psi\succ 0$, $Q_1\succ 0$ and $Q_2\succ 0$ are given matrices.

The existence of stabilizing K_1 and K_2 can be characterized in LMI terms as follows:

Theorem 10.12. *A differential LRP is robustly stable under the control law (10.22) if there exist the matrices $W_1\succ 0$, $W_2\succ 0$, N_1 and N_2 and a scalar $\epsilon>0$ such that the following LMI holds:*

$$\begin{bmatrix}
-W_2+2\epsilon H_2H_2^T & CW_1+DN_1+2\epsilon H_2H_1^T\\
W_1C^T+N_1^TD^T+2\epsilon H_1H_2^T & W_1A^T+AW_1+N_1^TB^T+BN_1+2\epsilon H_1H_1^T\\
W_2D_0^T+N_2^TD^T & W_2B_0^T+N_2^TB^T\\
0 & E_1W_1+E_3N_1\\
0 & 0\\
0 & N_1\\
0 & W_1\\
0 & 0
\end{bmatrix}$$

$$\begin{bmatrix}
D_0W_2+DN_2 & 0 & 0 & 0 & 0 & 0\\
B_0W_2+BN_2 & W_1E_1^T+N_1^TE_3^T & 0 & N_1^T & W_1 & 0\\
-W_2 & 0 & W_2E_2^T+N_2E_3^T & N_2^T & 0 & W_2\\
0 & -\epsilon I & 0 & 0 & 0 & 0\\
E_2W_2+E_3N_2 & 0 & -\epsilon I & 0 & 0 & 0\\
N_2 & 0 & 0 & -\Psi^{-1} & 0 & 0\\
0 & 0 & 0 & 0 & -Q_1^{-1} & 0\\
W_2 & 0 & 0 & 0 & 0 & -Q_2^{-1}
\end{bmatrix}\prec 0, \tag{10.37}$$

where $\Psi\succ 0$, $Q_1\succ 0$ and $Q_2\succ 0$ are given matrices for the cost function (10.32). Also, if this condition holds, then the stabilizing control law matrices K_1, K_2 are given by (10.31). The cost function (10.36) of the closed-loop process (10.35) satisfies the following upper bound:

$$J\le\sum_{k=0}^{k^*}x_{k+1}^T(0)W_1^{-1}x_{k+1}(0)+\int_0^{\alpha}y_0^T(t)W_2^{-1}y_0(t)\,\mathrm{d}t. \tag{10.38}$$

Note that it is possible to minimize the upper bound on the cost function (10.34) using the following optimization procedure:

$$\min_{W_1\succ 0,W_2\succ 0}\left[\sum_{k=0}^{k^*}x_{k+1}^T(0)W_1^{-1}x_{k+1}(0)+\int_0^{\alpha}y_0^T(t)W_2^{-1}y_0(t)\,\mathrm{d}t\right], \tag{10.39}$$

subject to (10.37).

The convex optimization algorithm cannot be applied in this case because of the non-linear terms W_1^{-1} and W_2^{-1}. However, a controller which ensures the minimization of the guaranteed cost (10.38) can be achieved as follows: First note that, from the fact that $\operatorname{trace}(XY) = \operatorname{trace}(YX)$, we have

$$\sum_{k=0}^{k^*} x_{k+1}^T(0) W_1^{-1} x_{k+1}(0) = \sum_{k=0}^{k^*} \operatorname{trace}\left(W_1^{-1} x_{k+1}(0) x_{k+1}^T(0)\right),$$

and

$$\int_0^\alpha y_0^T(t) W_2^{-1} y_0(t)\,\mathrm{d}t = \int_0^\alpha \operatorname{trace}\left(W_2^{-1} y_0(t) y_0^T(t)\right)\mathrm{d}t.$$

Next, recall that if a matrix M is symmetric and positive semi-definite, i.e., $M \succeq 0$, then eigenvalue decomposition of such a matrix gives

$$M = V \Theta V^T,$$

where V is some unitary matrix and Θ is a diagonal with non-negative diagonal entries. Therefore, the matrix square root of M can be defined as $M^{\frac{1}{2}} = V \Theta^{\frac{1}{2}} V^T$ and computed. Based on this, the matrices $\Upsilon^{\frac{1}{2}}$ and $\Sigma^{\frac{1}{2}}$

$$\Upsilon = \Upsilon^{\frac{1}{2}} \Upsilon^{\frac{1}{2}} = \sum_{k=0}^{k^*} x_{k+1}(0) x_{k+1}^T(0), \quad \Sigma = \Sigma^{\frac{1}{2}} \Sigma^{\frac{1}{2}} = \int_0^\alpha y_0(t) y_0^T(t)\,\mathrm{d}t,$$

can be obtained. The dimensions of $\Sigma^{\frac{1}{2}}$ and $\Upsilon^{\frac{1}{2}}$ are $n \times n$ and $m \times m$, respectively. Furthermore, introduce the symmetric matrices Ξ, Ω which satisfy

$$\operatorname{trace}(\Upsilon^{\frac{1}{2}} W_1^{-1} \Upsilon^{\frac{1}{2}}) < \operatorname{trace}(\Xi), \quad \operatorname{trace}(\Sigma^{\frac{1}{2}} W_2^{-1} \Sigma^{\frac{1}{2}}) < \operatorname{trace}(\Omega).$$

Hence we can write

$$\Upsilon^{\frac{1}{2}} W_1^{-1} \Upsilon^{\frac{1}{2}} \prec \Xi, \quad \Sigma^{\frac{1}{2}} W_2^{-1} \Sigma^{\frac{1}{2}} \prec \Omega. \tag{10.40}$$

Carrying out an obvious application of the Schur complement of (10.40) yields

$$\begin{bmatrix} -\Xi & \Upsilon^{\frac{1}{2}} \\ \Upsilon^{\frac{1}{2}} & -W_1 \end{bmatrix} \prec 0 \quad \text{and} \quad \begin{bmatrix} -\Omega & \Sigma^{\frac{1}{2}} \\ \Sigma^{\frac{1}{2}} & -W_2 \end{bmatrix} \prec 0, \tag{10.41}$$

respectively. Finally, the subsequent minimization problem can be formulated as

$$\begin{gathered} \min_{W_1 \succ 0, W_2 \succ 0, N_1, N_2} \quad (\operatorname{trace}(\Xi) + \operatorname{trace}(\Omega)), \\ \text{subject to (10.37) and (10.41)}, \end{gathered} \tag{10.42}$$

and the solution (10.31) now guarantees that the cost function is minimized over the finite pass length in the case when only a finite number of trials is actually completed. Since the minimization problem of (10.42) is the convex optimization problem, then it is simple to implement using a computer and computationally effective.

Analogical results for discrete LRPs can be found in (Paszke, 2005).

10.7. $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control

We start this section with the following signal space definition that will be extensively used during our further deliberations:

Definition 10.2. Consider a $q \times 1$ vector sequence $\{w_j(t)\}$ defined over the real interval $0 \le t \le \infty$ and the non-negative integers $0 \le j \le \infty$, which is written as $\{[0,\infty],[0,\infty]\}$ Then the $\mathcal{L}_2$ norm of this vector sequence is given by

$$\|w\|_2 = \sqrt{\sum_{j=0}^{\infty} \int_0^{\infty} w_j^T(t) w_j(t)\,\mathrm{d}t},$$

and this sequence is said to be a member of $\mathcal{L}_2^q\{[0,\infty],[0,\infty]\}$, or $\mathcal{L}_2^q$ for short, if $\|w\|_2 < \infty$.

This section addresses this key problem area in an $\mathcal{H}_2$, $\mathcal{H}_\infty$ and mixed $\mathcal{H}_2/\mathcal{H}_\infty$ framework starting from the following process state-space model over $0 \le t \le \alpha$, $k \ge 0$:

$$\begin{aligned} \dot{x}_{k+1}(t) &= Ax_{k+1}(t) + B_0 y_k(t) + Bu_{k+1}(t) + B_{11}w_{k+1}(t) + B_{21}\nu_{k+1}(t), \\ y_{k+1}(t) &= Cx_{k+1}(t) + D_0 y_k(t) + Du_{k+1}(t) + B_{12}w_{k+1}(t) + B_{22}\nu_{k+1}(t), \end{aligned} \tag{10.43}$$

where the vectors $x_{k+1}(t)$, $y_k(t)$ and $u_{k+1}(t)$ are defined as in (10.6)–(10.7), $w_{k+1}(t)$ and $\nu_{k+1}(t)$ are disturbance vectors which are taken as belonging to $\mathcal{L}_2$. (The boundary conditions are as per the disturbance free case.) Also, it is easy to conclude that stability along the pass of such a process is again governed by (10.21). Moreover, we use the induced $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms to measure the performance objective which is the attenuation of the effects w_{k+1} and ν_{k+1}. These are introduced next.

Definition 10.3. A differential linear repetitive process described by (10.43) is said to have the $\mathcal{H}_\infty$ disturbance attenuation (or the $\mathcal{H}_\infty$ norm bound) γ_∞ if it is stable along the pass and

$$\sup_{0 \ne \nu \in \mathcal{L}_2^r} \frac{\|y\|_2}{\|\nu\|_2} < \gamma_\infty. \tag{10.44}$$

In effect, this is a worst case bound as it corresponds to a bound on the maximum peak gain of the 2D frequency response between ν and y, and is given, with $\overline{\sigma}(\cdot)$ denoting the maximum singular value of its matrix argument, by

$$\|G_{y\nu}(s,z)\|_\infty = \sup_{\omega_1 \in \mathbb{R}, \omega_2 \in [0,2\pi]} \overline{\sigma}\left[G(j\omega_1, e^{j\omega_2})\right],$$

where

$$G_{y\nu}(s,z) = \begin{bmatrix} 0 & I \end{bmatrix} \left(\begin{bmatrix} sI - A & -B_0 \\ -zC & I - zD_0 \end{bmatrix} \right)^{-1} \begin{bmatrix} B_{21} \\ B_{22} \end{bmatrix}, \tag{10.45}$$

i.e., as the 2D transfer-function matrix between these two vectors.

To introduce the $\mathcal{H}_2$ norm, consider the 2D transfer-function matrix between w_{k+1} and y_{k+1}, i.e.,

$$G_{yw}(s,z) = \left[\begin{array}{cc} 0 & I \end{array}\right] \left(\left[\begin{array}{cc} sI - A & -B_0 \\ -zC & I - zD_0 \end{array}\right] \right)^{-1} \left[\begin{array}{c} B_{11} \\ B_{12} \end{array}\right]. \tag{10.46}$$

Then the $\mathcal{H}_2$ norm is defined as the square of the $\mathcal{L}_2$ norm of $w_{k+1}(t)$ and, by use of Parseval's theorem in the 2D signal case, it is given by

$$\|G_{yw}(s,z)\|_2 = \sqrt{\frac{1}{(2\pi)^2} \int_0^{2\pi} \int_{-\infty}^{\infty} \Theta \, \mathrm{d}\omega_2 \, \mathrm{d}\omega_1}, \tag{10.47}$$

where

$$\Theta = \operatorname{trace}\left(G^*(-j\omega_2, e^{j\omega_1}) G(-j\omega_2, e^{j\omega_1})\right)$$

and $G^*(\cdot)$ denotes the complex conjugate transpose of $G(\cdot)$. This is commonly termed the total energy in the signal (and an alternative formula for computing it will be given in a proper context later in this section).

10.7.1. $\mathcal{H}_\infty$ norm

We will first state the following known result on the formulation of the $\mathcal{H}_\infty$ norm as an LMI constraint.

Lemma 10.1. (Paszke *et al.*, 2004) *A differential linear repetitive process described by (10.43) is stable along the pass and has the $\mathcal{H}_\infty$ disturbance attenuation $\gamma_\infty > 0$ if there exist the matrices $R_1 \succ 0$, $R_2 \succ 0$ and $R_3 \succ 0$ such that*

$$\left[\begin{array}{cccc} -S & S\widehat{A}_2 & S\widehat{D}_1 & 0 \\ \widehat{A}_2^T S & \widehat{A}_1^T P + P\widehat{A}_1 - R & P\widehat{B}_1 & L^T \\ \widehat{D}_1^T S & \widehat{B}_1^T P & -\gamma_\infty^2 I & 0 \\ 0 & L & 0 & -I \end{array}\right] \prec 0, \tag{10.48}$$

where $P = \operatorname{diag}(R_1, 0)$, $S = \operatorname{diag}(R_3, R_2)$, $R = \operatorname{diag}(0, R_2)$ and

$$\widehat{B}_1 = \left[\begin{array}{c} B_{21} \\ 0 \end{array}\right], \quad \widehat{D}_1 = \left[\begin{array}{c} 0 \\ B_{22} \end{array}\right], \quad L = \left[\begin{array}{cc} 0 & I \end{array}\right]. \tag{10.49}$$

This result is the so-called bounded real lemma for differential linear repetitive processes. Note also that the matrix R_3 has no influence on the result (but is needed in its proof). Hence it can be deleted to give the following result:

Lemma 10.2. (Paszke *et al.*, 2004) *For some prescribed $\gamma_\infty > 0$, suppose that there exist the matrices $R_1 \succ 0$, and $R_2 \succ 0$ such that the following LMI holds for a differential linear repetitive process described by (10.43):*

$$\left[\begin{array}{cccc} -R_2 & R_2C & R_2D_0 & R_2B_{22} \\ C^T R_2 & A^T R_1 + R_1 A & R_1 B_0 & R_1 B_{21} \\ D_0^T R_2 & B_0^T R_1 & -R_2 + I & 0 \\ B_{22}^T R_2 & B_{21}^T R_1 & 0 & -\gamma_\infty^2 I \end{array}\right] \prec 0. \tag{10.50}$$

Then this process is stable along the pass and also $\|G_{y\nu}(s,z)\|_\infty < \gamma_\infty$.

Motivated by 1D system theory, where the $\mathcal{H}_\infty$ norm is used as a measure of system robustness, the above result has the following interpretation: Keeping the $\mathcal{H}_\infty$ norm of the controlled process 2D transfer-function matrix from ν to y below the level γ_∞ guarantees that the process under consideration is robust to unstructured perturbations of the form

$$\nu = \mathbf{\Delta} y, \quad \|\mathbf{\Delta}\|_\infty \leq \gamma_\infty.$$

This means that choosing a lower value of γ_∞ reduces robustness to unmodelled dynamics (as measured in this way) and vice-versa.

10.7.2. Static $\mathcal{H}_\infty$ controller

Here we study the solution to the problem of the $\mathcal{H}_\infty$ disturbance attenuation in the case of full state access, i.e., the case when the control law of the form (10.22) is applied. Under these assumptions, we have the following result:

Theorem 10.13. (Paszke, 2005) *Suppose that a differential LRP described by (10.43) is subject to a control law defined by (10.22). Then the resulting closed loop process is stable along the pass and has the prescribed $\mathcal{H}_\infty$ disturbance attenuation $\gamma_\infty > 0$ if there exist the matrices $W_1 \succ 0$, $W_2 \succ 0$, N_1 and N_2 of compatible dimensions such that the following LMI holds:*

$$\begin{bmatrix} -W_2 & CW_1+DN_1 & D_0W_2+DN_2 & B_{22} & 0 \\ W_1C^T+N_1^TD^T & W_1A^T+N_1^TB^T+AW_1+BN_1 & B_0W_2+BN_2 & B_{21} & 0 \\ W_2D_0^T+N_2^TD^T & W_2B_0^T+N_2^TB^T & -W_2 & 0 & W_2 \\ B_{22}^T & B_{21}^T & 0 & -\gamma_\infty^2 I & 0 \\ 0 & 0 & W_2 & 0 & -I \end{bmatrix} \prec 0. \tag{10.51}$$

If this condition holds, the $\mathcal{H}_\infty$ controller matrices K_1 and K_2 are given by (10.31).

10.7.3. $\mathcal{H}_2$ norm

In the 1D linear systems case, the $\mathcal{H}_2$ norm coincides with the total output energy in the impulse response. Moreover, this observation leads immediately to algorithms for computing this norm from the state-space model. Next we develop an LMI (and hence state-space-based) method for computing $\mathcal{H}_2$ for the differential linear repetitive processes considered here.

Consider first a single input stable along the pass process (note again that this property can be analysed mathematically by letting the pass length $\alpha \to \infty$) represented by (10.6)–(10.7), and let the $m \times 1$ vector $g_k(t)$ denote the response to an impulse, denoted by $\delta_k(t)$, applied at $t = 0$ on the pass k. Then, by invoking Parseval's theorem in the along pass direction on each pass and summing over the pass index, the $\mathcal{H}_2$ norm is given by

$$\|G\|_2 = \sqrt{\|g_{k+1}(t)\|_2^2} = \sqrt{\sum_{k=0}^{\infty} \int_0^\infty g_{k+1}^T(t) g_{k+1}(t)\, \mathrm{d}t}. \tag{10.52}$$

To extend this definition to vector-valued inputs, introduce

$$u_k^h(t) = \delta_k(t)e^h, \tag{10.53}$$

where e^h is the $l \times 1$ vector whose entries are zero except for a unit entry in the position $h, 1 \leq h \leq l$. Then

$$\|G\|_2 = \sqrt{\sum_{h=1}^{l}\sum_{k=0}^{\infty}\int_0^{\infty}(g_{k+1}^h)^T(t)g_{k+1}^h(t)\,\mathrm{d}t}. \tag{10.54}$$

The following result gives a sufficient condition for stability along the pass together with an upper bound on the $\mathcal{H}_2$ norm of the 2D transfer-function matrix (between the disturbance w and the pass profile y).

Theorem 10.14. (Paszke, 2005) *A differential linear repetitive process described by (10.43) is stable along the pass and has the $\mathcal{H}_2$ norm bound $\gamma_2 > 0$, i.e., $\|G_{yw}(s,z)\|_2 < \gamma_2$, if there exist the matrices $P_1 \succ 0$ and $P_2 \succ 0$ such that the following LMIs hold:*

$$\begin{bmatrix} -P_2 & P_2C & P_2D_0 \\ C^TP_2 & A^TP_1+P_1A+C^TC & P_1B_0+C^TD_0 \\ D_0^TP_2 & B_0^TP_1+D_0^TC & -P_2+D_0^TD_0 \end{bmatrix} \prec 0 \tag{10.55}$$

and

$$\operatorname{trace}(D^TD + B^TP_1B + D^TP_2D) - \gamma_2^2 < 0. \tag{10.56}$$

Remark 10.4. *The $\mathcal{H}_2$ norm bound here can be minimized using the following linear objective minimization algorithm:*

$$\begin{aligned} &\min_{P_1\succ 0, P_2\succ 0, \mu>0} \quad \mu, \\ &\text{subject to (10.55) and (10.56) with } \mu = \gamma_2^2. \end{aligned} \tag{10.57}$$

10.7.4. Static $\mathcal{H}_2$ controller

In this subsection we solve the static control law design problem to give stability along the pass plus a given level of disturbance attenuation as measured by the $\mathcal{H}_2$ norm. Hence we can consider the process state-space model (10.43) with the disturbance input vector ν deleted.

The application of the control law (10.22) results in the controlled process state-space model

$$\begin{aligned} \dot{x}_{k+1}(t) &= (A + BK_1)x_{k+1}(t) + (B_0 + BK_2)y_k(t) + B_{11}w_{k+1}(t), \\ y_{k+1}(t) &= (C + DK_1)x_{k+1}(t) + (D_0 + DK_2)y_k(t) + B_{12}w_{k+1}(t), \end{aligned} \tag{10.58}$$

and hence the 2D transfer-function matrix between the disturbance vector and the current pass profile is given by

$$G_{yw}^{cl}(s,z) = \begin{bmatrix} 0 & I \end{bmatrix} \left(\begin{bmatrix} sI-(A+BK_1) & -(B_0+BK_2) \\ -z(C+DK_1) & I - z(D_0+DK_2) \end{bmatrix} \right)^{-1} \begin{bmatrix} B_{11} \\ B_{12} \end{bmatrix}.$$

The following result gives a solution to this problem with an algorithm for designing the control law.

Theorem 10.15. (Paszke, 2005) *Suppose that a control law of the form (10.22) is applied to a differential linear repetitive process described by (10.43) (with the disturbance vector ν deleted). Then the resulting controlled process is stable along the pass and has the prescribed $\mathcal{H}_2$ disturbance attenuation bound $\gamma_2 > 0$ if there exist the matrices $W_1 \succ 0$, $W_2 \succ 0$, X, N_1 and N_2 such that the following LMIs hold:*

$$\begin{bmatrix} -W_2 & CW_1+DN_1 & D_0W_2+DN_2 & 0 \\ N_1^TD^T+W_1C^T & W_1A^T+AW_1+N_1^TB^T+BN_1 & B_0W_2+BN_2 & W_1C^T+N_1^TD^T \\ N_2^TD^T+W_2D_0^T & W_2B_0^T+N_2^TB^T & -W_2 & W_2D_0^T+N_2^TD^T \\ 0 & CW_1+DN_1 & D_0W_2+DN_2 & -I \end{bmatrix} \prec 0 \tag{10.59}$$

and

$$\begin{gathered} \operatorname{trace}(X) < \gamma_2^2 - \operatorname{trace}(B_{12}^TB_{12}), \\ \begin{bmatrix} X & B_{11}^T & B_{12}^T \\ B_{11} & W_1 & 0 \\ B_{12} & 0 & W_2 \end{bmatrix} \succ 0, \end{gathered} \tag{10.60}$$

where X is an additional symmetric matrix of compatible dimensions. If these conditions hold, the control law matrices K_1 and K_2 are given by (10.31).

10.7.5. Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control problem

In this section we address the question of when there exists a control law of the form (10.22) which, for processes described by (10.43), minimizes the $\mathcal{H}_2$ norm from w to y, denoted here by $\|G_{yw}^{cl}(s,z)\|_2$, and keeps the $\mathcal{H}_\infty$ norm from ν to y, denoted here by $\|G_{y\nu}^{cl}(s,z)\|_\infty$, below some prescribed level. Note also that if only w is present, then this problem reduces to the $\mathcal{H}_2$ control problem solved in (Paszke *et al.*, 2005). Similarly, if only ν is present, then we obtain the $\mathcal{H}_\infty$ control problem solved in (Paszke *et al.*, 2004).

Combining the results on the $\mathcal{H}_\infty$ and $\mathcal{H}_2$ control, we can provide the following result that gives the LMI condition for mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control law design:

Theorem 10.16. *Suppose that a control law of the form (10.22) is applied to a differential linear repetitive process described by (10.43). Then the resulting controlled process is stable along the pass and has the prescribed $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norms bounds $\gamma_2 > 0$ and $\gamma_\infty > 0$, respectively, if there exist the matrices $W_1 \succ 0$, $W_2 \succ 0$, X, N_1 and N_2 such that the LMIs (10.59), (10.60) and (10.51) hold. If this is the case, then the $\mathcal{H}_2/\mathcal{H}_\infty$ control law matrices K_1 and K_2 are given by (10.31).*

Remark 10.5. *The above result has been obtained by enforcing that the matrices $W_1 \succ 0$, $W_2 \succ 0$, N_1 and N_2 are the same in (10.59), (10.60) and (10.51). However, this assumption increases the level of conservativeness.*

It is important to note that by adjusting γ_∞ we can trade off between the $\mathcal{H}_\infty$ and $\mathcal{H}_2$ performance. Hence, a trade-off curve allows the designer to choose the controller that satisfies the compromise between robustness (measured with the $\mathcal{H}_\infty$ norm) and performance (measured with the $\mathcal{H}_2$ norm).

10.7.6. $\mathcal{H}_2/\mathcal{H}_\infty$ dynamic pass profile controller

The analysis performed so far in this chapter on control law design has assumed full access to the current pass state vector. If the current pass state vector is not available for measurement, then one option is to replace this term in the above control law by the current pass profile vector. This is feasible since, by definition, the pass profile is the process output vector (and hence available for measurement) and would only require that such measurement be not significantly corrupted by noise etc. However, it is possible that no static control law can be found and hence the next obvious option is to permit internal dynamics in the control law itself (at the (possible) cost of increased design complexity). Here we consider the case of the following so-called dynamic pass profile controller:

$$\begin{aligned}\begin{bmatrix} \dot{x}^c_{k+1}(t) \\ y^c_{k+1}(t) \end{bmatrix} &= \begin{bmatrix} A_{c11} & A_{c12} \\ A_{c21} & A_{c22} \end{bmatrix} \begin{bmatrix} x^c_{k+1}(t) \\ y^c_k(t) \end{bmatrix} + \begin{bmatrix} B_{c1} \\ B_{c2} \end{bmatrix} y_{k+1}(t), \\ u_{k+1}(t) &= \begin{bmatrix} C_{c1} & C_{c2} \end{bmatrix} \begin{bmatrix} x^c_{k+1}(t) \\ y^c_k(t) \end{bmatrix} + D_c y_{k+1}(t), \end{aligned} \tag{10.61}$$

where $x^c_{k+1}(t)$ and $y^c_k(t)$ are internal vectors for the controller.

Introduce now the following notation:

$$\begin{aligned} &\Phi = \begin{bmatrix} A & B_0 \\ C & D_0 \end{bmatrix}, \qquad \Omega_1 = \begin{bmatrix} B_{11} \\ B_{12} \end{bmatrix}, \quad \Omega_2 = \begin{bmatrix} B_{21} \\ B_{22} \end{bmatrix}, \quad B_2 = \begin{bmatrix} B \\ D \end{bmatrix}, \\ &C_2 = \begin{bmatrix} 0 & I \end{bmatrix}, \qquad C_c = \begin{bmatrix} C_{c1} & C_{c2} \end{bmatrix}, \\ &A_c = \begin{bmatrix} A_{c11} & A_{c12} \\ A_{c21} & A_{c22} \end{bmatrix}, \qquad B_c = \begin{bmatrix} B_{c1} \\ B_{c2} \end{bmatrix}, \end{aligned} \tag{10.62}$$

and also the so-called augmented state and pass profile vectors

$$\dot{\overline{x}}_{k+1}(t) = \begin{bmatrix} \dot{x}_{k+1}(t) \\ \dot{x}^c_{k+1}(t) \end{bmatrix}, \quad \overline{y}_k(t) = \begin{bmatrix} y_k(t) \\ y^c_k(t) \end{bmatrix}$$

and the matrices

$$\Pi = \begin{bmatrix} I\,0\,0\,0 \\ 0\,0\,I\,0 \\ 0\,I\,0\,0 \\ 0\,0\,0\,I \end{bmatrix}, \quad \Pi_1 = \begin{bmatrix} I\,0\,0\,0 \\ 0\,0\,I\,0 \\ 0\,0\,0\,0 \\ 0\,0\,0\,0 \end{bmatrix}, \quad \Pi_2 = \begin{bmatrix} 0\,0\,0\,0 \\ 0\,0\,0\,0 \\ 0\,I\,0\,0 \\ 0\,0\,0\,I \end{bmatrix}, \quad \Pi_3^T = \begin{bmatrix} 0 \\ I \\ 0 \\ 0 \end{bmatrix}.$$

Then, with the controller (10.61), the resulting controlled process state space model can be written in the form

$$\begin{aligned}\begin{bmatrix} \dot{x}_{k+1}(t) \\ \overline{y}_{k+1}(t) \end{bmatrix} &= \overline{A} \begin{bmatrix} \overline{x}_{k+1}(t), \\ \overline{y}_k(t) \end{bmatrix} + \overline{B}_1 w_{k+1}(t) + \overline{B}_2 w_{k+1}(t), \\ y_{k+1}(t) &= \overline{C} \begin{bmatrix} \overline{x}_{k+1}(t) \\ \overline{y}_{k+1}(t) \end{bmatrix},\end{aligned} \tag{10.63}$$

where

$$\begin{aligned}\overline{A} &= (\Pi_1 + \Pi_2) \begin{bmatrix} \Phi + B_2 D_c C_2 & B_2 C_c \\ B_c C_2 & A_c \end{bmatrix} \Pi^T \\ &= \Pi_1 \begin{bmatrix} \Phi + B_2 D_c C_2 & B_2 C_c \\ B_c C_2 & A_c \end{bmatrix} \Pi^T + \Pi_2 \begin{bmatrix} \Phi + B_2 D_c C_2 & B_2 C_c \\ B_c C_2 & A_c \end{bmatrix} \Pi^T \\ &= \overline{A}_1 + \overline{A}_2,\end{aligned}$$

$$\overline{B}_1 = \Pi_1 \begin{bmatrix} \Omega_1 \\ 0 \end{bmatrix} + \Pi_2 \begin{bmatrix} \Omega_1 \\ 0 \end{bmatrix} = \overline{B}_{11} + \overline{B}_{12},$$

$$\overline{B}_2 = \Pi_1 \begin{bmatrix} \Omega_2 \\ 0 \end{bmatrix} + \Pi_2 \begin{bmatrix} \Omega_2 \\ 0 \end{bmatrix} = \overline{B}_{21} + \overline{B}_{22},$$

$$\overline{C} = \begin{bmatrix} C_2 & 0 \end{bmatrix} \Pi^T.$$

Introduce the matrix of controller data as

$$\Theta = \begin{bmatrix} D_c & C_c \\ B_c & A_c \end{bmatrix} \tag{10.64}$$

and

$$\mathcal{A}_1 = \Pi_1 \begin{bmatrix} \Phi & 0 \\ 0 & 0 \end{bmatrix} \Pi^T, \quad \mathcal{A}_2 = \Pi_2 \begin{bmatrix} \Phi & 0 \\ 0 & 0 \end{bmatrix} \Pi^T, \quad \Gamma_1 = \Pi_1 \begin{bmatrix} B_2 & 0 \\ 0 & I \end{bmatrix},$$

$$\Gamma_2 = \Pi_2 \begin{bmatrix} B_2 & 0 \\ 0 & I \end{bmatrix}, \quad \mathcal{C}_2 = \begin{bmatrix} C_2 & 0 \\ 0 & I \end{bmatrix} \Pi^T$$

and hence we can write the matrices $\overline{A}_1, \overline{A}_2$ in the form (affine in the controller data matrix Θ):

$$\overline{A}_1 = \mathcal{A}_1 + \Gamma_1 \Theta \mathcal{C}_2, \quad \overline{A}_2 = \mathcal{A}_2 + \Gamma_2 \Theta \mathcal{C}_2.$$

Additionally, define $\overline{A}_3 = \mathcal{A}_3 + \Gamma_3 \Theta \mathcal{C}_2$, where

$$\mathcal{A}_3 = \Pi_3 \begin{bmatrix} \Phi & 0 \\ 0 & 0 \end{bmatrix} \Pi^T, \quad \Gamma_2 = \Pi_3 \begin{bmatrix} B_2 & 0 \\ 0 & I \end{bmatrix}.$$

Hence, based on Theorem 10.2, we have the following result expressed in terms of LMIs:

Theorem 10.17. *If there exist the matrices $P_{h_{11}} \succ 0$, $U_{h_{11}} \succ 0$, $S_{v_{11}} \succ 0$, $T_{v_{11}} \succ 0$ such that the LMIs defined by (10.65)–(10.70) hold, then there exists a controller of the form (10.61) which guarantees that a differential linear repetitive process described by (10.63) is stable along the pass and the prescribed $\mathcal{H}_2$ and $\mathcal{H}_\infty$ norm bounds $\gamma_2 > 0$ and $\gamma_\infty > 0$, respectively,*

$$\operatorname{trace}(B_{12}^T B_{12} + B_{11}^T P_{h_{11}} B_{11} + B_{12}^T S_{v_{11}} B_{12}) - \gamma_2^2 < 0, \tag{10.65}$$

$$\begin{bmatrix} \mathcal{W}_c^T & 0 \\ 0 & \mathcal{N}_D^T \end{bmatrix} \begin{bmatrix} -T_{v_{11}} + D_0^T T_{v_{11}} D_0 & CU_{h_{11}} + D_0^T T_{v_{11}} B_0 & D_0^T T_{v_{11}} D_0 \\ U_{h_{11}} C^T + B_0^T T_{v_{11}} D_0 & U_{h_{11}} A^T + A U_{h_{11}} + B_0^T T_{v_{11}} B_0 & U_{h_{11}} C^T + B_0^T T_{v_{11}} D_0 \\ D_0^T T_{v_{11}} D_0 & CU_{h_{11}} + D_0^T T_{v_{11}} B_0 & -I + D_0^T T_{v_{11}} D_0 \end{bmatrix} \times \begin{bmatrix} \mathcal{W}_c & 0 \\ 0 & \mathcal{N}_D \end{bmatrix} \prec 0, \tag{10.66}$$

$$\begin{bmatrix} A^T P_{h_{11}} + P_{h_{11}} A + C^T S_{v_{11}} C & C^T \\ C & -I \end{bmatrix} \prec 0, \tag{10.67}$$

$$\begin{bmatrix} \mathcal{W}_c^T & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} -T_{v_{11}} + D_0 T_{v_{11}} D_0^T & CU_{h_{11}} + D_0 T_{v_{11}} B_0^T & B_{22} & D_0 T_{v_{11}} \\ U_{h_{11}} C^T + B_0 T_{v_{11}} D_0^T & U_{h_{11}} A^T + A U_{h_{11}} + B_0 T_{v_{11}} B_0^T & B_{21} & B_0 T_{v_{11}} \\ B_{22}^T & B_{21}^T & -\gamma_\infty^2 I & 0 \\ T_{v_{11}} D_0^T & T_{v_{11}} B_0^T & 0 & -I + T_{v_{11}} \end{bmatrix} \begin{bmatrix} \mathcal{W}_c & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \prec 0, \tag{10.68}$$

$$\begin{bmatrix} A^T P_{h_{11}} + P_{h_{11}} A + C^T S_{v_{11}} C & P_{h_{11}} B_1 + C^T S_{v_{11}} D_1 \\ B_1^T P_{h_{11}} + D_1^T S_{v_{11}} C & D_1^T S_{v_{11}} D_1 - \gamma_\infty^2 I \end{bmatrix} \prec 0, \tag{10.69}$$

$$\begin{bmatrix} P_{h11} & I \\ I & U_{h11} \end{bmatrix} \succeq 0, \quad \begin{bmatrix} S_{v11} & I \\ I & T_{v11} \end{bmatrix} \succeq 0, \tag{10.70}$$

where $\mathcal{W}_c$ and $\mathcal{N}_D$ are a full column rank matrix whose images satisfy

$$\mathrm{Im}\,(\mathcal{W}_c) = \ker\left(\begin{bmatrix} D^T \\ B^T \end{bmatrix}\right), \quad \mathrm{Im}\,(\mathcal{N}_D) = \ker(D^T).$$

If the above Theorem holds, then the stabilizing controller matrices can be computed using the algorithm presented in (Paszke *et al.*, 2006).

10.8. Output feedback based controller design

The previous deliberations regarding the controller design of LRPs towards stability along the pass were based on the assumption that, for control issues, full information from the past (i.e., state vectors and pass profiles) was available for use. In some cases, the state vector $x_{k+1}(p)$ $(x_{k+1}(t))$ may not be available or, at best, only some of its entries are. In this situation, two approaches for stabilization are available. The first possibility to deal with this is the construction of the state observer and the application of that estimated state during the controller design procedure. The second way is to try to control the LRP directly, using only the pass profile (outputs) vectors. This approach was outlined in (Sulikowski *et al.*, 2004a) or, with an extended control law, in (Sulikowski *et al.*, 2004b).

Here, due to space limitations, the results regarding differential LRPs are presented only. For details on output controller design for discrete LRPs refer to, e.g., (Sulikowski, 2006).

As the first step, define the output control law which has the following form over $0 \le t < \alpha,\ k \ge 0$:

$$u_{k+1}(t) = \widetilde{K}_1 y_{k+1}(t) + \widetilde{K}_2 y_k(t). \tag{10.71}$$

In general, this control law is weaker than that of (10.22), and it is straightforward to provide examples where stability along the pass can be achieved using (10.22) but not (10.71).

Regarding output control, by definition, the pass profile produced on each pass is available for control purposes before the start of each new pass. Hence, this control law (and extensions) assumes the storage of the required previous pass profiles and that they are not corrupted by noise, etc.

To consider the effect of a control law of the form (10.71) on process dynamics, substitute the pass profile equation of (10.7) into (10.71) to obtain (assuming the required matrix inverse exists):

$$u_{k+1}(t) = (I - \widetilde{K}_1 D)^{-1}\widetilde{K}_1 C x_{k+1}(t) + (I - \widetilde{K}_1 D)^{-1}[\widetilde{K}_2 + \widetilde{K}_1 D_0] y_k(t), \tag{10.72}$$

and hence (10.72) can be treated as a particular case of (10.22) with

$$\begin{aligned} K_1 &= (I - \widetilde{K}_1 D)^{-1}\widetilde{K}_1 C, \\ K_2 &= (I - \widetilde{K}_1 D)^{-1}(\widetilde{K}_2 + \widetilde{K}_1 D_0). \end{aligned} \tag{10.73}$$

This route may encounter serious numerical difficulties (arising from the fact that (10.73) is a set of matrix non-linear algebraic equations), and hence it is purposeful to proceed by rewriting these last equations to finally obtain

$$(I - \widetilde{K}_1 D)K_1 = \widetilde{K}_1 C,$$

$$(I - \widetilde{K}_1 D)K_2 = \widetilde{K}_2 + \widetilde{K}_1 D_0,$$

and to assume that

$$K_1 = L_1 C. \tag{10.74}$$

Note that this assumption does not introduce any supplementary restrictions on the results developed but could be a source of difficulty in other cases, e.g., where uncertain processes are discussed.

It now follows that

$$\widetilde{K}_1 = L_1(I + DL_1)^{-1}, \tag{10.75}$$

$$\widetilde{K}_2 = [I - L_1(I + DL_1)^{-1}D]K_2 - L_1(I + DL_1)^{-1}D_0,$$

for any L_1 such that $I + DL_1$ is non-singular, and the following result is obvious:

Theorem 10.18. (Sulikowski, 2006) *Suppose that a differential LRP of the form described by (10.6)–(10.7) is subject to a control law of the form (10.71), and that (10.74) holds. Then the resulting closed loop process is stable along the pass if there exist the matrices $Y \succ 0$, $Z \succ 0$, $X \succ 0$ and N such that the following LMI holds:*

$$\begin{bmatrix} YA^T + AY + C^T N^T B^T + BNC & (*) & (*) \\ ZB_0^T + M^T B^T & -Z & (*) \\ CY + DNC & D_0 Z + DM & -Z \end{bmatrix} \prec 0, \tag{10.76}$$

$$XC = CY.$$

If this condition holds, then the control law matrices L_1 and K_2 are given by

$$L_1 = NX^{-1}, \quad K_2 = MZ^{-1}, \tag{10.77}$$

and it is required that $I + DL_1$ be non-singular. To compute the output controllers applicable in (10.72), it is necessary to apply (10.75).

Remark 10.6. *It is important to underline that here only the basic structure (containing only two factors) of the control law is presented. However, it is reasonable to extend the applied control law by additional delay factors. This leads to condition conservativeness reduction.*

To provide controllers applicable in such an extended control law it is necessary to perform the following operations:

- *choose additional delay factors that are used in the control law,*
- *define the mappings $\widetilde{K}_i$, $i = 1, 2, \cdots \Rightarrow K_i$, $i = 1, 2, \ldots$,*
- *apply delay operators to the closed loop system,*

- *apply appropriate elementary operations (leave the determinant invariant) to obtain the characteristic polynomial in the well-known form of (10.9),*
- *use an appropriate LMI condition to compute the partial controllers* $K_i,\ i = 1, 2, \ldots,$
- *re-map the obtained controllers into the output controllers.*

10.9. Control for performance

Return now to discrete LRPs. Note also that, in practical control schemes, the resulting stability in the closed loop is not enough. Now, when LMI conditions for computing controllers were presented in one of the previous sections, it is natural to extend the synthesis of the LRP to performance requirements under appropriate control.

To formalize the concept of performance used here, the goals of control considered in this section are defined as follows:

- stability along the pass in the closed loop system configuration,
- after a sufficiently large number of passes, the process is driven to the required reference signal $y_{ref}(p)$, $0 \le p \le \alpha - 1$ ($y_{ref}(t)$, $0 \le t < \alpha$),
- rejection of disturbances that influence the controlled LRP.

Here, disturbed state-space models of LRPs are considered. Hence the discrete LRP of (10.1)–(10.2) now has the following form over $0 \le p \le \alpha - 1$:

$$x_{k+1}(p+1) = Ax_{k+1}(p) + B_0 y_k(p) + Bu_{k+1}(p) + Ew(p), \tag{10.78}$$

$$y_{k+1}(p) = Cx_{k+1}(p) + D_0 y_k(p) + Du_{k+1}(p) + Fw(p), \tag{10.79}$$

and the differential LRP of (10.6)–(10.7) has the following form over $0 \le t < \alpha$:

$$\dot{x}_{k+1}(t) = Ax_{k+1}(t) + B_0 y_k(t) + Bu_{k+1}(t) + Ew(t), \tag{10.80}$$

$$y_{k+1}(t) = Cx_{k+1}(t) + D_0 y_k(t) + Du_{k+1}(t) + Fw(t). \tag{10.81}$$

It should be underlined that, in this case, it is assumed that the disturbances do not change in the direction from pass to pass (k), i.e., $w_{k_1}(p) = w_{k_2}(p),\ 0 \le p \le \alpha - 1$ ($w_{k_1}(t) = w_{k_2}(t),\ 0 \le t < \alpha$) for any two pass numbers $k_1,\ k_2 \in Z$. Indeed, this means that the disturbances are periodical (with the period equal α). Nevertheless, the disturbances can be dynamical in the along the pass direction p (or t). Due to that assumption, the disturbance vector is denoted as by $w(p)$ ($w(t)$) (without an explicit number of the pass given).

It is also important to underline that the assumed performance objectives are by no means exhaustive, and what is being undertaken here is an examination of the feasibility of designing one possible control law structure.

The selection of the reference signal $y_{ref}(p)$ (or $y_{ref}(t)$ for the differential case) should respect the physical constraints, namely, it has to be practically realizable.

However, more guidelines can be provided here regarding, e.g., the proper mathematical formulation of that signal. Hence, it can be assumed that the reference signal should be a continuous and differentiable function over the defined ranges.

To satisfy the the given control goals, there have been developed some approaches. The simplest one is based on fact that, for the LRP stable along the pass, $y_\infty(p)$ and $x_\infty(p)$ ($y_\infty(t)$ and $x_\infty(t)$) come closer to zero as $p \to \infty$ ($t \to \infty$). Then the additional factor is added to the control law and it drives the system to $y_{ref}(p)$ (or $y_{ref}(t)$ for the differential case). However, this approach does not concern the disturbance rejection at all.

In this section, the proportional plus integral control approach is assumed to be applied. For the classic 1D systems, a similar control scheme has been considered; for the description refer to (Liu *et al.*, 2001) and the references therein.

In terms of "acceptable", or desired, performance from a given example, a stronger demand regarding the stability is, in general, ensuring stability along the pass—this guarantees the existence of a limit profile which is stable as a 1D discrete linear system. The problem how to ensure stability along the pass for the LRPs considered has been presented before. In particular, it has been shown that a control law of the form (10.17) can be used to give this property. Moreover, the design of control law matrices can be implemented using LMIs where the basic result starts from interpreting Theorem 10.4 for the resulting closed loop state-space model.

Again, the question stated here is how to obtain a specified limit profile $y_{ref}(p)$ in the presence of disturbances.

Consider the disturbed state-space model of (10.78)–(10.79) at the point p on the pass k. Then the total tracking error at this point is defined as

$$\chi_k(p) := \sum_{j=0}^{k} \Big(y_j(p) - y_{ref}(p) \Big),$$

i.e., the error at the point p summed from the pass 0 to k. Substitution from the process state-space model gives

$$\begin{aligned} \chi_{k+1}(p) &= \chi_k(p) + y_{k+1}(p) - y_{ref}(p) \\ &= \chi_k(p) + Cx_{k+1}(p) + D_0 y_k(p) + Du_{k+1}(p) + Fw(p) - y_{ref}(p). \end{aligned} \tag{10.82}$$

Now, introduce the so-called extended output (pass profile) vector,

$$z_{k+1}(p) := \begin{bmatrix} y_{k+1}(p) \\ \chi_{k+1}(p) \end{bmatrix}.$$

Then (10.82) yields

$$z_{k+1}(p) = \begin{bmatrix} C \\ C \end{bmatrix} x_{k+1}(p) + \begin{bmatrix} D_0 & 0 \\ D_0 & I \end{bmatrix} z_k(p) + \begin{bmatrix} D \\ D \end{bmatrix} u_{k+1}(p) + \begin{bmatrix} 0 \\ -I \end{bmatrix} y_{ref}(p) + \begin{bmatrix} F \\ F \end{bmatrix} w(p).$$

Suppose now that the process of (10.78)–(10.79) is asymptotically stable, i.e., $r(D_0) < 1$. Then, as $k \to \infty$,

$$x_{k+1}(p) = x_k(p) \equiv x_\infty(p),$$
$$y_{k+1}(p) = y_k(p) \equiv y_\infty(p),$$
$$u_{k+1}(p) = u_k(p) \equiv u_\infty(p),$$

and let $\chi_\infty(p)$ denote $\lim_{k\to\infty} \chi_k(p)$. Then

$$\chi_{k+1}(p) = \chi_k(p) \equiv \chi_\infty(p),$$

and hence

$$x_\infty(p+1) = Ax_\infty(p) + B_0 y_\infty(p) + Bu_\infty(p) + Ew(p), \tag{10.83}$$

$$z_\infty(p) = \begin{bmatrix} C \\ C \end{bmatrix} x_\infty(p) + \begin{bmatrix} D_0 & 0 \\ D_0 & I \end{bmatrix} z_s(p) + \begin{bmatrix} D \\ D \end{bmatrix} u_s(p) + \begin{bmatrix} 0 \\ -I \end{bmatrix} y_{ref}(p) + \begin{bmatrix} F \\ F \end{bmatrix} w(p), \tag{10.84}$$

where $z_\infty(p) = \lim_{k\to\infty} z_k(p)$.

Next, define the following so-called incremental vectors:

$$\hat{z}_k(p) = z_k(p) - z_\infty(p),$$
$$\hat{u}_k(p) = u_k(p) - u_\infty(p),$$
$$\hat{x}_k(p) = x_k(p) - x_\infty(p).$$

Then, using (10.78)–(10.79) and (10.83)–(10.84), it is straightforward to obtain

$$\hat{x}_{k+1}(p+1) = A\hat{x}_{k+1}(p) + \hat{B}_0 \hat{z}_k(p) + B\hat{u}_{k+1}(p), \tag{10.85}$$

$$\hat{z}_{k+1}(p) = \hat{C}\hat{x}_{k+1}(p) + \hat{D}_0 \hat{z}_k(p) + \hat{D}\hat{u}_{k+1}(p), \tag{10.86}$$

where

$$\hat{B}_0 = \begin{bmatrix} B_0 & 0 \end{bmatrix}, \quad \hat{C} = \begin{bmatrix} C \\ C \end{bmatrix}, \quad \hat{D}_0 = \begin{bmatrix} D_0 & 0 \\ D_0 & I \end{bmatrix}, \quad \hat{D} = \begin{bmatrix} D \\ D \end{bmatrix},$$

and the key point now is that the influence of the disturbance has been completely rejected. The task now is to meet the specification when the limit profile (for the original process) is equal to the prescribed vector $y_{ref}(p)$. Note that (10.85)–(10.86) is of the structure of the discrete LRP of (10.1)–(10.2), and hence the known (formerly presented) methods of synthesis can be applied.

What is more, the matrix $\hat{D}_0$ in (10.86) always has eigenvalues with the modulus at least equal to unity, and hence this discrete LRP state-space model is asymptotically

unstable and, consequently, unstable along the pass. To obtain any (and, in particular, the required) limit profile from it, the control action must be applied. Moreover, in order to make this limit profile equal to $y_{ref}(p)$ with the given 1D transient performance specifications, stability along the pass in the closed loop is also required.

Now consider the control law of the form (10.17) applied to the extended model (10.85)–(10.86):

$$\begin{aligned}\hat{u}_{k+1}(p) &= K_x\hat{x}_{k+1}(p) + K_z\hat{z}_k(p)\\ &= K_x\hat{x}_{k+1}(p) + K_{z1}\hat{y}_k(p) + K_{z2}\hat{\chi}_k(p)\\ &= \begin{bmatrix} K_x & K_{z1} & K_{z2}\end{bmatrix}\begin{bmatrix}\hat{x}_{k+1}(p)\\ \hat{y}_k(p)\\ \hat{\chi}_k(p)\end{bmatrix}.\end{aligned} \tag{10.87}$$

Then the subsequent result gives an LMI based sufficient condition for closed loop stability along the pass together with a formula for computing control law matrices. The proof of this result follows immediately the interpretation of Theorem (10.7) and hence the details are omitted here.

Theorem 10.19. *Suppose that a control law of the form (10.87) is applied to a discrete LRP described by a state-space model of the form (10.85)–(10.86). Then the resulting closed loop process is stable along the pass if there exist the matrices* $Y \succ 0$, $Z \succ 0$, *and* N *such that the following LMI holds:*

$$\begin{bmatrix} Z-Y & 0 & Y\widetilde{A}_1^T + N^T\widetilde{B}_1^T\\ 0 & -Z & Y\widetilde{A}_2^T + N^T\widetilde{B}_2^T\\ \widetilde{A}_1Y + \widetilde{B}_1N & \widetilde{A}_2Y + \widetilde{B}_2N & -Y\end{bmatrix} \prec 0,$$

where

$$\widetilde{A}_1 = \begin{bmatrix} A & \hat{B}_0\\ 0 & 0\end{bmatrix},\quad \widetilde{A}_2 = \begin{bmatrix} 0 & 0\\ \hat{C} & \hat{D}_0\end{bmatrix},\quad \widetilde{B}_1 = \begin{bmatrix} B\\ 0\end{bmatrix},\quad \widetilde{B}_2 = \begin{bmatrix} 0\\ \hat{D}\end{bmatrix}.$$

If this condition holds, then the matrices in the control law are given by

$$[K_x\ K_{z1}\ K_{z2}] = NY^{-1}.$$

Suppose now that this last result holds. Then it follows immediately that $y_\infty(p) = y_{ref}(p)$ as required. Moreover,

$$\begin{aligned}u_{k+1}(p) = K_x\Big(x_{k+1}(p) - x_\infty(p)\Big) + K_{z1}\Big(y_k(p) - y_{ref}(p)\Big)\\ + K_{z2}\Big(\chi_k(p) - \chi_\infty(p)\Big) + u_\infty(p),\end{aligned}$$

and also

$$-K_xx_\infty(p) - K_{z1}y_{ref}(p) - K_{z2}\chi_\infty(p) + u_\infty(p) = 0.$$

Hence the final form of the control law to be applied to the original process is

$$u_{k+1}(p) = K_x x_{k+1}(p) + K_{z1} y_k(p) + K_{z2}\chi_k(p). \tag{10.88}$$

Rewrite now the part of the right-hand side of the control law (10.87) in the form

$$K_x x_{k+1}(p) + K_{z1} y_k(p) = \begin{bmatrix} K_x & K_{z1} \end{bmatrix} \begin{bmatrix} x_{k+1}(p) \\ y_k(p) \end{bmatrix} =: KX^a_{k+1}(p),$$

where $X^a_{k+1}(p)$ is termed the augmented state vector. Then, immediately, it is easy to see that the final control law (10.88) has a two-term structure, where the first term, $KX^a_{k+1}(p)$, is static (the proportional control action for stability) and the second one, $K_{z2}\chi_k(p)$, is the integral action to enforce the tracking of the requested limit profile $y_{ref}(p)$.

A similar result regarding PI control has also been obtained for differential LRPs, and a detailed description can be found in (Sulikowski, 2006).

10.10. Conclusions

In this chapter the basics and the major recent results in the area of linear repetitive processes have been outlined. The results reported here are a portion of problems solved by this research group, to which also Ł. Hładowski, B. Cichy, and J. Bochniak belong and continue their research towards Ph.D. As seen in the literature, the group enjoys significant international co-operation, where Prof. E. Rogers from Southampton, UK, holds a crucial position.

References

Agathoklis P., Jury E.I. and Mansour M. (1993): *Algebraic necessary and sufficient conditions for the stability of 2-D discrete systems.* — IEEE Trans. Circuits and Systems – Part II: Analog and Digital Signal Processing, Vol. 40, No. 4, pp. 251–257.

Amann N., Owens D.H. and Rogers E. (1998): *Predictive optimal iterative learning control.* — Int. J. Control, Vol. 69, No. 2, pp. 203–226.

Basu S. (2002): *Multidimensional causal, stable, perfect reconstruction filter banks.* — IEEE Trans. Circuits and Systems – Part I: Fundamental Theory and Applications, Vol. 49, No. 6, pp. 832–842.

Benton S.E. (2000): *Analysis and Control of Linear Repetitive Processes.* — PhD thesis, University of Southampton, UK.

Bliman P.-A. (2002): *Lyapunov equation for the stability of 2-D systems.* — Multidimensional Systems and Signal Processing, Vol. 13, pp. 201—222.

Boyd S., Ghaoui L.E., Feron E. and Balakrishnan V. (1994): *Linear Matrix Inequalities in System and Control Theory.* — SIAM Studies in Applied and Numerical Mathematics, Vol. 15, Philadelphia: SIAM.

Bracewell R.N. (1995): *Two-Dimensional Imaging.* — Upper Saddle River: Prentice Hall.

D'Andrea R. and Dullerud G.E. (2003): *Distributed control design for spatially interconnected systems.* — IEEE Trans. Automatic Control, Vol. 48, No. 9, pp. 1478–1495.

Du C. and Xie L. (2002): *H_∞ Control and Filtering of Two-dimensional Systems.* — Lecture Notes in Control and Information Sciences, Vol. 278, Berlin: Springer-Verlag.

Dudgeon D.E. and Merserau R.M. (1984): *Multidimensional Digital Signal Processing.* — Englewood Cliffs: Prentice Hall.

Fornasini E. and Marchesini G. (1978): *Doubly-indexed dynamical systems: state-space models and structural properties.* — Mathematical Systems Theory, Vol. 12, pp. 59–72.

Gałkowski K., Lam J., Rogers E., Xu S., Sulikowski B., Paszke W. and Owens D.H. (2003a): *LMI based stability analysis and robust controller design for discrete linear repetitive processes.* — Int. J. Robust and Nonlinear Control, Vol. 13, No. 13, pp. 1195–1211.

Gałkowski K., Paszke W., Rogers E., Xu S., Lam J. and Owens D.H. (2003b): *Stability and control of differential linear repetitive processes using an LMI setting.* — IEEE Trans. Circuits and Systems – Part II: Analog and Digital Signal Processing, Vol. 50, No. 10, pp. 662–666.

Gałkowski K., Paszke W., Sulikowski B., Rogers E. and Owens D.H. (2003c): *Stability and control of a physical class of 2D continuous-discrete linear systems using an LMI setting.* — Proc. *American Control Conf., ACC*, Denver, USA, Vol. 6, pp. 5058–5063.

Gałkowski K., Rogers E., Xu S., Lam J. and Owens D.H. (2002): *LMIs – A fundamental tool in analysis and controller design for discrete linear repetitive processes.* — IEEE Trans. Circuits and Systems – Part I: Fundamental Theory and Applications, Vol. 49, No. 6, pp. 768–778.

Gałkowski K. and Wood J. (Eds) (2001): *Multidimensional Signals, Circuits and Systems.* — London: Taylor and Francis.

Kaczorek T. (1985): *Two-dimensional Linear Systems.* — Lecture Notes in Control and Information Sciences, Vol. 68, Berlin: Springer-Verlag.

Khargonekar P.P., Petersen I. and Zhou K. (1990): *Robust stabilization of uncertain linear systems: quadratic stabilizability and H_∞ control theory.* — IEEE Trans. Automatic Control, Vol. 35, No. 3, pp. 356–361.

Liu G.P., Duan G.R. and Dixon R. (2001): *Robust control with stable proportional-integral-plus controllers.* — Proc. *European Control Conf., ECC*, Porto, Portugal, pp. 3201–3206.

Longman R.W. (2000): *Iterative learning control – dynamic systems that learn in time and repetitions.* — Proc. 2nd Int. Workshop *Multidimensional (nD) Systems*, Czocha Castle, Lower Silesia, Poland, pp. 55–65.

Owens D.H. and Rogers E. (1999): *Stability analysis for a class of 2D continuous-discrete linear systems with dynamic boundary conditions.* — Systems and Control Letters, Vol. 37, pp. 55–60.

Paszke W. (2005): *Analysis and Synthesis of Multidimensional System Classes Using Linear Matrix Inequality Methods.* — Lecture Notes in Control and Computer Science, Vol. 8, University of Zielona Góra Press.

Paszke W., Gałkowski K., Rogers E., Kummert A. and Owens D.H. (2006): *H_2 and mixed H_2/H_∞ control of differential linear repetitive processes.* — IEEE Trans. Circuits and Systems, (submitted).

Paszke W., Gałkowski K., Rogers E. and Owens D.H. (2004): *H_∞ control of differential linear repetitive processes.* — Proc. *American Control Conf., ACC*, Boston, USA, Vol. 2, pp. 1386–1391.

Paszke W., Gałkowski K., Rogers E. and Owens D.H. (2005): *H_2 control of differential linear repetitive processes.* — Proc. 16th IFAC *World Congress*, Prague, Czech Republic, CD-ROM.

Roberts P.D. (2002): *Two-dimensional analysis of an iterative nonlinear optimal control algorithm.* — IEEE Trans. Circuits and Systems – Part I: Fundamental Theory and Applications, Vol. 49, No. 6, pp. 872–878.

Roesser R.P. (1975): *A discrete state-space model for linear image processing.* — IEEE Trans. Automatic Control, Vol. 20, No. 1, pp. 1–10.

Rogers E., Gałkowski K. and Owens D.H. (2007): *Control Systems Theory and Applications for Linear Repetitive Processes.* — Lecture Notes in Control and Information Sciences, Vol. 349, Berlin: Springer-Verlag, (in print).

Rogers E. and Owens D.H. (1992): *Stability Analysis for Linear Repetitive Processes.* — Lecture Notes in Control and Information Sciences, Vol. 175, Berlin: Springer-Verlag.

Shi Y.Q. and Zhang X.M. (2002): *A new two-dimensional interleaving technique using successive packing.* — IEEE Trans. Circuits and Systems – Part I: Fundamental Theory and Applications, Vol. 49, No. 6, pp. 779–789.

Sulikowski B. (2006): *Computational Aspects in Analysis and Synthesis of Repetitive Processes.* — Lecture Notes in Control and Computer Science, Vol. 11, University of Zielona Góra Press.

Sulikowski B., Gałkowski K., Rogers E. and Owens D.H. (2004a): *Output feedback control of discrete linear repetitive processes.* — Automatica, Vol. 40, No. 12, pp. 2163–2173.

Sulikowski B., Gałkowski K., Rogers E. and Owens D.H. (2004b): *LMI based output feedback control of discrete linear repetitive processes.* — Proc. *American Control Conf., ACC*, Boston, USA, pp. 1998–2003.

Sulikowski B., Gałkowski K., Rogers E. and Owens D.H. (2005): *Control and disturbance rejection for discrete linear repetitive processes.* — Multidimensional Systems and Signal Processing, Vol. 16, No. 2, pp. 199–216.

Yamada M. and Saito O. (1996): *2D model-following servo system.* — Multidimensional Systems and Signal Processing, Vol. 10, pp. 71–91.

Chapter 11

QUANTUM INFORMATION PROCESSING WITH APPLICATIONS IN CRYPTOGRAPHY

Roman GIELERAK*, Eugeniusz KURIATA*
Marek SAWERWAIN*, Kamil PAWŁOWSKI*

11.1. Introduction

There seems no doubt that electronic communications have become one of the main pillars of the present day society and their ongoing boom requires the development of new technologies to secure data transmission and data storage. Nowadays we all see that many paper-based communications have already been replaced by theirs electronic counterparts, raising the challenge to find proper and maximally secure electronic equivalents to classical stamps, seals and hand-written signatures as examples.

The currently used cryptographic implementations only provide conditional security that relies on limited computational and technological capabilities of the adversary, which particularly concerns the security of publickey-based cryptography protocols. Even though secretkey-based protocols employing the Vernam one-time-pad algorithm offer unconditional security against adversaries possessing unlimited computational power and technological abilities, they face the problem of how to securely distribute the key.

As we have said, the security of the presently employed public-key cryptography systems and protocols, like RSA, ElGamal, Diffe-Hellman or DSA, for example, rests on various computational problems, which are believed, before the forthcoming era of quantum computations, to be intractable by using classical machines. The encryption and description algorithms in the above-mentioned cryptosystems heavily use the so-called one way functions that are easy to compute in one direction, although computations of their inverses should be unfeasible in realistic applications. For example,

* Institute of Control and Computation Engineering
e-mails: {R.Gielerak, E.Kuriata, K.Pawlowski, M.Sawerwain}@issi.uz.zgora.pl

the security of the RSA system rests entirely on the hypothesis that the problem of factoring large integers belongs to the class of problems that could not be solved in polynomial (in size of the number considered) time. However, the appearance of the polynomial time solution of this problem, the so-called Shor algorithm, completely destroyed this commonly expressed opinion on the security of the RSA system. The same danger concerns other public-key cryptosystems as well.

After discovering the new algorithm for the factorization of large numbers and working in polynomial time, the so-called Shor algorithm, enormous activity in the field of quantum information has apreared. In particular, a variety of new quantum algorithms for significantly speeding up solutions of several classical problems, like, for example, the searching problem of an unstructured data set and many others have been discovered.

In Section 11.2 the main ideas for quantum calculations and quantum algorithms will be presented very briefly. In particular, the concept of quantum labeled transition systems will be presented and applied to the discussion of the structure of some selected algorithms in Section 11.3. The linear structure of some quantum algorithms will be pointed out.

It is well known that the main obstacle in constructing a large scale quantum computer is the decoherence problem. The very nature and some mathematical description of decoherence processes will be presented in Section 11.4.

Other problems connected with public-key cryptography systems concern the problem of building up the necessary infrastructure. The main purpose of the Public-Key Infrastructure (PKI) is to provide mechanisms for issuing, storing and distributing public key certificates. In the absence of universal point-to-point channels or globally trusted third parties, this is a very non-trivial issue. After this discussion, recently formulated quantum protocols guarantying absolute secure communications in secret-key cryptography will be presented and their up-to-date state of the art from the point of view of present technological impletientation abilities will be presented.

Also, the main ideas connected with the quantum computer simulator, recently constructed by our research group, together with our plans concerning its future developments as well will be presented in Section 11.6 of the present chapter.

11.2. Quantum computation and quantum algorithms

A quantum logical unit, playing the role similar to that of a classical bit, is called the qudit, by which we mean an abstract quantum system whose Hilbert space of states is unitarly isomorphic to the d-dimensional complex Hilbert space $\mathcal{C}^d$. The best-known quantum logical unit of that kind corresponds to the case $d = 2$ and is known under the name qubit.

According to the laws of quantum theory, pure states of one qudit system are given by unit rays in the space $\mathcal{C}^d$, the set $R(\mathcal{C}^d)$ that is obtained by dividing the unit sphere of $\mathcal{C}^d$ by an action of the circle group S, i.e.

$$R(\mathcal{C}^d) = \{z \in \mathcal{C}^d : \|z\| = 1\}/S. \tag{11.1}$$

In several situations, the introduced notion of pure states will not be sufficient and the notion of *mixed states* of a qudit must be employed as well. Mixed states of a qudit

forms a convex and closed subset $E(\mathcal{C}^d)$ in the space of all linear mappings of $\mathcal{C}^d$, and are selected by the requirements $\rho \in E(\mathcal{C}^d) \Leftrightarrow \rho \geq 0$ and $\mathrm{Tr}(\rho) = 1$. In particular, the pure states correspond to extremal points of the set $E(\mathcal{C}^d)$ and can be identified with density matrices that are idempotent and from which it is easy to deduce that the pure states can be identified with the orthogonal projectors on one-dimensional subspaces in $\mathcal{C}^d$.

By the *quantum register* we mean a quantum system composed of a finite number of qudits. Then the corresponding space of states is given by the tensor product of one-qudit Hilbert spaces.Thus, if our quantum register is composed of n qudits, then the corresponding Hilbert space of states is $\mathcal{C}^{d^n}$. The pure states of a register are described by unit rays in the product space $\mathcal{C}^{d^n}$ and the corresponding mixed states by density matrices defined on $\mathcal{C}^{d^n}$.

Several gueinine quantum features significantly differentiate between classical and quantum information processing. Among them, the principle of superposition and the phenomenon of entanglement, although not well understood yet, seen to be most important. In the case of pure states, entanglement with respect to the twofold decomposition of the corresponding Hilbert space seems to be well understood at least from a mathematical point of view. For this, let us assume that the Hilbert space $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$, then we say that the vector $\mathcal{H}$ is entangled with respect to the decomposition as above iff the vector could not be presented in the form of a simple tensor. Straightforward application of the Singular Value Decomposition (SVD) algorithm leads to the following result known as the Schmidt decomposition:

Theorem 11.1. (Schmidt decomposition theorem)
Let $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$. Then, for any $|\psi\rangle \in \mathcal{H}$, there exist a number $r(|\psi\rangle)$ called the Schmidt rank of $|\psi\rangle$, a finite sequence of numbers $(\lambda_\alpha), \alpha = 1, \ldots, \min\{\dim \mathcal{H}_1, \dim \mathcal{H}_2\}$ called the Schmidt coefficients of $|\psi\rangle$, and a pair of complete orthonormal systems $\{|\Theta_i\rangle\} \subset \mathcal{H}_1, i = 1, \ldots, \dim \mathcal{H}_1$ and $\{|\Lambda_j\rangle\} \subset \mathcal{H}_2, j = 1, \ldots, \dim \mathcal{H}_2$, and such that

$$|\psi\rangle = \sum_{\alpha=1}^{r(|\psi\rangle)} \lambda_\alpha |\Theta_\alpha\rangle \otimes |\Lambda_\alpha\rangle. \tag{11.2}$$

The natural Hilbert space structure could be defined on the space $L(\mathcal{H})$ of bounded linear operations on $\mathcal{H}$ with the use of the Hilbert-Schmidt inner product $\langle A|B\rangle = \mathrm{Tr}(A^\dagger \cdot B)$. The arising Hilbert space will be denoted as $\mathcal{HS}(\mathcal{H})$. Let us assume now that $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$. Applying again the SVD theorem to the space $\mathcal{HS}(\mathcal{H})$ we obtain the following result:

Proposition 11.1. *Let $\mathcal{H} = \mathcal{H}_1 \otimes \mathcal{H}_2$ and let $A \in \mathcal{HS}(\mathcal{H})$. Then there do exist a number $s(A) \leq \min\{(\dim \mathcal{H}_1)^2, (\dim \mathcal{H}_2)^2\}$ called a Schmidt rank of a matrix A, a sequence of numbers $(\lambda_\alpha), \alpha = 1, \ldots, s(A)$ called the Schmidt numbers of A and two families of matrices $\{E_i\} \subset L(\mathcal{H}_1), \{F_j\} \subset L(\mathcal{H}_2)$ forming complete orthonormal systems in the corresponding spaces $\mathcal{HS}(\mathcal{H}_i), i = 1, 2$, and such that the following decomposition holds:*

$$A = \sum_{i=1}^{s(A)} \lambda_i E_i \otimes F_i. \tag{11.3}$$

Whether this canonical decomposition of matrices with respect to tensor product decomposition of the carrying Hilbert space $\mathcal{H}$ is of any relevance in the problem of entanglement will be studied elsewhere, see also the approach to this problem in (Terhal and Horodecki, 2004). The important point is that this decomposition is uniquely determined in opposition to many different representations of density matrices in terms of pure states. Here we mention only that this result provides us with some LOCC invariant partitioning and partial order relations as well on the space $L(\mathcal{H})$.

By a quantum computations process or a quantum algorithm leading to the solution of a particular problem P one usually means the process consisting of the following steps:

- careful preparations of the initial state(s) of the register(s),
- performance of a well prescribed finite sequence of changes of states of the register(s),
- recovery of the solution of the problem analysed from the final state(s) of the register(s).

Many things have to be explained in order to properly understand the above notion of quantum computations. It is because of a very limited space that we restrict ourselves to some selected remarks on this. Firstly we should specify what kind of operations on the quantum register(s) are allowed. At present several schemes of quantum computations are being invented. They differ mainly by the allowed set of quantum operations on the actual state of the register. Let us list some of them.

11.2.1. Unitary standard quantum machines (UQCM)

The basic set of operations by which the desidered changes of actual states of the system are being forced might consist of a small selected set of unitary gates forming a universal library of gates or an approximately universal library. It is not difficult to prove that, for example, the set of all unitary one-qubit gates together with any entangling two-qubit gate forms a universal set of gates for any $d \geq 2$. Additionally, this universal library of one, two and sometimes three-qudit gates is accompanied by measurement operations.

Proposition 11.2. *Let P be a problem which possess a solution by an algorithm that can be executed on the classical Turing machine. Then there exists at least one quantum algorithm solving the problem P.*

Proof. A sketch of the proof is the following: as is well known, if the problem P is computable on the Turing machine then there exist classical Boolean circuits implementing this solution. As all the classical logical gates could be exchanged by the reversible quantum counterparts, the proof follows. ■

In particular, it follows (from the proof of this proposition) that any irreversible classical computations can be replaced by reversible ones. As is well known, this might be of some importance in attempts to solve thermodynamical problems connected to the question of accelerating classical computations (on the hardware level).

However, the real interest in quantum computations methods comes from the observations that in some specific situations there exist quantum algorithms solving some classical problems faster than their known classical counterparts. The problem of factoring large integer numbers in polynomial time by implementing the so-called Shor algorithm is the most spectacular example of this sort. The famous Grover algorithm for searching an unstructured data basis is another example.

11.2.2. One Way Quantum Computing Machines (1WQCM)

In this kind of quantum computations a very important step of computations consists in careful preparations of the initial state of the register. The initial, different cluster states preparations seem to be the most popular approach at present. In contrast to unitary calculations measurement processes and classical communications play a dominant role in these processes. Therefore, it is not a big surprise that certain teleportations protocols have become to play an essential role in calculation processes. In fact, depending on whether the teleportation of quantum information encoded in the corresponding quantum states is dominating and transparent or not this sort of irreversible calculations is divided further into several subclasses. The so-called Teleportations-based Quantum Calculations (TQC) are an example of that sort.

11.2.3. Adiabatic Quantum Computer Calculations (AQCM)

The famous 3-SAT problem being representative of an NP-complete problem from the point of view of classical complexity theory is a very illustrative and suggestive example of how one can use the well known adiabatic analysis of the spectrum of the corresponding Hamiltonians to encode the solution of the 3-SAT problem in the lowest energy eigenfunction of the final Hamiltonian of the system. This kind of quantum computations is a tricky and ingenious application of UQCM, only the essential point being the ability to choice a suitable adiabatic evolution when discussing a particular problem P. In fact, there is known certain (polynomial) equivalence in between unitary standard quantum machines and this approach (Aharonov *et al.*, 2004), and it seems to be a very non-trivial task.

Other known quantum calculations schemes are known under names of TOPological Quantum Calculations (TOPQC) and Geometrical Quantum Calculations (GQC) (Kitayev, 2003; Vedral, 2003).

11.2.4. Discussion

Having sucha, rather large, variety of quantum computation possibilities, an interesting question seems to be that of computational equivalence of all these a priori quite different methods of quantum calculations. There are only very limited results obtained. Among them, polynomial, computational equivalence in between UQCM and AQCM seems to be rather a deep one (Aharonov *et al.*, 2004). The suitable uniforming mathematically language of labelled quantum transition systems has been proposed by us as for studying in a systematic way this kind of problems. Some particular results on the problem of computational equivalence in between different quantum

computing schemes was given in (Sawerwain *et al.*), see also some remarks in the section below.

Without any doubts, one of the main questions in this area is to characterise the class of classical problems for which there exists a quantum algorithm with less computational complexity than the classical one. In particular, the major question whether there exists a quantum algorithm solution of at least one NP-complete problem (and therefore of all of them) and with polynomial complexity might be a revolutionary breakthrough in the searching for efficient computations of hard problems, i.e. of the NP-class. But whether there exists any such a solution is still under discussion, and the factoring problem is believed not to be an NP-complete problem, unfortunately. Here we point out that a typical simulation problem of a genuine quantum system is by very nature of non-polynomial computational complexity from the point of view of classical computing machines but not on a quantum machine. So if there does not exist an NP-complete problem solvable in polynomial time on quantum machines, then there does not exist a quantum system the simulation of which is not of the NP-class.

11.3. Semantic aspects of quantum algorithms and quantum programming languages

Operational semantics (Plotkin, 2004a) developed in the mid 1970s is the first formal method describing the behavior of computer programs. Nowadays, three major approaches to the semantics of programming languages exist: operational, denotational and axiomatic semantics. In this section we consider only the operational approach to the semantics. It is especially significant in the quantum computation science, because today only few quantum algorithms are known and many researchers take a sceptical point of view concerning further development of quantum algorithms.

New quantum algorithms are very hard to discover because the synthesis of an algorithm is performed on a very low level. Additionally, the quantum computational model is still not completely understood. For example, the role of entanglement is not fully explained, and this phenomenon plays a significant role in several quantum algorithms: the teleportation protocol, superdense coding and the cryptographic protocol, called E91, are some examples.

We can give four ideas which are important to approach this field of research (Sawerwain *et al.*, 2006):

- operational semantics is a simple theory to express the meaning of classical programs: a similar theory can be formulated for quantum computation model,
- we need a theoretical notion to compare two seemingly different quantum algorithms or quantum programs,
- the quantum computation model is different from the classical computation model (superposition, entanglement, linear computations) but the application of some notions of classical semantics theory is quite useful to describe the meaning of quantum programs,
- natural algorithm synthesis quantum programs similar to the classical predicate calculus can be developed.

11.3.1. Quantum labelled transition system

Several types of quantum Labelled Transition Systems (qLTSs) can be introduced. Let M be a quantum system. Then, according to the general theory (Peres, 1995), there exists an associated Hilbert space $\mathcal{H}_M$ of states. In particular, pure states of M are given by unit rays in $\mathcal{H}_M$. Although there are situations in which the difference between the unit vector and the unit ray becomes important (i.e. topological quantum calculations), in the following analysis the set of (pure) states will be identified with (sometimes a subset of) the unit sphere $\partial E(\mathcal{H}) = (\{|\psi\rangle \in \mathcal{H} : \||\psi\rangle\| = 1\} \subset \mathcal{H})$.

Lemma 11.1. *Let $\{x_1, \ldots, x_k\}$ and $\{y_1, \ldots, y_k\}$ be two sets of vectors from a Hilbert space $\mathcal{H}$. If $\langle x_i|x_j\rangle = \langle y_i|y_j\rangle$ for all i,j, then there exists an unitary operator U acting in $\mathcal{H}$ such that $Ux_i = y_i$ for any $i = 1, \ldots, k$. The operator U is uniquely determined iff the rank of the Gram matrix $\langle x_i|x_j\rangle$ is equal to the dimension of $\mathcal{H}$.*

Proof. The proof of this lemma can be found in (Hirvensalo, 2001). ■

In the quantum labelled transition systems defined in this paper we consider two basic types of transition:

- the β type transition, denoted by the symbol $\overset{\beta_U}{\twoheadrightarrow}$, represented by a unitary operator from $\mathcal{U}(\mathcal{H})$. The β type transition is denoted by the following rule:

$$\frac{-}{|\psi\rangle \overset{\beta_U}{\twoheadrightarrow} |\psi'\rangle},$$

- the μ type transition, denoted by the symbol $\overset{\mu}{\twoheadrightarrow}$. This step is represented by the measurement operation from $\mathcal{O}(\mathcal{H})$, and denoted by the following rule, which measures the whole quantum register:

$$\frac{-}{|\psi\rangle \overset{\mu}{\twoheadrightarrow} |\psi_k\rangle_\perp}.$$

Of course, for the $\overset{\beta_U}{\twoheadrightarrow}$ type transition, we distinguish an adjoin Hermitian operator $\overset{\beta_{U^\dagger}}{\twoheadrightarrow}$. This operator represents the reverse operation and, naturally, it is still a unitary operator.

Lemma 11.2. *For any quantum computation process based on the β and μ step, whose corresponding states evolution is given by the sequence $\{q_0, q_1, q_2, \ldots, q_n\}$, there exists a sequence of unitary actions $\{u_0, u_1, u_2, \ldots, u_{n-1}\}$ such that $q_{i-1} \overset{u_i}{\twoheadrightarrow} q_i$.*

Definition 11.1. The quantum labelled transition system is the triple $\langle E(\mathcal{H}), \mathrm{Op}, \twoheadrightarrow\rangle$, where

- $E(\mathcal{H})$ is a closed subspace of the set of all states on $\mathcal{H}$,
- Op is a certain set of operations acting on the subset of the set of states $E(\mathcal{H})$ that might include, among others, some unitary operations, physical operations (i.e. completely positive maps) and some measurement operations (as described above), etc.,

- $\twoheadrightarrow$ is the labelled transition relation $\twoheadrightarrow \subseteq E(\mathcal{H}) \times \mathrm{Op} \times E(\mathcal{H})$. In particular, we write $\rho \overset{A}{\twoheadrightarrow} \rho'$ if $(\rho, A, \rho') \in \twoheadrightarrow$.

Trace semantics is an important and useful notion in the tree proof. We use this notion to prove determinism property of superdense coding in the next subsection.

Definition 11.2. Let $L = \langle \mathcal{S}, \mathcal{A}, \rightarrow \rangle$ be a labelled transition system and a pair $(s_{\mathrm{in}}, s_{\mathrm{fin}})$ of states called respectively the initial and the final state. Then the operational trace for the given pair is defined as

$$\mathrm{Top}(s_{\mathrm{in}}, s_{\mathrm{fin}}) = \{\bar{a} = (a_1, a_2, \ldots, a_n) \in \mathcal{A}^* \mid \exists\, \bar{s} = (s_1, s_2, \ldots, s_n) \in \mathcal{S}^* \text{ such that } s_{\mathrm{in}} \xrightarrow{a_1} s_1 \xrightarrow{a_2} s_2 \xrightarrow{a_3} s_4 \ldots s_n \xrightarrow{a_n} s_{\mathrm{fin}}\}.$$

The space of all traces of L will be denoted as $\mathrm{Top}(L)$:

$$\mathrm{Top}(L) = \bigcup_{(s_{\mathrm{in}}, s_{\mathrm{fin}}) \in \mathcal{S}^{\times 2}} \mathrm{Top}(s_{\mathrm{in}}, s_{\mathrm{fin}}).$$

A detailed content of the allowed set of quantum actions Op and the proper choice of $E(H)$ are factors differentiating between the types of qLTSs. Let us note two examples of such a kind of qLTSs. The first one is defined as

Definition 11.3. A qLTS $\langle E_r(H), \mathrm{Op}, \twoheadrightarrow \rangle$ is a purely unitary qLTS if and only if $E_r(H) = \partial E(H)$, $\mathrm{Op} = \{\mathcal{U}(H)\}$.

The second system allows measuring operations in the action set:

Definition 11.4. A qLTS $\langle E(H), \mathrm{Op}, \twoheadrightarrow \rangle$ is a closed non-unitary qLTS (a cnuqLTS) iff $E(H) = \partial E(\mathcal{H})$ and Op contains $\mathcal{U}(\mathcal{H})$ and some $ob \subseteq \{ob(A) : A \in \mathcal{O}(\mathcal{H})\}$.

If the quantum algorithm is composed of unitary gates and a measurement process is applied to the base states, the computation process is deterministic.

Proposition 11.3. *The deterministic quantum algorithm has the following operational description:*

$$\overline{|\psi^{n-1}\rangle \overset{\beta_U}{\twoheadrightarrow} |\psi^n\rangle}, \quad \overline{|\psi^n\rangle \overset{\beta_U}{\twoheadrightarrow} |\psi^{n+1}\rangle_\perp}, \quad \overline{|\psi^{n+1}\rangle_\perp \overset{\mu}{\twoheadrightarrow} |\psi^{n+1}\rangle_\perp}.$$

The class of unitary qLTSs introduced in Definition 11.3 can be used to replace any non-unitary qLTS from Definition 11.4. However, the price for it is the necessity to use probabilistic type qLTSs:

Proposition 11.4. *A closed non-unitary quantum labelled transition system L (defined in Definition 11.4) can be simulated by a probabilistic, unitary quantum labelled system U (defined in Definition 11.3). The system U is weaker than the system L, which is expressed as*

$$U \sqsubseteq L.$$

Proof. This proposition is a direct consequence of Lemma 11.2. From Lemma 11.1 it follows that for any given pair of the pure quantum state x and y there exists a unitary map U such that $Ux = y$. ■

More information about the quantum labelled transition systems is presented in (Sawerwain *et al.*).

11.3.2. Operational description of superdense coding

This section presents the deterministic property of superdense coding (superdense coding protocol). To obtain better legibility, the proof tree is split into two sections; i.e. one for the classical states $(00)_2$ and $(01)_2$:

$$\frac{\dfrac{|AB\rangle \xrightarrow{B} |00\rangle_\perp}{(00)_2, |AB\rangle \xrightarrow{I(A)} |AB\rangle} \quad \dfrac{|A^1B\rangle \xrightarrow{B} |01\rangle_\perp}{(01)_2, |AB\rangle \xrightarrow{X(A)} |A^1B\rangle}}{|AB\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)}, \tag{11.4}$$

and the other for the classical states $(10)_2$ and $(11)_2$:

$$\frac{\dfrac{|A^1B\rangle \xrightarrow{B} |10\rangle_\perp}{(10)_2, |AB\rangle \xrightarrow{F(A)} |A^1B\rangle} \quad \dfrac{|A^1B\rangle \xrightarrow{B} |11\rangle_\perp}{(11)_2, |AB\rangle \xrightarrow{X(A),F(A)} |A^1B\rangle}}{|AB\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)}. \tag{11.5}$$

Proposition 11.5. *Superdense coding fulfils Proposition 11.3.*

Proof. The proof is very short if the notion of the operational trace is used. The space of the operational traces of the superdense protocol contains four sequences:

$$\xi_0 = (I, B), \quad \xi_1 = (X, B), \quad \xi_2 = (F, B), \quad \xi_3 = (X, F, B).$$

Only unitary actions are used in these traces. Therefore, we can conclude that the superdense protocol is deterministic. ■

An additional measurement step is executed on states which are eigenstates of observable corresponding to B. The corresponding measurement operator represents the μ computational step: $|\psi\rangle_\perp \overset{\mu}{\twoheadrightarrow} |\psi\rangle_\perp$.

11.4. Decoherence processes

As is well known, quantum states are unfortunately extremely fragile and in the future physical implementation of the quantum computing machine we will be confronted with the impossibility of isolating the quantum register on which the desired computational process runs from disturbances caused by unavoidable interactions with the environment. This process is called quantum noise. Quantum noise affects the desired states of the quantum register in the course of any specific computations in a way that is called the quantum decoherence process.

It is the main aim of the present section to provide us with some mathematical models of quantum noise and the corresponding decoherence of states of the register. When the decoherence process sets in, the measurement of the final states of the register will not, with very high probability, produce the desidered result of computations, causing a serious failure of our quantum calculations.

The theory of quantum error correcting codes was established a decade ago as the primary tool for fighting decoherence in quantum computers and quantum communication systems. The first Quantum Error Correction Code (QECC) was provided by (Shor, 1995) and is known under the name the of nine-qubit single error correction code. The first nine-qubit single error-correcting code was a quantum analog of the classical repetition code, which stores information redundantly by duplicating each bit several times. Probably the most striking development in quantum error correction theory is the use of the stabilizer formalism (Gottesman and Lomonaco, 2002; Preskill *et al.*, 1998), where quantum codes are subspaces ("code spaces") in the Hilbert space, and they are specified by giving the generators of an Abelian subgroup of the Pauli group, called the stabilizer of the code space. Essentially, all QECCs developed to date are stabilizer codes. The problem of finding QECCs was reduced to that of constructing classical dual-containing quaternary codes. When binary codes are viewed as quaternary, this amounts to the well known Calderbank-Shor-Steane construction (Calderbank *et al.*, 1997). The requirement that a code contain its dual is a consequence of the need for a commuting stabilizer group. The virtue of this approach is that we can directly construct quantum codes from classical codes with a certain property, rather than having to develop a completely new theory of quantum error correction from scratch.

Unfortunately, the need for a self-orthogonal parity check matrix presents a substantial obstacle in importing the classical theory in its entirety, especially in the context of modern codes such as Low-Density Parity Check (LDPC) codes.

It is our aim in the present section to prepare a ground for a careful discussion of different possible scenarios for decoherence processes that can disturb actual quantum computations in situations when we are dealing with higher dimensional quantum logical units called qudits.

We adopt the semigroup approach to describe the evolution of a quantum system S, which will be identified with a quantum register composed of qudits with arbitrary d, which is embedded in a quantum enviroment called the bath B. Under the assumptions of the Markovianity of the underlying dynamics, complete positivity and the initial state decoupling between the system and the bath, the global unitary dynamics of the system plus the bath when restricted to the system Hilbert space sector $\mathfrak{h}_s$ of the total Hilbert space $\mathfrak{h}_s \otimes \mathfrak{h}_b$ are described by the following dynamical equation:

$$\frac{\partial \varrho}{\partial t} = \mathcal{L}[\varrho] = -\frac{i}{\hbar}[\mathcal{H}, \varrho] + \mathcal{L}_{\mathrm{diss}}[\varrho], \tag{11.6}$$

where $H^{tot} = H^s \otimes \mathbb{I}_b + \mathbb{I}_s \otimes H^b + H^{sb}$ represents the total Hamiltonians with the piece H^{sb} describing the interaction between the system and the bath, and H in 11.6 describes the standard unitary piece of the total dynamics. All the non-unitary, dissipative aspects of the underlying dynamics are accounted for by the dissipative term $\mathcal{L}_{\mathrm{diss}}$. A general possible form of $\mathcal{L}_{\mathrm{diss}}$ is given by the so-called Kraus-like representa-

tion (in principle calculable from H^{tot}) and given by

$$\mathcal{L}_{\text{diss}}[\varrho] = \frac{1}{2}\sum_{\alpha,\beta} C^{\alpha,\beta}\mathcal{L}_{\alpha,\beta}[\varrho], \tag{11.7}$$

$$\mathcal{L}_{\alpha,\beta}[\varrho] = [F_\alpha, \varrho F_\beta^\dagger] + [F_\alpha\varrho, F_\beta^\dagger] \tag{11.8}$$

for some sequence $F_\alpha \in \mathcal{HS}(\mathfrak{h}_S)$ called Kraus operators.

Let $\mathcal{E} = \{E_\alpha\}, \alpha = 1, 2, \ldots, N$ be a system of matrices spanning the space $\mathcal{HS}(\mathfrak{h})$. Matrices forming $\mathcal{E}$ will be identified with the system of elementary errors due to the decoherence and the system will be called an errors spanning system on $\mathfrak{h}$. In a particular case when the system $\mathcal{E}$ forms a Lie algebra, it will be called a Lie algebra system.

Lemma 11.3. *Let $\mathcal{H}^{rb} \in \mathcal{HS}(\mathfrak{h}_s \otimes \mathfrak{h}_b)$ and let $\mathcal{E}$ be an error spanning system $\{E_\alpha\}$ on $\mathfrak{h}$. Then there exists a representation of the form*

$$\mathcal{H}^{rb} = \sum_{\alpha=1} E_\alpha \otimes B_\alpha \tag{11.9}$$

for some $B_\alpha \in \mathcal{HS}(\mathfrak{h}_b)$.

Proof. It is given by the application of Proposition (11.1). ■

The representation (11.3) of the interactions Hamiltonian $\mathcal{H}^{rb}$ will be called a $\mathcal{E}$-representation of $\mathcal{H}^{rb}$. Similarly, we can obtain the corresponding representation of the dissipative part of the dynamics caused by 11.7.

For this, let $\mathcal{J} = (F_\alpha)$ be the corresponding Kraus operators representing the dissipative piece of the dynamics (11.7) generator $\mathcal{L}_{\text{diss}}$.

Lemma 11.4. *For any $\mathcal{L}_{\text{diss}}$ of the form*

$$\mathcal{L}_{\text{diss}}[\varrho] = \frac{1}{2}\sum_{\alpha,\beta} C^{\alpha,\beta}[F_\alpha, \varrho F_\beta^\dagger] + [F_\alpha\varrho, F_\beta^\dagger] \tag{11.10}$$

and any error spanning system $\mathcal{E} = (E_\alpha)$, there exists a $\mathcal{E}$-representation of $\mathcal{L}_{\text{diss}}$ of the form

$$\mathcal{L}_{\text{diss}}[\varrho] = \frac{1}{2}\sum_{\alpha,\beta} J^{\alpha\beta}[E_\alpha, \varrho E_\beta^\dagger] + [E_\alpha\varrho, E_\beta^\dagger] \tag{11.11}$$

Let $\mathcal{E} = (E_\alpha)$ be an error spanning system on the space $\mathfrak{h}$. A subspace $\hat{\mathfrak{h}} \subseteq \mathfrak{h}_r$ will be called a decoherence free subspace of the dynamics (11.7) iff

(i) $\mathfrak{h}_r$ is an invariant subspace for the system $\mathcal{E}$,

(ii) $\forall_{\hat{\varrho}\in E(\hat{\mathfrak{h}})}\ \mathcal{L}_{\text{diss}}[\hat{\varrho}] = 0$.

If $\hat{\mathfrak{h}}$ is a decoherence free subspace for $(\mathcal{H}^{rb}, \mathcal{L}_{\text{diss}})$, the corresponding evolution restricted to $\hat{\mathfrak{h}}$ is purely a unitary one. So, let $\hat{\mathfrak{h}}$ be a decoherence free subspace spanned by the system of vectors $\{|c_i\rangle, i = 1, 2, \ldots, N\}$.

Lemma 11.5. *A subspace* $\hat{\mathfrak{h}} \subseteq \mathfrak{h}_r$ *is a decoherence free subspace for* $(\mathcal{H}^{rb}, \mathcal{L}_{\mathrm{diss}})$ *iff there exists anerrors spanning system* $\mathcal{E} = (E_\alpha)$ *and a spanning system of vectors* $bc^{(\hat{\mathfrak{h}})} = \{|c_i\rangle i = 1, 2, \ldots, N\}$ *such that*

$$\forall_\alpha \exists_{e_\alpha} : \forall_i E_\alpha |c_i\rangle = e_\alpha |c_i\rangle. \tag{11.12}$$

Proof. See (Lidar *et al.*, 1998). ■

Let $\mathrm{er} = (e_1, e_2, \ldots, e_{d^2-1}, \mathbb{I}_d)$ be any spanning system for $d \cdot d$ matrices, which will be called an elementary one-qudit error system.

Consider a quantum register composed on N qudits. Then the corresponding states $\mathbb{E}$ are modelled on the space $\mathcal{C}^{d^N}$ It seem to be a matter of convenience to describe the following classification of a possible decoherence process caused by the environment on the actual state of this register as given by (11.7).

11.4.1. Scenario 1 – "Total decoherence"

With the elementary one-qudit error system $er = (e_1, e_2, \ldots, e_{d^2-1}, \mathbb{I}_d)$ properly selected, we define the corresponding system $\mathcal{E}$ as

$$\mathcal{E} = er \otimes \cdots \otimes er \quad N - \text{fold tensor product.} \tag{11.13}$$

Is is easy to observe that such a system is a spanning system for $L(C^{d^N})$ and thus any error caused by the decoherence process can be registered. Such a choice provides the maximum possible complexity of possible errors generation, in which combined error from any number of qudits are generated.

Let $\pi(A)$ stand for the set of all π partitions of a given set A and let us denote $J_N = \{1, 2, \ldots, N\}$. For a given partition $\pi = (\pi_1, \ldots, \pi_k) \in \pi(J_N)$, we define $\bigotimes_{i=1}^{N} (\mathrm{er})_i = \bigotimes_{i=1}^{k} (\bigotimes_{j \in \pi_k} err_j) = \bigotimes_{i=1}^{k} \pi_i(\mathrm{err})$, where $er_i = 1 \otimes \cdots \otimes er \otimes \cdots \otimes \mathbb{I}$ and the set er is located on i^{th} factors. For further use we define $\pi_i(\mathrm{err})^{\mathrm{diag}} = \{\otimes_{j \in \pi_k}(e_\alpha), e_\alpha \in er\}$.

11.4.2. Scenario 2 – "Cluster decoherence"

For a given partition $\pi = (\pi_1, \ldots, \pi_k) \in \pi(J_N)$ we define the corresponding set of register errors:

$$\mathcal{E}(\pi) = \bigotimes_{i=1}^{k} \pi_i(\mathrm{err})^{\mathrm{diag}}. \tag{11.14}$$

This scenario corresponds to the division of the N-qudit register into k-subregisters selected by π in which decoherence takes place in such a way that the same collective error of the form $\otimes e_i$ is produced but the subregisters are decohering independently.

In particular, if $k = N$, this corresponds to Scenario 1 and in opposition to this case, when $k = 1$ we have only d^2 errors in $\mathcal{E}$. For a general case $1 < k < N$ we have the allowed error system $\mathcal{E}$ composed of $(d^2)^k$ errors.

As is well known the notion of decoherence free subspaces plays a significant role in the construction of several quantum error correction codes for qubits and can be applied also to secure the transmission of the quantum key from quantum noise effects (Walton *et al.*, 2003). As we have seen this line of reasoning can be extended without any serious problems to deal with the case of qudits manipulation. Here we present some examples for $d = 3, 4$ showing a manifest similarity to the case of $d = 2$.

Example 11.1. The case $d = 3$ corresponds to the so-called qutrits. A convenient simple errors generating system can be given as the system composed of eight Gell-Mann matrices $(\lambda_i), i = 1, \ldots, 8$ and the matrix $\lambda = \mathbb{I}_3$. The matrices λ_j are given explicitly as

$$\lambda_1 = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \lambda_2 = \begin{bmatrix} 0 & -i & 0 \\ i & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \quad \lambda_3 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 0 \end{bmatrix},$$

$$\lambda_4 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix}, \quad \lambda_5 = \begin{bmatrix} 0 & 0 & -i \\ 0 & 0 & 0 \\ i & 0 & 0 \end{bmatrix}, \quad \lambda_6 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix},$$

$$\lambda_7 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -i \\ 0 & i & 0 \end{bmatrix}, \quad \lambda_8 = \begin{bmatrix} \frac{1}{\sqrt{3}} & 0 & 0 \\ 0 & \frac{1}{\sqrt{3}} & 0 \\ 0 & 0 & -\frac{2}{\sqrt{3}} \end{bmatrix}.$$

The following identities are very helpful for the construction of several errors correcting codes:

$$\begin{aligned} \operatorname{Tr} \lambda_i \lambda_j &= 2\delta_{i,j} \qquad i, j = 1, \ldots, 8 \\ \operatorname{Tr} \lambda_i &= 0 \qquad i = 1, \ldots, 8 \\ [\lambda_i, \lambda_j] &= 2i \sum_k f_{ijk} \lambda_k, \end{aligned} \tag{11.15}$$

where f_{ijk} are real, antisymmetric structure constants of semisimple Lie algebra su(3). In particular, from the semisimplicity of su(3) it follows that the allowed value of constant defining, as in Lemma (11.5), the corresponding decoherence free subspace is equal to zero. ♦

Example 11.2. In the case of $d = 4$ corresponding to the so-called ququats a very convenient choice of the simple errors spanning system is that provided by 16 Hermitean matrices $\tau_i, i = 1, \ldots, 16$ also known under the name of Dirac matrices. Defining as $\sigma_i(2), i = 1, 2, 3$ the corresponding Pauli matrices of dimension (2×2), we can construct the following (4×4) matrices:

$$\sigma_i(4) = 2\mathbb{I}_2 \otimes \sigma_i(2), \tag{11.16}$$

and then

$$\rho_i(4) = 2\sigma_i(2) \otimes \mathbb{I}_2 \tag{11.17}$$

for $i = 1, 2, 3$. Then the system $\tau = (\tau_i), i = 1, \ldots, 16$ composed of 16 matrices τ_i and defined by the equality $\tau = (\mathbb{I}_4, \sigma_i(4), \rho_i(4), \sigma_i\rho_k)$ for $i, k = 1, 2, 3$ forms a basis for the space of 4×4 matrices with complex coefficients and obeys, the following identities:

$$\text{Tr } \tau_i\tau_j = 4\delta_{i,j} \tag{11.18}$$

and the su(4) Lie algebra relations

$$[\tau_i, \tau_j] = 2i\epsilon_{ikl}\rho_l, \tag{11.19}$$

where ϵ_{ikl} is a totally antisimmetric symbol with normalisation $\epsilon_{123} = +1$. ♦

Several error correction codes could be described in a strong similarity to the known codes in the case of qubits. For some additional material in the case of qudits see, e.g. (Aharonov and Ben-Or, 1996; Ashikmin and Knill, 2001; Gassl *et al.*, 2003; Gottesman, 1998; Rains, 1999).

11.5. Quantum cryptography protocols, their security and technological implementations

In fact, classical cryptography provides one unbreakable cipher which resists even opponents equipped with unlimited computational power (Schneier, 2002). This algorithm is called the Vernam cipher (Vernam, 1926) and was invented in 1917 by an AT&T engineer, Gilbert Vernam. This algorithm is a symmetric cipher and it offers unconditional security, but only when the following requirements are satisfied:

i) the key must be at least as long as the message;

ii) the key must be generated randomly (not pseudo-randomly, so that all pseudo-random generators cannot be used);

iii) the key must be used only once;

iv) the key must be delivered to the recipient using a secure channel in order to make impossible for non-authorized person to intercept the key.

These requirements were proved by Claude E. Shannon, who laid the foundations of communication theory in 1949. If any of these are not fulfilled, the system can be easily broken. The main drawbacks of the Vernam cipher are the points 2) and 4), which cannot be satisfied using only classical cryptography. Fortunately, quantum mechanics comes in handy and offers a solution – quantum cryptography, which provides a method to generate purely random keys and, furthermore, is also capable of exchanging the generated keys safely, which solves the point 4). This rather new technology is called QKD (Quantum Key Distribution).

The main problem of classical cryptosystems is the security of keys distribution. The security of classical cryptographic methods cannot be guaranteed as there are no mathematical proofs of the security of these algorithms, so that advances in technology and mathematical algorithms can introduce security holes. The quantum approach can provide unconditional security. The principle of quantum cryptography consists in the use of non-orthogonal quantum states or entanglement. Its security is guaranteed

by the Heisenberg uncertainty principle, which does not allow discriminating non-orthogonal states with certainty and without disturbing the measured system. This means that it is impossible to eavesdrop without being detected because, in contrast to all classical signals that can be monitored passively, quantum information cannot be acquired (copied, read or altered in any way) without affecting the state of the object that it has been encoded in. It is very important to see that quantum mechanics does not disallow eavesdropping, it only enables to detect the presence of an eavesdropper. If only the cryptographic key is transmitted, no information leak can take place, even when somebody attempts to read the transmission. In this situation, QKD protocols discards the key and the users start the key exchange procedure over to generate and exchange a new key.

In general, there are two types of QKD algorithms. The first one is based on entanglement. The most popular protocol here is the Ekert protocol, called E91, which, in short, is based on producing pairs of entangled qubits and sending one qubit of each pair to the receiver. Any eavesdropping will require measurement, which destroys entanglement and allows detecting it. The security of the original proposal was ensured by checking the violation of Bell's inequalities (Ekert, 1991). The original version of the Ekert protocol is not used; instead, there is a simplified version of Ekert's protocol, proposed by Bennett (1992). This simplified version of Ekert's protocol works in a very similar way as the BB84 protocol (which will be discussed later). The parties choose only from two bases corresponding to two perpendicular orientations of their spin-measurement devices. The only difference from BB84 is that the sender does not send particles in a chosen spin (or polarization) state, but he/she measures his/her particle from the entangled pair in one of two conjugated bases. He/she must select the bases randomly and independently from the receiver. The rest is the same as in BB84: after the transmission they both compare their bases and keep only those results when they used the same bases.

This type of QKD algorithms is rarely used in practice, so this work is focused on the second type of QKD algorithms, based on non-orthogonal quantum states. This type of QKD algorithms was initiated by Charles Bennett and Gilles Brassard in 1984, when they came up with idea of how to use quantum mechanics to securely distribute a random cryptographic key. Bennett and Brassard presented a protocol that allows two parties to establish an identical and purely random sequence of bits at two different locations, while allowing to reveal any eavesdropping with a very high probability. The protocol is now called BB84. In todays' implementations, information is encoded into polarization states of individual quanta of light – photons; and, to be specific, four photon polarizations, as shown in Fig. 11.1:

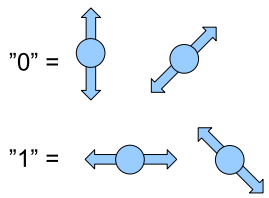


Fig. 11.1. Four photon polarizations and their classical representation

The BB84 protocol also uses two conjugated bases. One basis can consist, e.g. of horizontal and vertical polarization states of photons and is called rectilinear. The other basis, called diagonal, consists of states of linear polarizations at 45° (anti-diagonal) and 135° (diagonal). To establish a secure key, the sender of the message encodes logical zeros and ones into two orthogonal states of a quantum system, but for each bit he/she randomly chooses one of the two bases. Then he/she sends the encoded bits through the quantum communication channel. The receiver of the message measures the gained data — he/she does not know which base the sender used, so that he randomly chooses which basis should be used to make the measurement. If he/she is right and measures using correct basis, he/she will acquire correct data that represents classical bits (this is always the case) – Fig. 11.2.

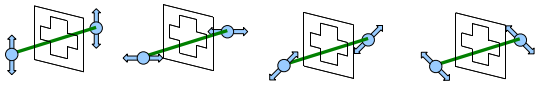


Fig. 11.2. Photon measured using the right basis

On the other hand, if the selection of the basis is wrong, the outcome from the measurement produces random outcomes with equal probabilities. Furthermore, the measurement effect cannot be reversed, so that the information is lost, as shown in Fig. 11.3.

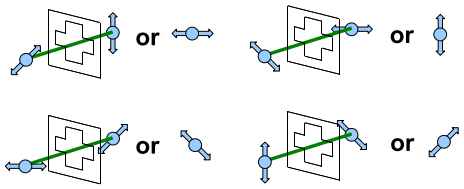


Fig. 11.3. Photon's polarization change while measuring using the wrong basis

Statistically, the sender's and receiver's bases coincide in 50% of cases. This means that in 50% of cases the receiver's measurements provide deterministic outcome and agree with the sender's bits. The parties must learn which measurements are bad and which are wrong, so that they need an auxiliary, classical, public communication channel to tell each other what basis was used for each transmitted and measured photon. When the bases coincide, the parties keep the bit, otherwise they discard it. The kept photons are then treated as classical bits and make a secret key. This procedure is called key sifting.

If there is an eavesdropper on the line, he/she can easily be detected. To get information, the eavesdropper must measure photons, but he/she is in a situation similar to that of the receiver's – he/she does not know which basis was used by the sender, so that he/she must randomly choose one. This means that the error rate will increase (statistically by 25%).

The described scheme is of course applicable only in ideal conditions. In reality there are many factors that must be taken into consideration, like quantum errors

(the quantum line is not ideal and introduces errors while transmitting photons). This forces checking the key for errors after the key sifting procedure and is called key distillation, which consists of two steps: the first one corrects key errors using the classical error correction algorithm (like BCH or Reed-Solomon codes – (Vernam, 1926)) to precisely define the error level. The key distillation procedure usually requires the parties to reveal part of the key (usually half of it) to calculate the number of different bits and estimate the error level in the rest of the key. If the error level is high, the whole QKD protocol must be started over and the key is cancelled, because this means that there may be an eavesdropper on the quantum channel. The key distillation procedure also introduces user authentication, which prevents the popular "man-in-the-middle" attack, in which the eavesdropper pretends to be a sender for the receiver and to be a receiver for the sender (so that the parties do not even know that they are both communicating with the eavesdropper only). All QKD protocols assume that the quantum channel is authenticated, so that the eavesdropper may only read data, but not alter it.

There is one more additional step that requires a key, established earlier by the sender and the receiver. This key is used to authenticate all the communication on classical channel and serves only to authenticate the first session. After every session it is replaced by part of the secret key that is being exchanged.

The next step is called privacy amplification. It relies on key compression to reduce the eavesdropper's information about it below some threshold. The privacy amplification procedure works only to a given error threshold; if it is higher, the whole key exchange procedure must be restarted from scratch because the eavesdropper may have too much information about the key. The simplest privacy amplification procedure may be shown as follows: the parties assume that the shared key should be divided into two-bit parts and the eavesdropper knows only one bit from the part but does not know the other bit. Then a sum without carry (XOR) of all the parts should be calculated and the new key should be built from the resulting bits. For example, if one of the parts was 01, the result will be $0 + 1 = 1$. If the eavesdropper does not know one of the bits, he/she knows nothing about the result, so that his/her knowledge about the key is decreased. Unfortunately, this solution shortens the final key by 50%, and it is safer to extend the parts to more than two bits, which cause larger key reduction. There are of course better privacy amplification procedures that utilize classical correction codes – the sender publishes only the parity bits of parts of the key. Then the receiver compares these values with his/her and is capable of finding and then correcting errors. To increase security, the parties may agree upon random permutation and use it after the final key is acquired (Pawłowski and Gielerak, 2006).

Besides BB84, there are a few more QKD protocols. One of them is SARG, called after the names of its authors, proposed to beat the Photon-Number Splitting attack (PNS) in QKD schemes based on weak laser pulses. In the photon-number splitting attack, the eavesdropper exploits multi-photon states present in weak laser pulses. This is usually the case because todays' photon sending machines are imperfect and often send a few photons in one pulse, although there should be only one photon sent. The protocol relies on the eavesdropper's inability to perfectly distinguish between two non-orthogonal states (Scarani *et al.*, 2004) and requires the same hardware as for BB84; only the classical communication between parties is modified. In contrast to

BB84, two values of a classical bit are encoded into pairs of non-orthogonal states. The sender prepares four quantum states and the receiver makes measurements exactly as in the BB84 protocol. But here the sender does not reveal the basis but the pair of non-orthogonal signal states such that one of these states is the one he/she has sent. The receiver guesses correctly the bit if he/she finds a state orthogonal to one of two announced non-orthogonal states (Scarani *et al.*, 2004). In comparison with the BB84 protocol, SARG enables to increase the secure QKD radius when the source is not a single-photon source.

There is a proof that BB84 is secure and is capable of detecting any eavesdropper or attack (Shor and Preskill, 2000). However, this proof has been carried out only for ideal conditions, and in practice one should take into consideration the influence of external disturbances, which increas along with the distance of the signal travel distance. This means that many of the transmitted photons should be cancelled because they are distorted during the transmission. The problem is to distinguish whether the disturbances are caused by the eavesdropper or just by channel imperfections. QKD protocols assume that when the level of errors is high, the key must be retransmitted. This allows launching succesful DoS (Denial of Service) attack — it is enough for the attacker to gain passive access to the quantum transmission line and just measure the signals without checking the results. There are no results regarding an attack using the Buzek-Hillery optimal copying machine, which is capable of making approximate copies of transmitted qubits, although this attack can probably can be detected because the original states are also changed by that machine, but the attacker will have some information about the key.

Nowadays hardware implementations of QKD protocols can be bought. There are mainly two producers: Id Quantique from Switzerland and MagiQ from USA. Both of them offer very similar hardware that implements quantum BB84 or SARG protocols and the AES (Advanced Encryption Standard) algorithm with the key length of 128, 192 and 256 bits. The AES is used to do encryption, and its key is constantly changing. This approach unfortunately does not offer unconditional security. The reason is that decoherence (photons' entanglement with environment and themselves) reduces transfer rates of quantum data to a fairly low level (Id Quantique says about 1500 bits per second), which is not enough to utilize the Vernam cipher but enough for any classical symmetric algorithm, like the AES, with a short key that may be changed to increase security when the quantum channel provides enough portion of data. Unlike in classical networks, there is another problem – transmission distances, which are rather short. Current products offer distances up to 100 km, which in many cases are not very satisfactory, the newest techniques may achieve even 270 km, which still may not be enough. Obviously the distance limit is caused by the impossibility to introduce quantum repeaters (because of the no-cloning theorem).

Summing up, quantum cryptography technology is only beginning; nevertheless, it can be seen that it offers a secure quantum key distribution protocol that can detect any eavesdropping. As the technology will be developed, the Vernam cipher surely will be utilised, offering unconditional security. There is still much to be done – the decoherence effects must be eliminated, the photon transmission distance must be extended, the security proof of QKD protocols security in real environment should be introduced – but even today quantum cryptography opens a new era in security.

11.6. Quantum computer simulator and its applications

In the paper (R.P. Feynman 1982) it is argued that classical computers never permit a simulation of a quantum system in polynomial time. Feynman's remark can be formalised by the formulation of the following lemma:

Lemma 11.6. *Let us assume that there exists an algebra which permits effective representation (with compression) of the quantum register with entanglement for n qudits, where the symbol $\widetilde{\otimes}$ denotes special operators which allow composing several qudits and preserve the entanglement if exists between the qudits:*

$$|\psi_0\rangle\widetilde{\otimes}|\psi_1\rangle\widetilde{\otimes}|\psi_2\rangle\widetilde{\otimes}\ldots\widetilde{\otimes}|\psi_{n-1}\rangle = |\psi\rangle.$$

Nevertheless, the compressed vector still contains an exponential amount of information about the state. Any extraction from $|\psi\rangle$ of the classical information process requires at least $O(d^n)$ operations.

Proof. A sketch of the proof is the following: let T stand for the classical Turing machine. On the tape of T the state vector for d-level qudits (even with compression) is written and its representation contains n fields of the tape. Without compression the tape has length equal to d^n complex values. In the measurement process of the whole register, which is assumed to be not in homogenous superposition of the basic states, the register state collapses to one of base states. It is necessary to look through all states to find the one with a maximal modulus of the probability amplitude coefficient and this search requires again at least $O(d^n)$ comparison operations to be performed. ■

Thus, quantum computations simulators are the only widely available tools for testing quantum algorithms known today. Unfortunately, the simulations can be executed only for small registers which contain 20–26 qubits on typical PC hardware. What is even worse, the physical experiments are very costly and difficult to implement. Therefore, simulators of quantum computation are very important especially for educational purposes and for researches who want to make some numerical experiments, for example, with different types of gates for an actually developed quantum circuit.

A quantum computing simulator (Sawerwain, 2005; Sawerwain, 2006) has under development at University of Zielona Góra since 2005. Several open source tools are used: Gnu Compiler Collection v4.x, the Python script language v2.4.x, the SWIG 1.3.29 wrapper generator, the specialised LAPACK and BLAS linear algebra library and the MPI library for parallel implementation.

The core library of the Quantum Computer Simulator (QCS) system is written in a pure ANSI C programming language which allows making ports for many other platforms. The main port of QCS is prepared in the Python language but a port for Java also exists. Yet for the MPI version a simple language similiar to the classical assembler *qasm* (quantum assembler) is prepared. This language makes the script executing in parallel environment easier. Generally, the QCS works mainly in the Windows and Linux operating systems in 32 and 64 bits environment.

Figure 11.4 depicts elementary script in the Python language. The QCS port in Python has two important properties. First, the port allows making scripts which are

easily executed by the Python interpreter. The second advantage is the possibility of interactive work from the console. The user can execute instruction by instruction and instantly checks the actual state of the quantum register.

```
import qcs                    | r=qcs.QubitReg(4)
def makePsiPlus(r):           | r.Reset()
    r.SetKet("0000")          | makePsiPlus(r)
    r.HadN(0)                 | r.Pr()
    r.CNot(0,1)               | del r
```

Fig. 11.4. Small script in the Python language which produces one of the maximally entangled states – the so-called Bell state

The QCS system allows working with several types of quantum computation models. At present, the following features are implemented:

- PQCs – pseudo Quantum Circuits, which consist of two gates only: not and cnot
- CHP – quantum circuits composed of the following gates: cnot, hadamard, phase change and single qubit measurements
- QCM – standard Quantum Circuit Model (a universal library of gates is allowed)
- density matrices and several standard matrices tools like fidelity (and some others metrics for quantum states), eigenvector and eigenvalue functions and the Schmidt decomposition are implemented

We are planning to enrich the future version of our simulator with a one-way quantum computation model which is based exclusively on teleportation and measuring operations.

Table 11.1 depicts time needed for the execution of special calculations for a simple example where only Hadamard gates are applied to the quantum register. The implementation of QCS is very effective but the total amount of operations in the process of applying the one-qubit gate to the register containing n qubits takes $\mathcal{O}(2^n)$ steps. The result confirms this relation, because adding one qubit to the register causes the doubling of time necessary for script executing.

Table 11.1. Time interval (measured in seconds) necessary to implement the following calculational process: step one – set the initial state of a register (composed of n-qubits) to state $|0\rangle$, step two – apply ten hadamard gates to randomly chosen ten qubits of the register

size of register	10	20	21	22	23	24	25
time	$\approx$ 0.0s	1.5s	3.1s	6.5s	13.5s	28.2s	58.7s

11.7. Summary and conclusions

There is no doubt that many scientific and technological breakthrough are still necessary in order to construct a large scale quantum computing machine. It is our hope that we were able to describe the main ideas of the recently emerging field of quantum computations. That the project of constructing a quantum computer is one of the greatest challenges of the present day's science and technology one can prove simply by displaying the huge list of active institutions, all over the world, involved into this project and the number of scientific projects carried out there. However, despite so many efforts we are still far from a satisfactory solution.

In the present contribution we pointed a few lines of research presently carried out by the Q-INFO group, located at the Institute of Control and Computation Engineering at the University of Zielona Góra.

Although most of our scientific projects are at a preliminary stage of development, we presented at least some introductory remarks on them. In particular, the idea of using labelled transition systems for the analysis of quantum programs seems to be novel in certain sense. However, still many specific concepts developed in the classical theory are waiting for their deep elaboration in the quantum regime. The decoherence problem seems to be one of the main obstacles in constructing a large-scale quantum machine and this is why we have concentrated our presentation in the area of fault-tolerant quantum computing schemes focusing the main effort on the algebraic aspects of quantum error correction codes in Section 11.4.

It seems that the only real, in present-day applications of quantum technologies, is the area of secure communications, and this is the topic which was exclusively reviewed in the present contribution from the point of view of conceptual and technological aspects as well. The main concept of our discussion in Section 11.5 was focused on attempts to explain why the key transfer in the quantum channel offers absolute security of the communication. And it is our hope that we were able to convince our reader that the unavoidable forthcoming era of quantum information processing and communications offers us completely new solutions of many problems that we are fighting permanently in present days.

References

Aharonov D., van Dam W., Kempe J., Langau Z. and Reger O. (2004): *Adiabatic quantum computation is equivalent to standard quantum computation.* — Proc. 45-th Annual IEEE Symp. *Foundation of Computer Sciences*, Rome, Italy, pp. 42–51.

Aharonov D. and Ben-Or M., *Fault tolerant quantum computations with constant error.* — In: `www.arxiv.org/quant-ph/9611025`.

Ashikmin A. and Knill E. (2001): *Nonbinary quantum stabilizer codes.* — IEEE Trans. Inform. Theory, Vol. 47, pp. 3065–3072.

Bennett C.H., Brassard G. and Mermin N.D. (1992): *Quantum cryptography without Bell's theorem.* — Phys. Rev. Lett., Vol. 68, pp. 557–559.

Calderbank A.R., Rains E.M., Shor P.W. and Sloane N.J. (1997): *Quantum error correction and orthogonal geometry.* — Phys. Rev. Lett., Vol. 78, pp. 405–408.

Ekert A. (1991): *Quantum cryptography based on Bell's theorem*. — Phys. Rev. Lett., Vol. 67, pp. 661–663.

Feynman R.P. (1982): *Simulating physics with computers*. — Int. J. Theoretical Physics, Vol. 21, Nos. 6/7., pp. 467–488.

Gassl A., Rötteler A. and Beth T. (2003): *Efficient quantum circuits for non-qubit quantum error-correcting codes*. — Int. J. Foundations of Computer Science, Vol. 14, No. 5, pp. 757–775.

Gottesman D. (1998): *Fault-tolerant quantum computation with higher-dimensional systems*. — Lect. Notes in Comp. Science, Vol. 1509, pp. 302–313.

Gottesman D. and Lomonaco S.L. Jr (Ed.) (2002): *An introduction to quantum error correction in quantum computation*. — Proc. Symp. *Applied Mathematics*, American Mathematical Society, AMS, Providence, Rhode Island, Vol. 58, pp. 221–225.

Hirvensalo M. (2001): *Quantum Computing*. — Berlin Heidelberg: Springer-Verlag.

Kitayev A.Y. (2003): *Fault tolerant computations with anyons*. — Annals Phys., Vol. 303, pp. 2–30.

Lidar D.A., Chuang I.L. and Whaley K.B. (1998): *Decoherence free subspaces for quantum computations*. — Phys. Rev. Lett., Vol. 81, pp. 2594–2597.

Pawłowski K. and Gielerak R. (2006): *Security of key transmission*, In: New Technologies in Computer Networks, (Węgrzyn S., Znamirowski L., Czachórski T., Kozielski S., Eds.), Vol. 2, pp. 375–384. — Warsaw: Wydawnictwa Komunikacji i Łączności, WKiŁ, (in Polish).

Peres A. (1995): *Quantum Theory: Concepts and Methods*. — Dordrecht: Kluwer Academic Publishers.

Plotkin G.D. (2004a): *A structural approach to operational semantics*. — J. Logic and Algebraic Programming, Vol. 60, pp. 17–139.

Preskill J., Lo H.K., Papesku S. and Spiller T., (Ed.) (1998): *Fault Tolerant Quantum Computations*. — Singapore: World Scientific, pp. 213–269.

Rains E.M. (1999): *Nonbinary quantum codes*. — IEEE Trans. Information Theory, Vol. 45, pp. 1827–1832.

Sawerwain M., Gielerak R. and Pilecki J.: *Some applications of quantum labelled transition systems to quantum programming languages and algorithms*. — Int. J. Quantum Information, (submitted).

Sawerwain M., Gielerak R. and Pilecki J. (2006): *Operational semantics for quantum computation*. — In: New Technologies in Computer Networks, (Węgrzyn S., Znamirowski L., Czachórski T., Kozielski S., Eds.), Vol. 1, pp. 69–77. — Warsaw: Wydawnictwa Komunikacji i Łączności, WKiŁ, (in Polish).

Sawerwain M. (2005): *Quantum Computing Simulator*. — Proc. Int. Conf. *Computer Methods and Systems*, CMS '05, Kraków, Poland, Vol. 2, pp. 185–190, (in Polish).

Sawerwain M. (2006): *Parallel implementation of quantum computing Simulator*. — Proc. 14th Conf. *Computer Newtorks*, Łódź, Poland, pp. 241–244, (in Polish).

Scarani V., Acin A., Ribordy G. and Gisin N. (2004): *Quantum cryptography protocol robust against photon number splitting attacks for weak laser pulse implementations*. — Phys. Rev. Lett., Vol. 92, 057901.

Schneier B. (2002): *Applied Cryptography*. — Warsaw: Wydawnictwa Naukowo-Techniczne, WNT, (in Polish).

Shor P.W. (1995): *Scheme for reducing decoherence in quantum computer memory.* — Phys. Rev. A, Vol. 52, pp. 2493–2496.

Shor P.W. and Preskill J. (2000): *Simple proof of security of the BB84 quantum key distribution protocol.* — Phys. Rev. Lett., Vol. 85, pp. 441–444.

Terhal B.M and Horodecki P. (2004): *A Schmidt number for density matrices.* — Phys. Rev. A, Vol. 61, No. 4, 040301.

Walton Z.D., Abauraddy A.F., Sergienko A.V., Salen B.E.A. and Teich M.C. (2003): *Decoherence free subspaces in quantum key distribution in quantum key distributions.* — Phys. Rev. Lett., Vol. 91, No. 8, 087901-4.

Vedral V. (2003): *Geometric phases and topological quantum computation.* — Int. J. Quantum Information, Vol. 1, No. 1, pp. 1–23.

Vernam G. (1926): *Cipher printing telegraph systems for secret wire and radio telegraphic communications.* — J. IEEE, Vol. 55, pp. 109–115.

Chapter 12

SELECTED METHODS OF DIGITAL IMAGE ANALYSIS AND IDENTIFICATION FOR THE PURPOSES OF COMPUTER GRAPHICS

Sławomir NIKIEL*, Piotr STEĆ*

12.1. Introduction

Digital images are a common part of business presentations, communication systems, user interfaces. They form the basis of digital media broadcasting systems. Image analysis and identification are key elements in image processing leading to image understanding. Two sub-chapters present recent research results in the fields of static image analysis and digital video classification/segmentation for compression and storage.

Museums, galleries and other cultural heritage institutions process large amounts of digital data representing their collections. Multimedia data can be exploited to create more engaging learning experience to on-site and remote visitors. This presents a series of challenges to researchers and developers of systems and tools for digital cultural heritage stakeholders. New areas of spatial imaging (including auto-stereoscopic displays, virtual and mixed realities and pervasive gaming) are evolving rapidly, notably in the field of 3D digitization and virtual representation of artifacts. This chapter presents recent advances in research into and development of tools designed to meet one of those challenges, especially in the area of rapid prototyping of virtual models. Photogrammetric reconstruction of architectural artifacts plays an important role in digital archaeology, where spatial properties of the object need to be specified 'en situ' and should be open for several changes along with new hypotheses on the excavation. The data are usually obtained with a number of analog and digital cameras. Every model of digital or analog camera has a lens system with geometrical distortions, which causes some degree of distortion of the output image. Since photographs

* Institute of Control and Computation Engineering
e-mails: {S.Nikiel, P.Stec}@issi.uz.zgora.pl

are used for many purposes: measuring, texturing, image registration and stitching, it is required to compensate distortion introduced by the image acquiring system (optical lens system). In most cases cameras are calibrated in specialized labs with expensive customized devices. We propose a method of correction in a simple way – a survey method, performed by a single user, where results are sent to a server which performs auto-calibration and collects calibration data (profiles). To determine the coefficients of the calibration procedure we use evolutionary algorithms. This allows users to download correction parameters for their cameras afterwards. The prototype application based on the corrected images helped with rapid spatial reconstruction of architectural artifacts. This tool was designed to quickly re-create accurate virtual models, where the user can study spatial properties of the reconstructed architecture. The prototype was effectively used to quickly prototype several models of 17th and 18th century architecture including the Qasr Al-Kharaneh castle (cooperation with the Queen Rania University, Jordan).

Another subject covered in this chapter is video segmentation. A lack of efficient real-time segmentation techniques is a significant obstacle in the wide-spreading and practical applications of MPEG-4 object-based compression. This object-based approach gives the possibility of manipulating the transmitted objects. A video object can be easily replaced by another one and transmitted using different bandwidths depending on the required picture quality (e.g., less important objects can be transmitted with a lower bitrate). To take advantage of these features, the video sequence must be precisely segmented into objects. MPEG-4 and MPEG-7 do not specify how to extract objects. The only requirement is that the objects must be defined prior to coding or indexing.

In most cases, video sequence segmentation is aimed at the extraction of the so-called semantic objects that have meaning for humans (e.g., man, car, table, etc.), but are very hard to define using low-level features. Only such features can be extracted directly from a video sequence. It makes the process of the extraction of such objects from a highly textured back-ground very complicated.

The desired quality of segmentation and the speed of the process of segmentation depend on the application. For off-line video editing applications, the most important issue is quality, while speed is a secondary matter. On the contrary, in many real-time applications such as vehicle navigation and surveillance, the most important feature is the detection and location of moving objects, while the determination of the exact object boundary is not so important. A method that would be fully automatic, fast and accurate is still an open problem in video segmentation.

The method proposed in Section 12.2 is intended for the preparation of the source material for object-based video processing and requires a high quality input, i.e., no compression artefacts, a low noise level, etc. For the sake of simplicity, the deliberations are restricted to non-interlaced (progressive) sequences only. The method is able to work in the presence of moving background without global motion compensation. Additionally, multiple objects can be detected at the same time. In this implementation only the sequences with translational and rigid motion can be segmented. A modified version of the multilabel fast marching algorithm is used in the segmentation process. The most important modification is the possibility of segment merging as well as pushing back the borders of other segments. Thanks to this, the limitation

of one-way propagation for fast marching is removed. The side effect of the algorithm are regularized seed regions overlaid on the original frame from motion field.

12.2. Complex solution to the lens distortion problem in photogrammetric reconstruction for digital archaeology

12.2.1. Basic concepts

Cultural heritage consists of countless monuments that are often forgotten. A number of techniques have been developed to preserve and distribute information on them (Nikiel, 2000a; 2000b; 2004; 2007; Nikiel and Steć, 1998). One of these methods is to store digitally reconstructed models of monuments with the possibility of their further display. The commonly used techniques (i.e., laser scanning) preserve detailed spatial information about given objects. Object geometry can be acquired with the help of various scanning techniques relying on telemetric and remote sensing. The acquired volumetric content needs further processing and mesh optimization (Nikiel and Goiński, 2005). The problem is how to choose a method that can be more efficient and better for archeological reconstruction purposes. This is particularly important for virtual reconstructions with the environmental context including the terrain (Nikiel, 2004; Nikiel and Steć, 2000). The usage of laser beam or automatic photogrammetric systems gives in the output a large amount of data, which causes problems with the storage and display of the reconstructed models and further processing of texture images (with no automatic selection of objects and the level of detail assigned to them) (Nikiel and Stachera, 2004; Nikiel *et al.*, 2001). Another way is to take some shots of the target objects – but static photographs are not explanatory, they give little or no spatial information. Sometimes it is better to rely simply on manual photogrammetric reconstructions (Nikiel, 2001; 2002; 2003). Manual reconstructions exploit a great thing – human knowledge that is used in the reconstruction process. The user decides what details are to be reconstructed and can personally supervise the modeling process. Of course, it involves some difficulties like analysis and manual processing of the data. Typically photogrammetric reconstructions rely on design from perspective photographs. Expensive photogrammetric cameras are used, although it is possible to use amateur cameras to obtain high-resolution images. Results not always are good – sometimes the user has to correct structure of the model. Photogrammetric methods include reconstruction from orthophotographs, that are processed perspective photographs. In this chapter we focus on selected modeling technologies. The paper describes the correction methods and provides tips how to reconstruct a spatial model from even poor quality materials in order to get acceptable results.

12.2.2. Modeling based on orthogonal projection

Modeling based on orthogonal views is one of the ways to visualize artifacts for archeological purposes. Modeling from orthogonal views relies on three fundamental phases:

- processing data, the production of orthophotos from perspective photographs;
- calibration of virtual orthogonal cameras;
- main reconstruction of the object.

Orthogonal/parallel views are very helpful in modeling (calculations in the reconstruction process are simpler than for perspective photographs) but require the preprocessing phase, which consists of:

- removal of lens distortion from perspective photographs,
- removal of perspective with the use of the perspective correction algorithm.

For blueprints (technical drafts/orthogonal views), the processing is limited only to the correction (rotation) of the blueprint.

12.2.3. Image correction

Photographs taken with amateur cameras show the lens distortion effect. In most cases, the pictures are emphasized in the central part of the image; distortions can be found also in the corners of the image. Lens distortions make the image look like a barrel or a pincushion. In order to remove this correction, every pixel of the image must be recalculated. The lens distortion model is usually based on polynomial equations (Basu and Licardie, 1995; Cucchiara *et al.*, 2003; Karras and Mavrommati, 2001; Karras *et al.*, 1998; Krauss, 1997; Quan, 1996). This means that there are three or more coefficients that define radial image expression. A sample non-polynomial approach is discussed in (Karras *et al.*, 1998).

The image is transformed geometrically into the corrected one with the use of polynomial calculations. This work exploits the model proposed in (Kupaj, 2005; Nikiel and Kupaj, 2004):

$$\begin{aligned} x_{src} &= x_{dst} - (x_{dst} - c_x)\left[r_0 + r_1(r_d\ d_w)^2 + r_2(r_d\ d_w)^4\right] \\ y_{src} &= y_{dst} - (y_{dst} - c_y)\left[r_0 + r_1(r_d\ d_w)^2 + r_2(r_d\ d_w)^4\right] \end{aligned}, \tag{12.1}$$

where

$$c_x = \left(\frac{W}{2} + \Delta x_0\right), \quad c_y = \left(\frac{H}{2} + \Delta y_0\right),$$

$$r_d = \sqrt{(x_{dst} - c_x)^2 + (y_{dst} - c_y)^2}$$

and

$$d_w = \frac{2000}{\sqrt{W^2 + H^2}}.$$

The symbols denote the following:

x_{src}, y_{src}	source image coordinates
x_{dst}, y_{dst}	destination image coordinates
c_x, c_y	transformation center
r_d	radial distance (from the point to the transformation center)
d_w	radial distance scaling factor
r_0, r_1, r_2	internal distortion correction coefficients
$W \times H$	source/destination image size [width × height]

Because the coefficients r_0, r_1, r_2 are non-linear and not normalized, they should be calculated from the normalized parameters k_0, k_1, k_2 (percentage factors):

$$r_0 = \frac{k_0}{100}, \quad r_1 = \mathrm{sgn}(k_1)10^{-8+\left|\frac{k_1}{50}\right|}, \quad r_2 = \mathrm{sgn}(k_2)10^{-14+\frac{k_2}{50}}, \tag{12.2}$$

where x_{src} and x_{dst} are coordinates on the distorted image and coordinates on the equalized image.

The Hough transform (called HTRDC) can be used to calibrate the image (Cucchiara *et al.*, 2003). This method facilitates automatic extraction and calibration of data with the Canny's edge detector. There are other methods to calibrate distortion. The estimation of distortion from the curvature of straight lines or regular grids is given in (Karras and Mavrommati, 2001). The regular the grid of points is needed to perform the calibration process. The photographed calibration plane is filtered and then grid recognition process is performed. The output set of points is used to conduct further processing. Next, four vertices in the grid corner are used to make an undistorted grid. The undistorted grid is necessary for further calculations and through the corner vertices it is possible to create one.

Next, the undistorted grid is subsequently corrected (transformed) and compared with the distorted one. The process is repeated until a given difference is minimal. As a result of this process, correction coefficients are obtained. The coefficients are arguments for inverse transformation used in image correction. The right set of coefficients must be calculated. It is a minimalization problem where the difference between the grid must be minimal. For the error function denoted as F_{err},

$$F_{err}(R, S^G) = \sum_{i=1}^{n} \sqrt{\left(\frac{R_i X - S_i^G X}{X}\right)^2 + \left(\frac{R_i Y - S_i^G Y}{Y}\right)^2}, \tag{12.3}$$

where R is the undistorted grid and S^G identifies the undistorted grid after inverse transformation.

Evolutionary algorithms can be used to determine the coefficients. The parameters should be selected by experiments to get applicable convergence, stability and short time of execution (Kupaj, 2005; Nikiel, 2003). The next correction process is the perspective to be removed. Also the tilted photograph must be corrected in order to get a straight image that looks like an orthogonal view.

The work (Criminisi *et al.*, 1999) describes the following camera model that might be used to remove the effect of perspective:

$$\begin{bmatrix} XW \\ YW \\ W \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \tag{12.4}$$

which can be developed into the equation

$$X = \frac{ax + by + c}{gx + hy + 1}, \quad Y = \frac{dx + ey + f}{gx + hy + 1}. \tag{12.5}$$

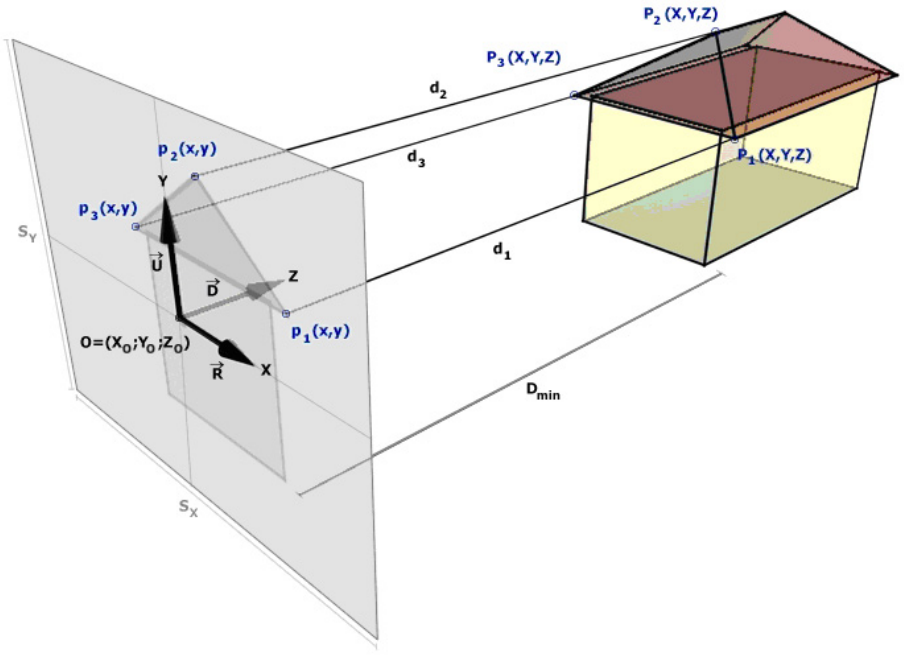


Fig. 12.1. Calibration of the orthogonal camera

To perform geometrical transformation of the perspective image, four control points are necessary. Sometimes an orthophotograph must be created by a collage of layers extracted from several source images. For example, this problem concerns creating an orthophotograph of the façade that has details exposed over the main face of the façade (e.g. recess, especially when photographed from a side).

After the processing of the photographs, the calibration of the photograph must be performed. Because the rectified orthophoto image is used as an orthocamera plane, this process is also called the orthocamera calibration.

In (Kupaj, 2005; Nikiel, 2003), the following camera model is described:

$$\begin{bmatrix} R_X & R_Y & R_Z \\ U_X & U_Y & U_Z \end{bmatrix} \left(\begin{bmatrix} X_p \\ Y_p \\ Z_p \end{bmatrix} - \begin{bmatrix} X_O \\ Y_O \\ Z_O \end{bmatrix} \right) = \begin{bmatrix} x_p \\ y_p \end{bmatrix}. \tag{12.6}$$

The model is used with the calibration procedure. The main purpose of the calibration is to find the size (width and height) of the plane and the position of the camera. Camera orientation is not calculated – we must decide which side of the object is represented by the orthopotograph.

Figure 12.1 depicts the main principle of calibration. To perform successful calibration, three pairs of control points are necessary. Each pair of the control points consists of the XYZ point on the object and its reference on the camera plane. As it is visible, the user has to know the size of the object. With the object size the

control points on the object (XYZ) can be manually calculated. If the dimensions of the object are not available – one must decide by a trial and error method the object dimensions:

$$P_i = \vec{O} + \frac{1}{2} S_X \; p_{i,x} \; \vec{R} + \frac{1}{2} S_Y \; p_{i,y} \; \vec{U} + d_i \; \vec{D}. \tag{12.7}$$

Each point can be described with the use of the equation (12.5). The distance between the point on the model (XYZ) and the point referenced on the camera plane is described as d_i:

$$\underset{i=1\ldots3}{d_i} = d_{P_i} - (d_{\mathrm{MIN}} - D). \tag{12.8}$$

As is shown, the D_{P_i} distances have to be determined. This might be performed with the following equation:

$$\underset{i=1\ldots3}{d_{P_i}} = \frac{D_X P_{iX} + D_Y P_{iY} + D_Z P_{iZ}}{\sqrt{D_X^2 + D_Y^2 + D_Z^2}}. \tag{12.9}$$

Then, for each pair of the control points, the following equation might be written:

$$\begin{bmatrix} 1 & 0 & 0 & \frac{1}{2} p_{ix} \; R_X & \frac{1}{2} p_{iy} \; U_X \\ 0 & 1 & 0 & \frac{1}{2} p_{ix} \; R_Y & \frac{1}{2} p_{iy} \; U_Y \\ 0 & 0 & 1 & \frac{1}{2} p_{ix} \; R_Z & \frac{1}{2} p_{iy} \; U_Z \end{bmatrix} \begin{bmatrix} X_O \\ Y_O \\ Z_O \\ S_X \\ S_Y \end{bmatrix} = \begin{bmatrix} P_{iX} - d_i \; D_X \\ P_{iY} - d_i \; D_Y \\ P_{iZ} - d_i \; D_Z \end{bmatrix}. \tag{12.10}$$

With the use of the full (three) pair of the points, a simple set of equations appears that might be easily solved.

12.2.4. Virtual reconstruction

With the cameras calibrated we can start the main reconstruction process. With the help of two camera planes (perpendicular to each other), one can extract characteristic points of the model. This process relies on selection of a point that is located near two rays which are created by the projection on the camera plane. The method used to reconstruct the point in the space for reconstruction purposes is described in (Nikiel, 2003; Nikiel and Kupaj, 2004). The reconstruction point algorithm is based on the spatial line intersection algorithm described in (Kupaj, 2005; Nikiel, 2003).

The prototype application of the BluePrint Modeler is a CAD system designed for rapid development of low-count polygon models. It includes basic tools that help to create a spatial object. 3D objects can be exported to virtual reality environments in the format compliant to the VRML/X3D language.

Although the application is in the development phase, reconstructions are possible – one can use the following tools:

- lens distortion correction tool (where lens distortion is removed from the source photographs),

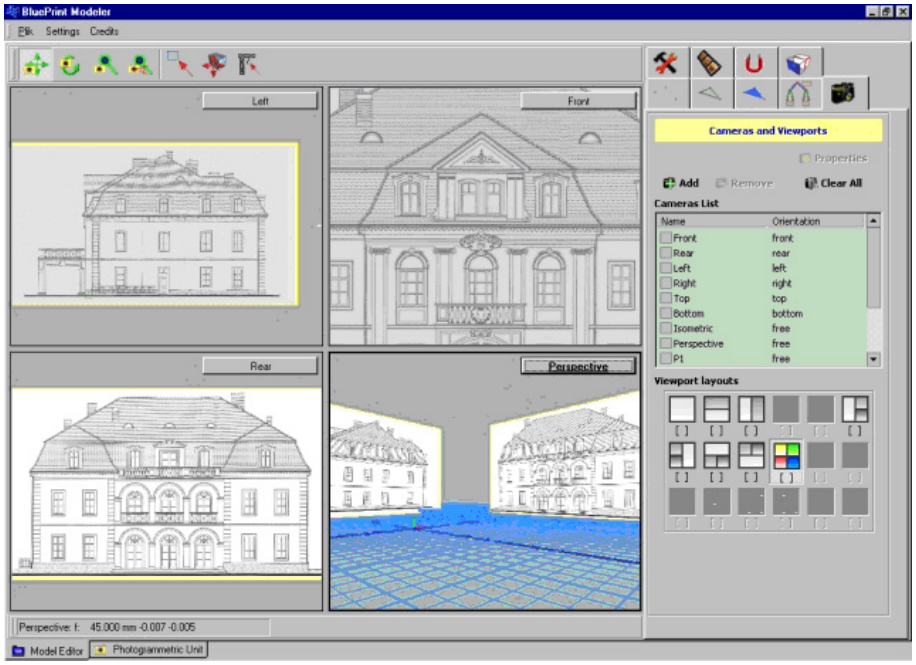


Fig. 12.2. BluePrint Modeler – a sample screen of the Model Editor

- perspective correction tool (the user can use this tool to straighten perspective shots in order to get orthophotos. This tool is used also when one want to create the texture for the model),
- orthocamera calibration tools (used to calibrate fundamental orthogonal views: front elevation, left-right side, etc.),
- model editor (used to extract the geometry of the model and to apply texture coordinates).

The Jordan Qasr Al-Kharaneh castle was chosen to illustrate the photogrammetric reconstruction. The choice was consulted with the Queen Rania University, Jordan. The materials that depict the castle are based on images found on the Internet. The internet can be used extensively used in data sharing, which is particularly effective in long-distance collaboration (Nikiel and Steć, 1998). The photographs came from different cameras without known orientation and interior parameters. In that case, correct lens calibration was impossible. It was decided to correct (by the perspective tool) the front side of the building to get the proper source for modeling.

The castle has cylindrical shapes (towers) that make modeling and texturing harder – hence the decision to find the top section of the castle. The section was calibrated (by the method of trials and errors – due to a lack of information about the object's dimensions). Only half of the object was modeled first (the castle is symmetric). With the function of mirror/symmetry, the shape was extended to create a rough spatial model. Next, additional elements of the building (entrance/doors) were

Fig. 12.3. Finished low-polygon reconstruction of the Qasr Al-Kharaneh castle view 1

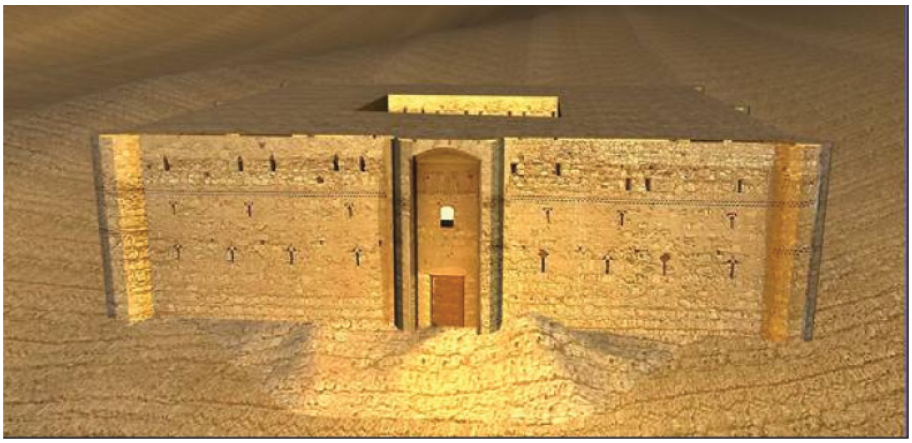


Fig. 12.4. Finished low-polygon reconstruction of the Qasr Al-Kharaneh castle view 2

modeled from the orthocamera's planes. The process is called geometry extraction. All polygons/faces of the object were created. Sometimes the points on the object do not match exactly the points on the image. This situation is typical for poor quality of images or poorly calibrated materials. To get good reconstruction, one should always rely on precise data. Then, screenshots of the profiles were taken. Those screenshots were used as templates to create textures for the castle. The design of the texture was performed in the perspective correction tool. Ready textures were applied to polygons that were created during the geometry extraction/reconstruction from the planes. The completed model was exported to the VRML/X3D language. Then it was rendered in the Cinema 4D environment. The results appear in Figs. 12.3 and 12.4. The whole reconstruction took only a few hours.

12.2.5. Conclusions

Low-count polygon generalized models with limited geometry offer something different. The idea of rapid development systems for archeological purposes should be treated as a fast 3D sketching tool. With the use of limited geometry, details still

might appear – on the texture, which can be easily prepared. If the texture is created with enough quality, then the result is satisfactory. The final effect will depend only on the user's work – automatic systems usually produce too much complexity; only human might create the best generalized low-count polygon model.

The reconstruction shown in the previous section shows that the process with satisfactory results might be completed within short time. The resulting model can be used as a part of a complex scene, even for realtime purposes, due to its low complexity (for the discussed castle only 240 faces/polygons). For rapid reconstructions, image requirements are also important. For good and precise modeling, photographs must be taken according to some rules (http://cipa.icomos.org/). Then, they should be calibrated in order to remove distortions. The measurements of the object should also be known and with a given precision. Reconstruction from image planes permits better analysis of the geometry of artifacts as compared to the automatic process. We can decide what is to be reconstructed and this permits to create an optimal low-count polygon model (3D sketch). It is possible to get satisfactory results even if the source materials are of low quality. The reconstruction process is cost effective and can be done with a minimal amount of time unless it is based on optically corrected digital images. Therefore it is possible to include it in the process of archaeological analysis, adding a new possibility of on-site 3D reconstruction and direct porting/distribution of the model on the Internet.

12.3. Extraction of multiple objects using multi-label fast marching

In some applications, foreground-background segmentation (Steć and Domański, 2003a) is not sufficient. The detection of all objects in the scene would be more appropriate. The original method that would be presented in this chapter is aimed at performing such a task. As the main tool, the multi-label Fast Marching Method (FMM) will be used. The idea of simultaneous propagation of multiple contours using the FMM was introduced by Sifakis and Tziritas (2001; 2002). However, the method presented in this section shares with the Sifakis approach merely the idea of multi-label fast marching. In the original method only two labels are used, and each of them has individual propagation speed. Such an approach is hard to extend to multiple object segmentation, especially when the number of objects is unknown. The approach presented here uses the same propagation speed for all labels, thus the number of labels does not influence algorithm work. Moreover, there is no limitation to a static or motion-compensated background. An additional advantage of this approach is that it is easy to define the stop condition since contours are propagating toward each other. In the original method, the algorithm stops when the contours meet. In the method presented in this section, some additional actions can be performed when two segments meet.

12.3.1. Initialization

The initialization procedure is based on displaced frame difference (*dfd*) between two consecutive frames and requires the dense motion field to be computed prior to the initialization. Here it is assumed that the motion field was computed using one of the

Fig. 12.5. Seed regions overlayed on the original frame from the sequence (indicated by the arrows)

already known methods (Horn and Shunk, 1981; Lim and El Gamal, 2001; Lucas and Kanade, 1981).

It is assumed that regions with zero *dfd* are likely to be inside objects presenting the same motion properties. Therefore, such areas are good starting points for contour propagation. A similar procedure was successfully applied in the method presented in the papers (Steć and Domański, 2003b; 2004). Displaced frame difference at the point (x, y) is computed as follows:

$$dfd = \frac{1}{9} \sum_{i=-1}^{1} \sum_{j=-1}^{1} |I_n(x+i, y+j) - I_{n-1}(mx+i, my+j)|, \tag{12.11}$$

where mx and my represent the position of the point in the previous frame according to motion information. Such a computation procedure has two purposes: reducing image noise influence and rejecting single points as starting region candidates. All connected pixels with the *dfd* value equal to zero are labelled as one region with an additional constraint on motion uniformity. This prevents the regions on the border between the object and the background with zero *dfd* but with different motion properties from merging into one region. Each region is assigned an individual label. Such regions will be seeds for contours propagated with the FMM (Fig. 12.5). The number of seed regions is always larger than the number of final segments.

12.3.2. Initial segments propagation

All segments initialized earlier are propagated outwards using a modified fast marching algorithm. Segment labels for points visited by contours are positive integers. Trial points (boundary points sorted by contour arrival times, see (Sethian, 1998)) for each contour are marked with negative numbers of segment labels. All trial points from all segments are included into the same sorted list. Thanks to this, no additional time synchronization between the segments is required. This situation is naturally handled by the fast marching algorithm since it can propagate contours of any topology. At

Fig. 12.6. Segments during the initial stage of propagation

this stage of propagation, there is in fact no difference between the standard and the multi-label implementation apart from the fact that the new label for the trial point is inherited from the segment that propagates at the current algorithm step (Fig. 12.6).

Propagation speed is based only on the current image properties. It is proportional to the inverse gradient value combined with blurred image components:

$$F = \frac{1}{\max(\nabla Y_\sigma, \nabla Cb_\sigma, \nabla Cr_\sigma) + 1}, \tag{12.12}$$

where σ denotes Gaussian blurring. Such a speed definition makes contour motion fast in smooth areas and slow as they approach edges. Thanks to this, contours are likely to meet on the object edge rather than inside the object.

12.3.3. Dynamic regularization of the motion field

Since estimation starts from the points where motion vectors are estimated correctly, these correct vectors are propagated along with the contour and replace the originally computed motion vectors. This situation is correct until the contour stays inside the object from which the propagation started. When the contour crosses the object border, motion vectors that belong to that object are assigned to the background or another object. Such a situation leads to two errors: segments are missaligned with real objects and some parts of the frame have erroneous motion vectors assigned. This situation will last until two contours meet. Section 12.3.4 will give a solution to errors introduced in this step. After segmentation is finished, every segment has a uniform motion field. The final result is shown in Fig. 12.7.

Such a simplification reduces the application of this method to sequences with translational motion of rigid objects. For example, traffic monitoring sequences fall into this group. Nevertheless, the application of a more sophisticated motion regularization method and using full motion information will extend the reliability of this

(a) (b)

Fig. 12.7. Regularized motion field for the frame 190 from the 'Mobile' sequence; (a) – horizontal component, (b) – vertical component

method. This is possible since the only requirement of this segmentation method is motion consistency within the propagating segment.

12.3.4. Segment merging and pushing

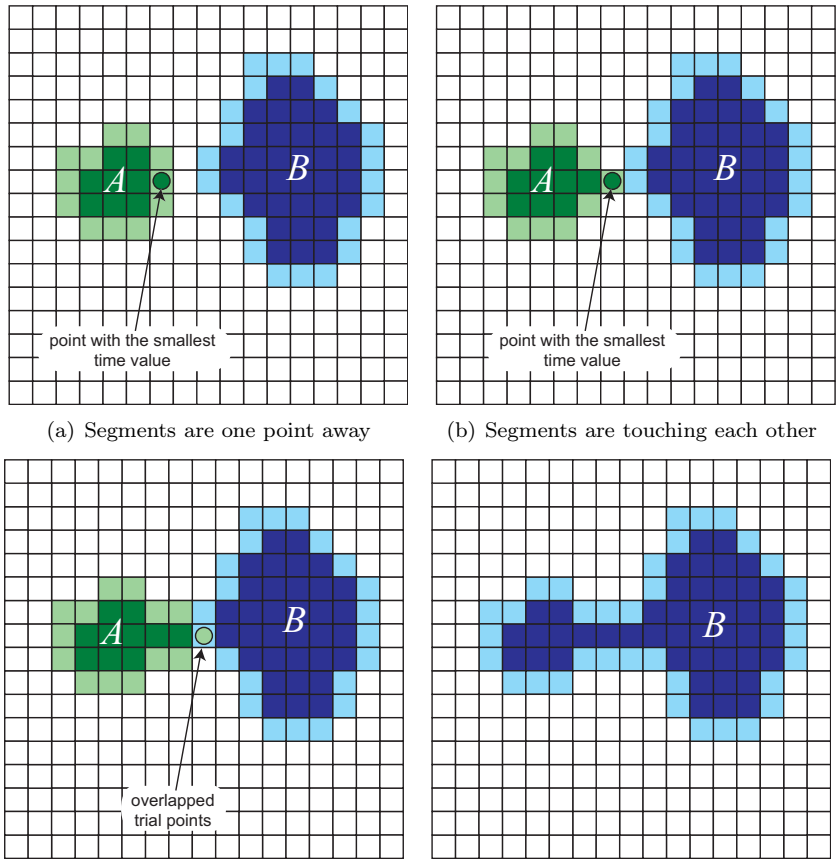
The expansion of the segment described in Section 12.3.2 is performed as long as new trial points can be set on the area not visited by any of the propagating curves. When a new trial point is going to be set in a place occupied by a trial point from another segment, two actions can be performed: the segments can be merged or one contour can be pushed back by another.

Segments merging. When two segments meet, the motion of these segments is compared. The meeting point is a trial point from one segment that must be placed over a trial point from another segment (Fig. 12.8(c)). Since motion within segments is the same for all points, it is sufficient to take one point from each segment for comparison. Motion from the segment A is compared with motion from the segment B according to the following expression:

$$|mx_A - mx_B| < \varepsilon \wedge |my_A - my_B| < \varepsilon, \tag{12.13}$$

where mx and my are motion vector components and ε is an empirically chosen merging threshold. During tests that were performed on a number of sequences, the best results gave $\varepsilon = 0.9$. This means that segments with motions different by less than one pixel per frame are connected. Motion vectors are estimated with sub-pixel accuracy. Additional research is needed to find a way of automatically adjusting ε. When the expression (12.13) is true, the segments A and B are merged.

To ensure maximum efficiency, labels from the smaller segment are changed to the value of those from the larger segment. Also, trial points from smaller segments are assigned the value from larger segments (Fig. 12.8(d)). Motion vectors from smaller regions are replaced with motion vectors from larger regions to ensure motion uniformity within segments.



(a) Segments are one point away

(b) Segments are touching each other

(c) Trial point overlaps the existing one. Motion similarity is checked

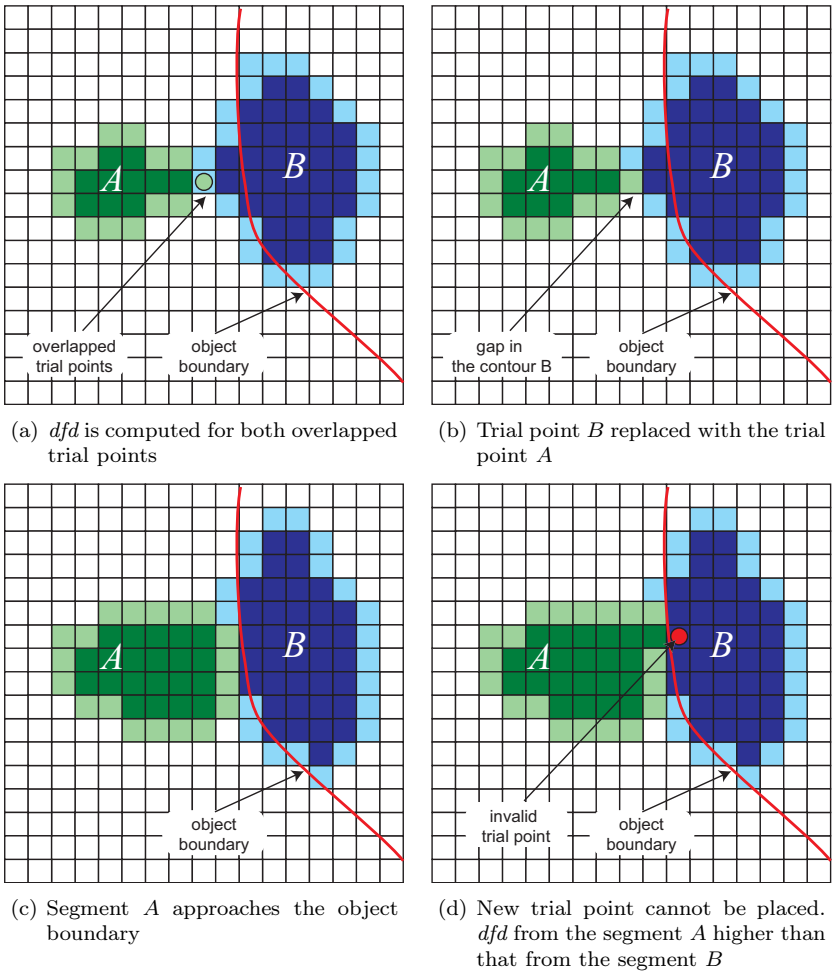
(d) Segments merged together

Fig. 12.8. Procedure of merging segments with high motion similarity. Trial points are marked in a brighter colour

Segment pushing. If two segments that meet are not classified to be merged, the propagating segment can push back another segment under certain circumstances. When a trial point from the propagating segment A is going to be placed at the position (x, y) occupied by a trial point from another segment B (Fig. 12.9(a)) and motion similarity is not high enough, then the displaced frame difference is computed for both segments:

$$dfd_A = \frac{1}{9} \sum_{i=-1}^{1} \sum_{j=-1}^{1} |I_n(x+1, y+j) - I_{n-1}(mx_A + i, my_A + j)|, \qquad (12.14)$$

$$dfd_B = \frac{1}{9} \sum_{i=-1}^{1} \sum_{j=-1}^{1} |I_n(x+1, y+j) - I_{n-1}(mx_B + i, my_B + j)|, \qquad (12.15)$$



(a) *dfd* is computed for both overlapped trial points

(b) Trial point B replaced with the trial point A

(c) Segment A approaches the object boundary

(d) New trial point cannot be placed. *dfd* from the segment A higher than that from the segment B

Fig. 12.9. Segment A with lower *dfd* pushes the segment B with higher *dfd* back to the object boundary

where I_n is the n-th image from the sequence, mx, my are motion compensated positions of the pixels, and the indexes A and B denote the segment being the source of motion information. If $dfd_A < dfd_B$, then the trial point from the segment A replaces the trial point from the segment B. In the case when $dfd_A > dfd_B$, the trial point from the segment A is not placed and no further propagation is performed. The latter case means that the meeting point belongs to the segment B. At that point only the segment B has the possibility of propagating further, because the trial point from the segment A was not set. If the point considered lies on the object border, the segment B cannot propagate either because the trial point from B will have higher *dfd* than the point from A set earlier. The segment B can propagate further if the segment A passed its object border and the meeting took place on the object that belongs to B. The segment B will push back the segment A to the nearest object border.

The replacement of the point from the trial list of the segment B creates a gap on the segment boundary (Fig. 12.9(b)). Nonetheless, it has no influence on further propagation of neither the segment B nor the segment A. The replaced point has no chance of propagating anyway because its *dfd* was higher than that of the segment A. The remaining portion of the segment B is propagated normally. The fast marching algorithm does not require a closed contour for propagation.

The segment A stops pushing back the segment B on the boundary of the object which has motion properties similar to those represented by the segment B. In such a case, the segment A cannot propagate further, because its *dfd* for the trial point that is going to be set inside the object occupied by the segment B will be higher than that for the segment B (Fig. 12.9(d)).

When a contour has no possibility of propagating further, no new trial points are set. This implies the reduction of the total length of the sorted list used by the fast marching algorithm and the same performance improvement.

12.3.5. Stop condition

The presented algorithm stops propagation when all image points are assigned to segments and there is no segment that could push back another segment. The algorithm cannot run infinitely because oscillations between segments are impossible. No segment can visit twice the same area. Namely, when a segment was pushed back by another segment, it cannot get the lost pixels back.

12.3.6. Experiments

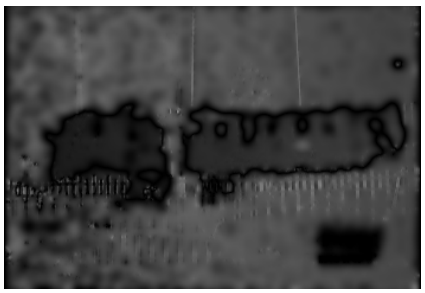
The proposed method of segmentation using multi-label fast marching was evaluated experimentally. The algorithm was able to segment complex scenes with multiple overlapping objects and with objects partially visible in the scene. An example of such a scene is presented in Fig. 12.10.

The algorithm requires a *partially reliable* motion field for correct performance. This means that the motion estimation algorithm must be able to produce at least some parts of the motion field, with motion vectors that point precisely onto the corresponding pixels from the previous frame (*dfd* for these points is zero). An example of such a motion field is shown in Figs. 12.10(c) and 12.10(d). The algorithm fails when there is no reliable motion field. Figure 12.13 shows the segmentation of Frame 8 from the 'Bus' sequence. For this frame, motion was too fast for the currently implemented motion estimation algorithm. Motion vectors are mostly erroneous and the consequence is a wrongly segmented image. However, for the testing purposes, only simple classical motion estimation algorithms were implemented. The implementation of a faster and more precise motion estimation method will improve the performance of the segmentation algorithm.

The current speed definition allows calculating speed for the whole frame before propagation begins using fast convolution filters. During the propagation, speed is only read from the table. However, the total length of the propagated contours is quite big and the biggest impact on the performance comes from the implementation of the sorting algorithm used by the FMM. The performance of the algorithm can be improved by the parallelization of the propagation process. Because timing between

(a) Original image

(b) Segmented image



(c) Motion in the x direction

(d) Motion in the y direction

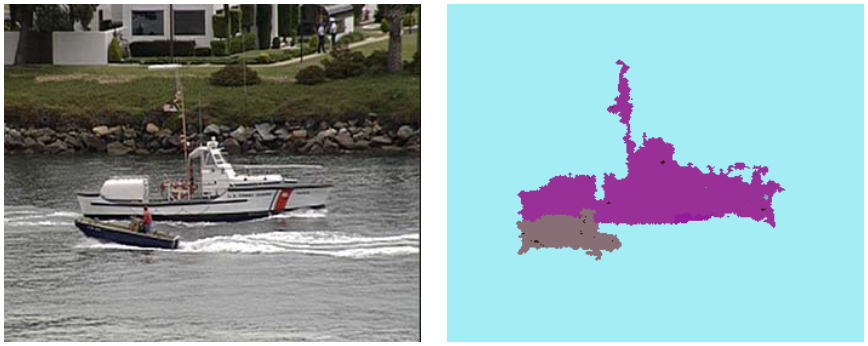
(e) Regularized motion in the x direction

(f) Regularized motion in the y direction

Fig. 12.10. Frame 112 from the 'Bus' sequence segmented using multi-label fast marching

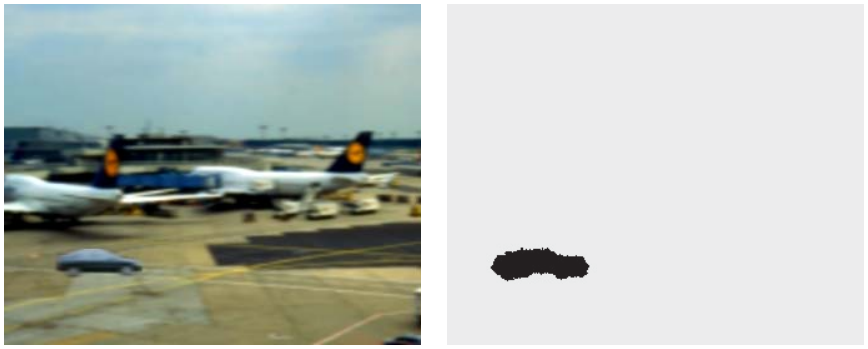
contours is not important, the propagation of the segments can be divided between an arbitrary number of threads. This is possible because multiple contours can be propagated in a single thread like in the implementation presented here. Another way is to use parallel implementation of the FMM like the one proposed by Dejnozkova and Dokladal (2003).

In this implementation, only sequences with translational and rigid motion can be segmented. However, this limitation is dependent on the motion regularization method, which is an integral part of the algorithm and can be replaced with another method without interference with the core algorithm (Steć, 2005). Figure 12.14 presents a sequence with complex non-rigid motion that causes problems for the algorithm.



(a) Original image (b) Segmented image

Fig. 12.11. Frame 78 from the 'Coast Guard' sequence segmented using multi-label fast marching



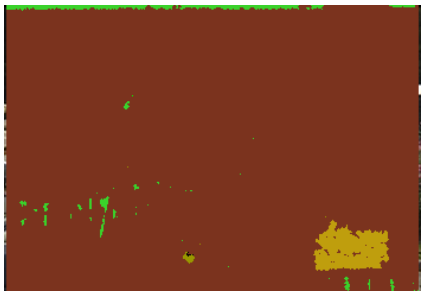
(a) Original image (b) Segmented image

Fig. 12.12. Exemplary frame from the semi-artificial 'Car' sequence segmented using multi-label fast marching

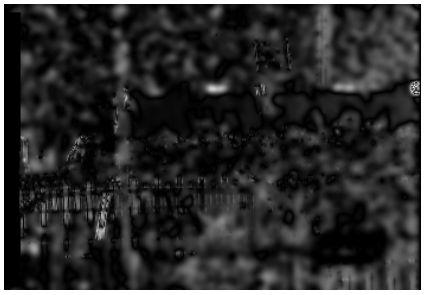
Computational efficiency of this method is similar to the one presented in (Steć and Domański, 2004). Nevertheless, the total time of frame segmentation is now higher despite the much simpler propagation speed definition. The current speed definition allows calculating speed for the whole frame before propagation begins using fast convolution filters. During the propagation, speed is only read from the table. However, the total length of the propagated contour is much bigger than it was in the mentioned method. As a consequence, segmentation with multi-label fast marching can last even three times longer. This is a proof of the fact that the biggest impact on the performance comes from the implementation of the sorting algorithm used by the FMM.

Multi-label propagation was implemented using Sifakis's (2001; 2002) approach, namely, by a single sorted list. This approach has some advantages when all propagated contours must be synchronized in time. They share the sorted list, thus the fastest point of all from all contours is always propagated. For Sifakis this was essential since two propagated contours were expected to meet on the object boundary. In his method, there is no possibility of correction when the boundary is missed.

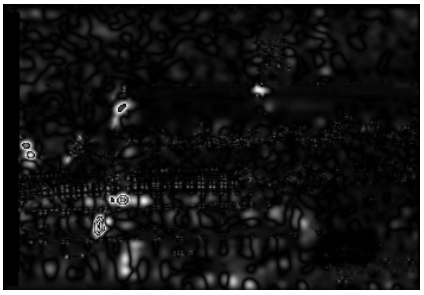
(a) Original image



(b) Segmented image



(c) Motion in the x direction

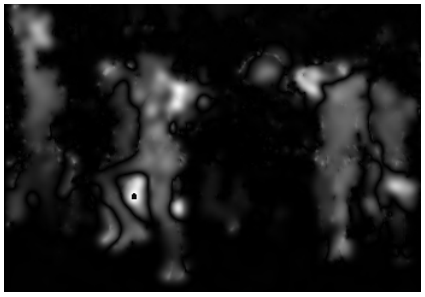


(d) Motion in the y direction

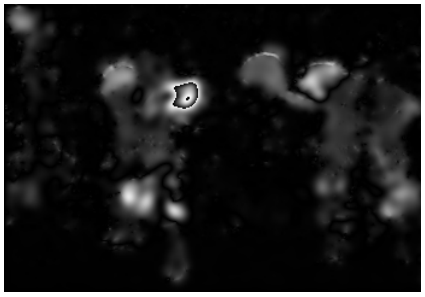
Fig. 12.13. Frame 8 from the 'Bus' sequence segmented using multi-label fast marching. Problems with the erroneous motion field

(a) Original image

(b) Segmented image



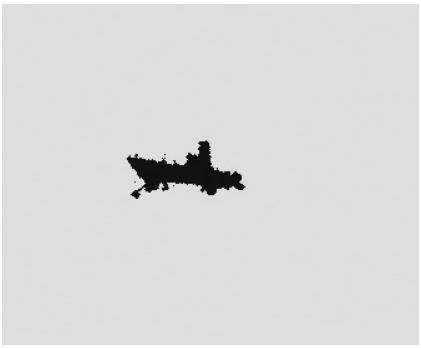
(c) Motion in the x direction



(d) Motion in the y direction

Fig. 12.14. Frame 166 from the 'Cheer' sequence segmented using multi-label fast marching. Problems with complex elastic motion

(a) Original image



(b) Segmented image

(c) Result obtained by Patras

Fig. 12.15. Frame 2 from the 'Coast Guard' sequence segmented using multi-label fast marching compared to the result obtained by another author

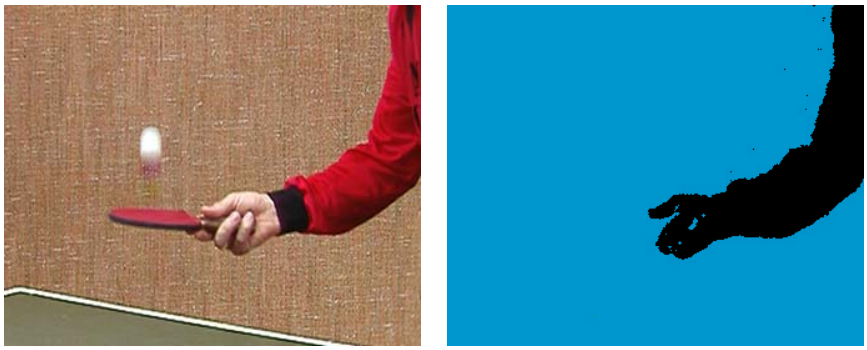
In the method presented in this chapter, such time synchronization is not essential. It is important because the first stage of propagation is much faster than segment pushing but it is not critical. This opens the way for parallel implementation of propagating contours. Each contour can be propagated by a different thread until the contours are merged. During the merging, the thread with a smaller segment is destroyed and its contour is taken over by the thread with a bigger segment (Patras *et al.*, 2001). The implementation of segment pushing can be virtually the same as in the version with the single list since threads do not process points already visited by the contour.

Figure 12.15 shows a comparison of a frame segmented with the method presented in this chapter and with the method presented in (Patras *et al.*, 2001). While segmentation quality is on a similar level, algorithm complexity is much lower in the method developed in this work. Patras uses for segmentation several steps, and each of them converges iteratively to the final solution. However, Patras's method deals better with large object displacements.

Figures 12.16 to 12.18 show exemplary segmentations obtained with the method developed in this chapter. Objects with large displacements are usually undetectable, which is caused by the limitation of the motion estimation method used for tests. In

(a) Original image (b) Segmented image

Fig. 12.16. Frame 199 from the 'Coast Guard' sequence segmented using multi-label fast marching



(a) Original image (b) Segmented image

Fig. 12.17. Frame 11 from the 'Table Tennis' sequence segmented using multi-label fast marching

some cases the method tends to oversegment the frame or leave small false objects (Fig. 12.18). A way of keeping the segments more consistent must be found through further development of the method.

12.3.7. Conclusions

The algorithm presented in this chapter facilitates fast and fully automatic (unsupervised) segmentation of colour video sequences in the presence of a moving background without the necessity for global motion compensation. The algorithm is designed to segment individual frames without object tracking. The motivation for such an approach is the fact that there is a large number of object tracking algorithms (Dubuisson Jolly *et al.*, 1996; Guo *et al.*, 1999; Iannizzotto and Vita, 2000; Irani *et al.*, 1992; Mansouri and Konrad, 1999) that require manual initialization of the object boundary. However, there exists the problem of automatic search of objects at the beginning of the sequence.

(a) Original image (b) Segmented image

Fig. 12.18. Frame 165 from the 'Mobile' sequence segmented using multi-label fast marching

The presented algorithm was designed to segment video frames into multiple disjoint objects. Segmentation is proposed for natural sequences, i.e., sequences that represent the natural world as perceived through a camera, and not created by computer graphic tools.

Here, the main concern was algorithm speed and stability rather than segmentation quality. The algorithm is suitable for real-time processing of video with the use of fast processors. The current version of the algorithm cannot deal with complex motion and sometimes may produce oversegmented frames. Nevertheless, the authors have found that further extensions and improvements are possible.

References

Basu A. and Licardie S. (1995): *Alternative models for fish eye lenses.* — Pattern Recognition Letters, Vol. 16, pp. 433–441.

Criminisi A., Reid I. and Zisserman A. (1999): *DA plane measuring device.* — Report, Department of Engineering Science, University of Oxford.

Cucchiara R., Grana R., Prati A. and Vezzani R. (2003): *A Hough transform-based method for radial lens distortion correction.* — Proc. 12th IEEE Conf. *Image Analysis and Processing*, Mantova, Italy, pp. 182–187.

Dejnozkova E. and Dokladal P. (2003): *A parallel algorithm for solving the eikonal equation.* — Proc. IEEE Int. Conf. *Image Processing, ICIP*, Barcelona, Spain, pp. 325–328.

Dubuission Jolly M., Lakshmanan S. and Jain A. (1996): *Vehicle segmentation and classification using deformable templates.* — IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 18, No. 3, pp. 293–308.

Guo J., Kim J. and Kuo C. (1999): *An interactive object segmentation system for MPEG video.* — Proc. IEEE Int. Conf. *Image Processing, ICIP*, Kobe, Japan, pp. 140–144.

Horn B. K. and Shunk B. G. (1981): *Determining optical flow.* — Artificial Intelligence, Vol. 17, pp. 185–203.

Iannizzotto G. and Vita L. (2000): *Real-time object tracking with movels and affine transformations*. — Proc. IEEE Int. Conf. *Image Processing, ICIP*, Vancouver, Canada, Vol. I, pp. 316–322.

Irani M., Rousso B. and Peleg S. (1992): *Detecting and tracking multiple moving objects using temporal integration*. — Proc. European Conf. *Computer Vision*, Berlin: Springer-Verlag, Lecture Notes in Computer Science, Vol. 588, pp. 282–287.

Karras G.E. and Mavrommati D. (2001): *Simple Calibration Techniques for Non-Metric Cameras*. — Report, Department of Surveying, National Technical University, Athens.

Karras G.E, Mountrakis G., Patias P. and Petsa E. (1998): *Modelling distortion of super-wide-angle lenses for architectural and archeological applications*. — Int. Archives for Photogrammetry and Remote Sensing, Vol. 32, No. 5, pp. 570–573.

Krauss K. (1997): *Photogrammetry*. — Bonn: Dummler Verlag.

Kupaj M. (2005): *Photogrammetric Reconstruction Systems for Purposes of Virtual Reality Systems*. — M.Sc. thesis, Faculty of Electrical Engineering, Computer Science and Telecommunications, University of Zielona Góra, (in Polish).

Lim S. and El Gamal A. (2001): *Optical flow estimation using high frame rate sequences*. — Proc. IEEE Int. Conf. *Image Processing, ICIP*, Thessaloniki, Greece, pp. II:925–II:928.

Lucas B. D. and Kanade T. (1981): *An iterative image registration technique with an application to stereo vision*. — Proc. *DARPA Image Understanding*, Washington, DC, USA, pp. 121–130.

Mansouri A.-R. and Konrad J. (1999): *Motion segmentation with level sets*. — Proc. IEEE Int. Conf. *Image Processing, ICIP*, Kobe, Japan, pp. 126–130.

Nikiel S. (2000a): *VRML – a 3D environment for virtual classes*. — Proc. EUNIS Conf. *Towards Virtual Universities*, Poland, Poznań, pp. 129–134.

Nikiel S. (2000b): *Virtual environments for distributed projects*. — Proc. Polish-German Conf. *Science, Research, Education*, Zielona Góra, Poland, pp. 177–180.

Nikiel S. (2001): *Virtual Office – a project of 3D internet GUI*. — Proc. PIONIER Conf. *Polish Optical Internet*, Poznań, Poland, pp. 203–206, (in Polish).

Nikiel S. (2002): *Erstellung einer internetbasierten, virtuellen Umgebung zur Darstellung das Palastes in Żagań*. — Proc. EVA Conf. *Elektronische Bildverarbeitung und Kunst, Kultur, Historie*, Berlin, Germany, pp. 258–261.

Nikiel S. (2003): *Blue-print based modeling of architectural artifacts*. — Proc. EVA Conf. *Elektronische Bildverarbeitung und Kunst, Kultur, Historie*, Berlin, Germany, pp. 189–192.

Nikiel S. (2004): *Bildung eines virtuellen Wiederaufbaus mit dem Umgebungskontext*. — Proc. EVA Conf. *Elektronische Bildverarbeitung und Kunst, Kultur, Historie*, Berlin, Germany, pp. 47–49.

Nikiel S. (2007): *Iterated function systems for real-time image synthesis*. — Springer-Verlag, UK, (in print).

Nikiel S. and Goiński A. (2005): *A recursive subdivision scheme for isosurface construction*. — Computers and Graphics, Vol. 29, No. 1, pp. 155–164.

Nikiel S., Kirby G.H. and Clausse J. (2001): *Fractal palettes for texturing*. — Computers and Graphics, Vol. 27, No. 6, pp. 977–982.

Nikiel S. and Kupaj M. (2004): *Photogrammetry in computer aided design for architecture.* — Proc. 5th Conf. *Measurement Systems in Sci. Research and in Industry Appl., SP*, Łagów, Poland, (in Polish), CD.

Nikiel S. and Stachera P. (2004): *Fractal image compression for efficient texture mapping.* — Int. Workshop on *Computer Graphics*, Plzen, Czech Republic, Vol. 12, No. 1–3, pp. 169–172.

Nikiel S. and Steć P. (1998): *Marketing stretegies for web-based applications.* — Proc. MISSI Conf. *Multimedia and Networked Information Systems*, Wrocław, Poland, pp. 305–311 (in Polish).

Nikiel S. and Steć P. (2000): *Automated terrrain generation for virtual environments.* — Proc. MISSI Conf. *Multimedia and Networked Information Systems*, Wrocław, Poland, pp. 387–392 (in Polish).

Patras I., Hendriks E. and Lagendijk R. (2001): *Video segmentation by MAP labeling of watershed segments.* — IEEE Trans. Pattern Analysis and Machine Intelligence, Vol. 23, No. 3, pp. 326–332.

Quan L. (1996): *Self callibration of the affine camera form the multi views.* — Int. J. Computer Vision, Vol. 19, No. 1, pp. 93–105.

Sethian J. (1998): *Fast marching methods and level set methods for propagating interfaces.* — Computational Fluid Dynamics, Vol. 1 of VKI Lectures series, von Karman Institute.

Sifakis E. and Tziritas G. (2001): *Moving object localisation using a multi-label fast marching algorithm.* — Signal Processing: Image Communication, Vol. 16, No. 10, pp. 963–976.

Sifakis E. and Tziritas G. (2002): *Video segmentation using fast marching and region growing algorithms.* — EURASIP J. Applied Signal Processing, pp. 379–388.

Steć P. (2005): *Segmentation of Colour Video Sequences Using the Fast Marching Method.* — Vol. 6 of Lecture Notes in Control and Computer Science, University of Zielona Góra Press, Poland.

Steć P. and Domański M. (2003a): *Efficient unassisted video segmentation using enhanced fast marching.* — Proc. IEEE Int. Conf. *Image Processing, ICIP*, Barcelona, Spain, pp. 246–253.

Steć P. and Domański M. (2003b): *Two-step unassisted video segmentation using fast marching method.* — Proc. 10th Int. Conf. *Computer Analysis of Images and Patterns, CAIP 2003*, Lecture Notes in Computer Science, Groningen, Holland: Springer-Verlag, Vol. 2756, pp. 246–253.

Steć P. and Domański M. (2004): *Fast two-step unassisted video segmentation technique evaluated by tolerant ground truth.* — Proc. 5th Int. Workshop *Image Analysis for Multimedia Interactive Services*, Lisboa, Portugal, pp. 867–870.

Chapter 13

LOW DELAY THREE-DIMENSIONAL WAVELET CODING OF VIDEO SEQUENCES

Andrzej POPŁAWSKI*, Wojciech ZAJĄC*

13.1. Introduction

Since the early 1980s we have been observing a significant progress in the area of image and video compression and transmission. There have been proposed new and more efficient compression techniques concerning both multimedia information and video sequences. Despite the fact that the new techniques are getting more complex, they are successfully implemented in the environment of still faster and better computing systems. One of many known coding (and compression) techniques is the hybrid approach. Based on it, since the late 1980s there have been developed numerous subband coding techniques, again for still and moving pictures. Since the 1990s the subband coding has been commonly referred to as wavelet coding, despite the fact that the wavelet coding is a particular case of more general subband coding. The rise of new techniques has led to the acceptance of new standards, used in a variety of applications.

Recently, numerous standards of video compression have appeared. The H.261 recommendation (ITU-T Rec. H.261) is intended to be applied in videotelephony and teleconference systems and it ensures that coding delay will be no greater than 150 ms. H.263 (ITU-T Rec. H.263) is an extension of H.261 dedicated for use in videotelephony of higher quality. H.264 (ISO/IEC 14496-10 AVC / ITU-T Rec. H.264) provides even more effective compression. Finally, one of the most popular standards, MPEG-2 (ISO/IEC International Standard 13818), is widely applied in digital media distribution (DVD) and transmission (Internet and television systems, including HDTV).

The common feature of all coders mentioned above, often referred to as hybrid coders, is the use of the Discrete Cosine Transform – DCT (Ahmed and Natarajan,

* Institute of Computer Engineering and Electronics
e-mails: {A.Poplawski, W.Zajac}@iie.uz.zgora.pl

1974). The practical alternative to the DCT is the Discrete Wavelet Transform – DWT (Jayant and Noll, 1984; Strang and Nguyen, 1996; Topiwala, 1998; Vaidyanathan, 1993; Vetterli and Kovačević, 1995; Woods, 1991), widely examined in recent years. Wavelet-based coders are an attractive alternative to popular hybrid coders, due to the natural feature of full scalability (temporal, spatial and SNR scalability), allowing the control of the transmission bitrate (Domański, 1998). Furthermore, the coding efficiency of wavelet coders is competitive to the coding efficiency of hybrid coders (Hsiang and Woods, 2000; 2001; Xu *et al.*, 2001).

Contemporary wavelet coders are mostly of a 3-dimensional character, applying the wavelet analysis in two spatial dimensions (X and Y) and one temporal dimension (time). Currently used 3D wavelet coding techniques introduce significant time delay (temporal delay), caused by temporal filtering (Ohm, 2002).

Coding delay emerges because certain number of pictures of a given sequence must be grouped (Group of Pictures – GOP) and processed simultaneously. Such a group can consist of, e.g., 8 or 16 pictures. The encoder, to start processing, must wait to collect all the pictures in the GOP, which causes significant delay. Such a delay is the main reason of problems in using wavelet coders in numerous solutions.

Low delay video coding is crucial in applications based on interactions, such as Internet applications, videotelephony, videoconferencing, remote surveillance systems, online education, etc. For example, coding delay for videoconferencing is defined in H.621 as no greater than 150 ms.

The reduction of wavelet coding delay caused by temporal filtering is a subject of intense research. There were proposed some approaches (Huang *et al.*, 2003; Li *at al.*, 2005 Pau *et al.*, 2004; 2005; Seran and Kondi, 2005; Schwarz *et al.*, 2004b; Viéron *et al.*, 2005), though there were judged as unsatisfactory. In research and result estimation, there are used different test video sequences and different versions of codecs.

The aim of this chapter is to carry out an exhaustive experimental research of different temporal filtering schemes of reduced coding delay and to assess the compression efficiency of test video sequences for these schemes in order to select the best solutions for the assumed reduced coding delay.

13.2. Temporal filtering in 3D wavelet coders

Subband temporal analysis with motion compensation is the key operation in three-dimensional wavelet coders. It takes advantage of subsequent images similarity to eliminate a great amount of redundant information. To perform wavelet analysis of a video sequence, the pictures of the sequence are grouped. Such a group is filtered with motion compensation, producing one low frequency picture and seven high frequency pictures (Chen, 2003; Choi and Woods, 1999; Ohm, 1992; 1993; 1994).

Figure 13.1 presents a temporal filtering scheme for a group of eight pictures. The analysis begins on the first level of temporal decomposition ($k = 1$) and consists in filtering subsequent pairs of the input pictures A and B. As a result, each pair produces two filtered images: a low frequency component L and a high frequency component H (Fig. 13.1). On the next level of temporal decomposition ($k = 2$), there are processed only low frequency components, producing one low frequency component LL and high frequency components LH. The same scheme is repeated

on consecutive decomposition levels, as long as we obtain only one low frequency component and one high frequency component. For our group of eight pictures the final components are: one low frequency component LLL and seven high frequency components: four H, two LH and one LLH.

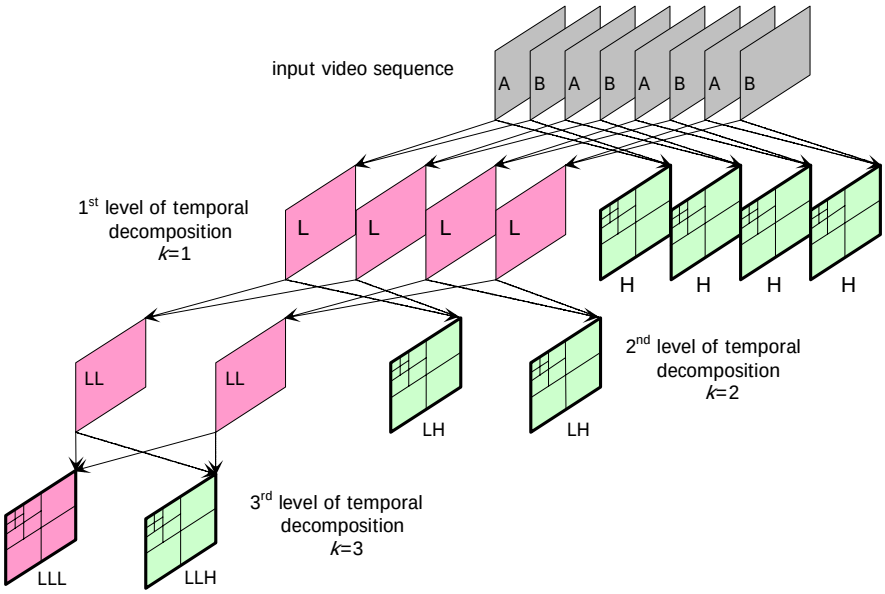


Fig. 13.1. Temporal filtering scheme for a group of 8 pictures

The products of temporal filtering are then subject to two-dimensional spatial analysis, leading to the final product of three-dimensional filtering of video sequence. In practice, wavelet analysis is performed with the use of the so-called lifting structure (Claypoole *et al.*, 2003; Daubechies and Sweldens, 1996; Sweldens and Schröder, 1996). The first stage of such an analysis is splitting the input signal y into even and odd samples y_0 and y_1 (Fig. 13.2). The next step is the actual analysis: the output component y'_1 is a result of the operation of the predictor P (which represents high frequency filtering) and the y'_0 component is produced by the update operator U (representing low frequency filtering). The operation can be represented by

$$y'_1 = y_1 - P(y_0), \quad y'_0 = y_0 + U(y'_1). \tag{13.1}$$

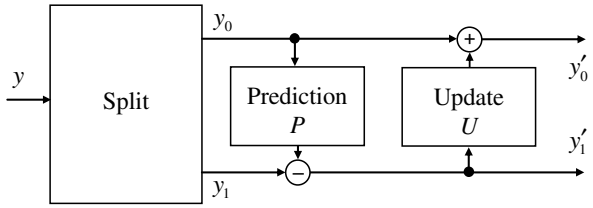


Fig. 13.2. Lifting filtering scheme in wavelet analysis

Such a filtering scheme is widely used in wavelet coders of video sequences mainly due to the perfect reconstruction feature as well as lower numerical cost and memory consumption than those of the classic solution (Daubechies and Sweldens, 1996; Sweldens, 1996; Sweldens and Schröder, 1996).

13.2.1. Temporal filters

An important issue of coding system design is the selection of proper filters. In the case of temporal filtering, the use of high order filters is disadvantageous. Firstly, the high order of a filter implies the necessity of calculating a higher number of motion vectors, producing more data to process and increasing the numerical cost of the operation. Secondly, the longer the filter, the bigger the coding delay, caused by temporal analysis. In practice, contemporary wavelet coders use two types of temporal filters: Haar filters (Robbani and Jones, 1991; Topiwala, 1998; Woods, 1991) and LeGall 5/3 filters (LeGall and Tabatabai, 1988).

13.2.1.1. Temporal filtering with the use of Haar filters

Motion compensated temporal filtering is performed with the use of a lifting filtering scheme. For Haar filters, the analysis is performed as described in Eqn. (13.2) and the synthesis as in Eqn. (13.3):

$$h_t = x_{2t+1} - MC(x_{2t}, mv^+_{2t+1}), \quad l_t = x_{2t} + \frac{1}{2}MC^{-1}(h_t, mv^+_{2t+1}), \tag{13.2}$$

$$x_{2t} = l_t - \frac{1}{2}MC^{-1}(h_t, mv^+_{2t+1}), \quad x_{2t+1} = h_t + MC(x_{2t}, mv^+_{2t+1}), \tag{13.3}$$

where x_t is the picture of the input sequence in time t, h_t is a sample of a high-band signal in time t, l_t is a sample of a low-band signal in time t, mv^+_{2t+1} is the set of motion vectors between pictures x_{2t+1} and x_{2t}, (m, n) are spatial coordinates of the picture, $MC(x_t, mv_t)$ is the motion compensation operator, defined as $x_t(m, n - mv_t(m, n))$.

Figure 13.3 illustrates temporal filtering for three-level temporal decomposition. The dotted arrow indicates the prediction step and the dashed one – the update step.

13.2.1.2. Temporal filtering with the use of 5/3 filters

Motion compensated temporal filtering performed with the use of a lifting scheme and based on LeGall 5/3 filters is described by (13.4) – the analysis, and (13.5) – the synthesis:

$$\begin{aligned} h_t &= x_{2t+1} - \frac{1}{2}\big(MC(x_{2t}, mv^+_{2t+1}) + MC(x_{2t+2}, mv^-_{2t+1})\big), \\ l_t &= x_{2t} + \frac{1}{4}\big(MC^{-1}(h_{t-1}, mv^-_{2t-1}) + MC^{-1}(h_t, mv^+_{2t+1})\big), \end{aligned} \tag{13.4}$$

$$\begin{aligned} x_{2t} &= l_t - \frac{1}{4}\big(MC^{-1}(h_{t-1}, mv^-_{2t-1}) + MC^{-1}(h_t, mv^+_{2t+1})\big), \\ x_{2t+1} &= h_t + \frac{1}{2}\big(MC(x_{2t}, mv^+_{2t+1}) + MC(x_{2t+2}, mv^-_{2t+1})\big), \end{aligned} \tag{13.5}$$

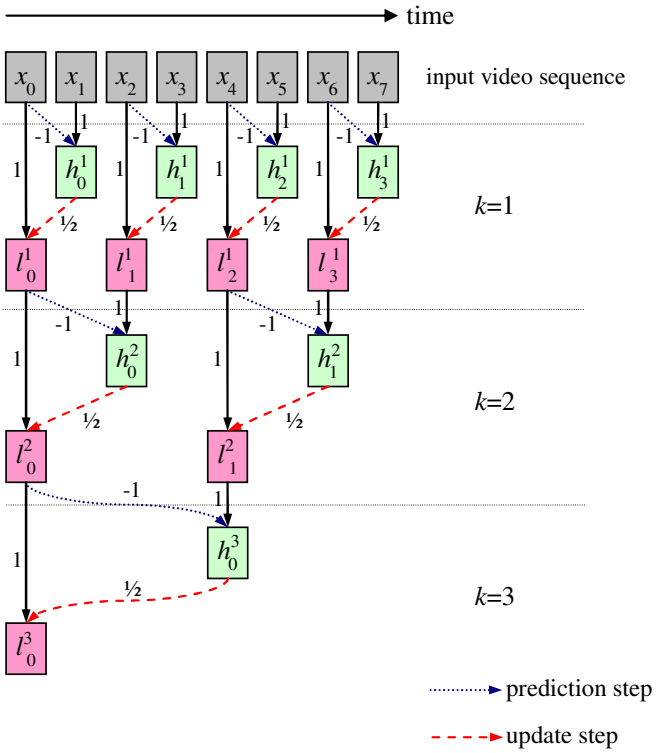


Fig. 13.3. Scheme of temporal wavelet analysis based on Haar filters

where x_t is the picture of the input sequence in time t, h_t is a sample of a high-band signal in time t, l_t is a sample of a low-band signal in time t, mv^{+}_{2t+1} is the set of motion vectors between the pictures x_{2t+1} and x_{2t}, mv^{-}_{2t+1} is the set of motion vectors between the pictures x_{2t+1} andx_{2t+2}, (m, n) are spatial coordinates of the picture, $MC(x_t, mv_t)$ is the motion compensation operator, defined as $x_t(m, n - mv_t(m, n))$.

Figure 13.4 illustrates temporal filtering for three-level temporal decomposition based on LeGall filters. The dotted arrow indicates the prediction step and the dashed one – the update step.

13.2.2. Temporal filtering delay

Delays occurring in video sequence coding are caused by the necessity of waiting for future pictures. There are two types of such delays:

- encoding delay D_E – a delay occurring during the encoding of a sequence,
- coding delay D_C – total delays in the entire coding system (coding and decoding).

For temporal filtering based on Haar filters, the delays are described by the equations (Ohm, 2002; Pau *et al.*, 2005):

$$D_E = 2^k - 1, \tag{13.6}$$

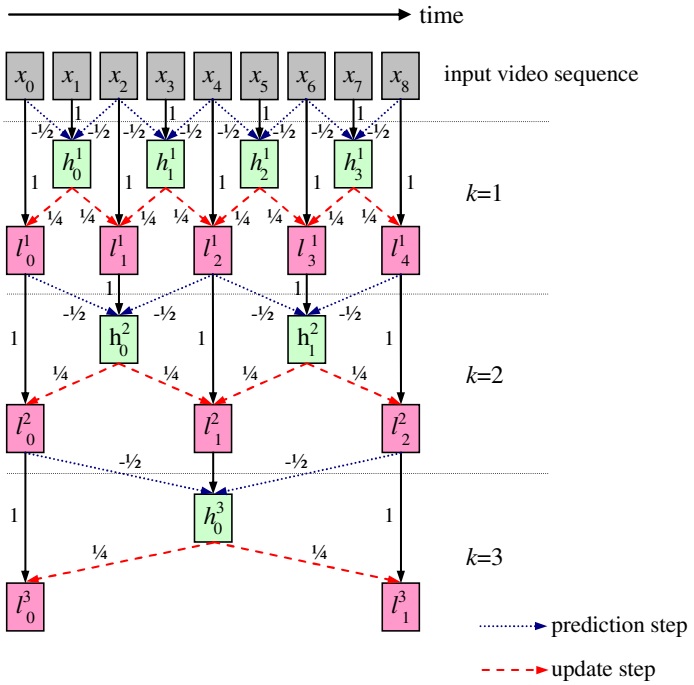


Fig. 13.4. Scheme of temporal wavelet analysis based on 5/3 filters

$$D_C = \sum_{i=1}^{k} 2^{i-1}, \tag{13.7}$$

where k is the number of temporal decomposition levels.

With the use of Eqns. (13.6), (13.7), it is possible to determine the delay expressed by the number of the pictures of a sequence. Equation (13.8) allows expressing the delay in milliseconds:

$$D_T = \left\lfloor \frac{D}{f} \cdot 1000 \right\rfloor, \tag{13.8}$$

where D denotes delay expressed in the number of the pictures, f is the frequency of the pictures in a sequence.

Table 13.1 presents a comparison of Haar filters temporal delays (in the number of pictures and milliseconds) for various numbers of temporal decomposition levels k.

Figure 13.5 explains the reason encoding delay emerging in the case of Haar filters and three-level temporal decomposition. Bold lines indicate operations introducing delays.

As presented in Table 13.1, encoding delay in the discussed case (k=3) is seven pictures. It is implied by the necessity of waiting for the h_0^3 component, and to obtain it are required h_1^2 and h_3^1 components and the original image of the input sequence x_7. In this case you can observe that resignation from the last update step (bold line for $k = 3$) reduces the encoding delay to three pictures.

Table 13.1. Delays introduced by wavelet coders using Haar filters for various numbers of the temporal decomposition levels k and the frequency $f = 30$ Hz

k	encoding delay D_E		coding delay D_C	
	[number of pictures]	[ms]	[number of pictures]	[ms]
1	1	34	1	34
2	3	100	3	100
3	7	234	7	234
4	15	500	15	500
5	31	1034	31	1034
6	63	2100	63	2100

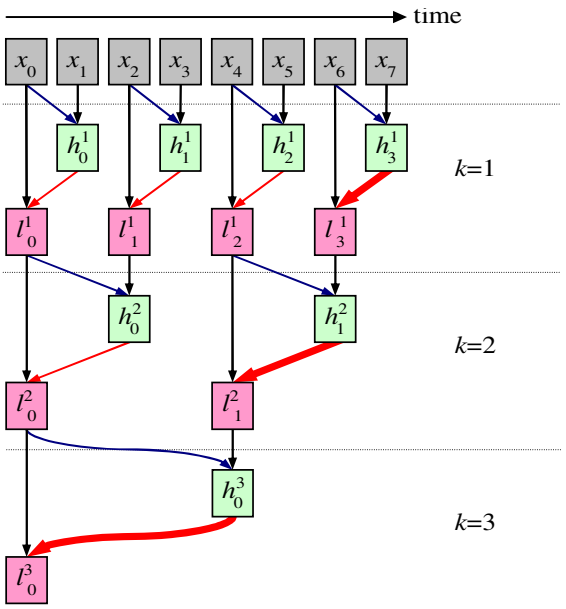


Fig. 13.5. Temporal delays scheme for Haar filters

In the case of temporal filtering with the use of LeGall 5/3 filters, encoding and coding delays are described by the equations (Ohm, 2002; Pau *et al.*, 2005):

$$D_E = 2^{k+1} - 2, \tag{13.9}$$

$$D_C = 3 \cdot (2^k - 1), \tag{13.10}$$

where k is the number of temporal decomposition levels.

Table 13.2 presents a comparison of LeGall 5/3 filters temporal delays for various numbers of temporal decomposition levels k.

Table 13.2. Delays introduced by wavelet coders using 5/3 filters for various numbers of the temporal decomposition levels k and the frequency $f = 30$ Hz

k	encoding delay D_E		coding delay D_C	
	[number of pictures]	[ms]	[number of pictures]	[ms]
1	2	67	3	100
2	6	200	9	300
3	14	467	21	700
4	30	1000	45	1500
5	62	2067	93	3100
6	126	4200	189	6300

Figure 13.6 explains the reason for encoding delay emerging in the case of Haar filters and three-level temporal decomposition. The bold lines indicate operations introducing delays. As presented in Table 13.2, encoding delay in discussed case ($k = 3$) is fourteen pictures. It is implied by the necessity of waiting for the h_0^3, l_2^2, h_2^2 and l_6^1 components and the original image of the input sequence x_{14}. In this case you can observe that resignation from the last update step (bold dashed line for $k = 3$) reduces the encoding delay to ten pictures. If we decide not to perform the update step for future components in two last decomposition steps ($k = 2, 3$), the encoding delay will be reduced to six pictures.

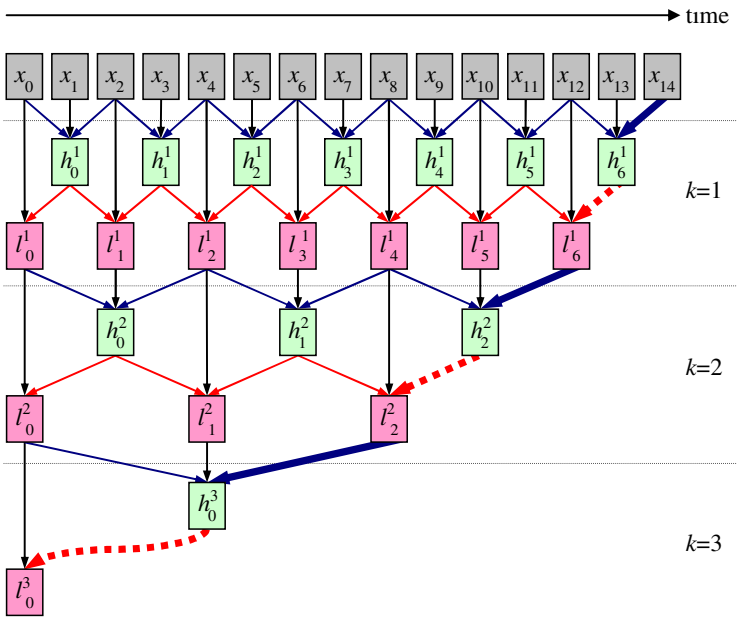


Fig. 13.6. Temporal delays scheme for 5/3 filters

13.2.3. Estimation of results

The effectiveness of video sequence temporal compression for different filtering schemes was estimated by performing a series of experiments with a set of test video sequences. The first step was to encode and then decode the test sequence with the assumed bitrate. The resulting signal was compared with the original one and the quality was confronted. The test sequences used were represented in the YUV format, comprising the luminance component Y and two chrominance components U and V, with the sampling scheme 4:2:0 (ITU-R Rec. BT.470-3).

To measure the difference, the PSNR (*Peak Signal-to-Noise Ratio*) was used defined as (Domański, 1998; Pau *et al.*, 2004):

$$PSNR = 10 \log \frac{255^2 N^2}{\sum\limits_i e_i^2} \text{ [dB]}, \tag{13.11}$$

where 255 is the dynamic range of the signal, N is the number of picture elements, e_i is the difference between the i point of the original and processed image.

As subjective quality measures (ITU-R Rec. BT.500-6) take a lot of time and effort, they were not carried out. Since the codecs compared use similar compression methods, the type of distortion introduced is approximate and according to (ITU-R Rec. BT.813) the PSNR is a satisfactory measure to estimate the results. Measurements were performed for the luminance component by calculating the mean value of the PSNR for all pictures in a given sequence. This method is widely applied by many researchers.

For the experiments, nine test sequences were used: *City, Crew, Harbour, Ice, Soccer, Football, Silent, Mobile, Foreman* in the formats CIF 15 Hz (352×288), CIF 30 Hz (352×288), 4CIF 30 Hz (704×576) and of length 6,4 seconds (192 pictures for 30 Hz and 96 pictures for 15 Hz). The sequences selected are commonly used to assess video sequences compression effectiveness.

Since there is no standardized wavelet codec, it was important to select a proper codec for the research. As a software basis for experiments, the MC-EZBC (*Motion Compensation Embedded Zero Block Coding*) codec was selected (Hsiang and Woods, 2001; Rusert *et al.*, 2004). This complex codec (approx. 43 thousand lines of the C code) is widely used as a reference wavelet video codec.

13.3. Reduction of coding delay

The main reason behind coding delay is the necessity of waiting for pictures that at the given moment are not available yet. If we decide not to perform some operations, referring to such "future" images, e.g., by omitting some prediction and update steps in the temporal lifting filtering scheme, we will reduce or even eliminate the delay. In a practical approach of the wavelet coders using Haar and 5/3 filters, the reduction of the coding delay can be achieved by using one of possible schemes of prediction and update steps elimination.

In the case of Haar filters the prediction step does not introduce delay. The only way to reduce the delay is to eliminate the update step and thus to break the reference

to future pictures. Figure 13.7 presents the update components (marked bold) causing the delay.

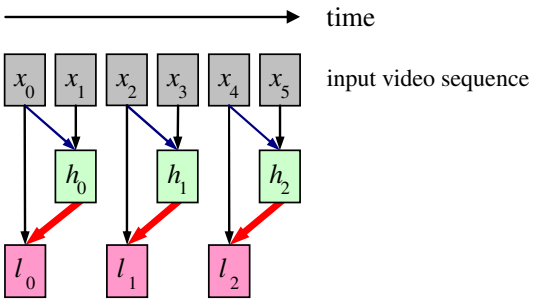


Fig. 13.7. Update components responsible for delay in the case of Haar filters

For 5/3 filters both prediction and update steps cause the delay. Figure 13.8 illustrates the prediction and update components (marked bold) causing the delay.

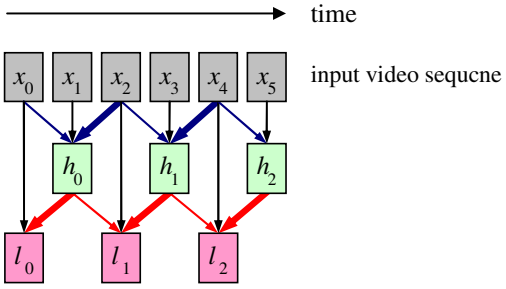


Fig. 13.8. Prediction and update components responsible for delay in the case of 5/3 filters

13.3.1. Modified filtering schemes

Figure 13.9 presents the temporal wavelet analysis and synthesis scheme for modified Haar filters, with the update step removed (Schwarz *et al.*, 2004a; Pau *et al.*, 2005). For the presented filtering structure, the analysis and synthesis equations are respectively

$$h_t = x_{2t+1} - MC(x_{2t}, mv^{+}_{2t+1}), \quad l_t = x_{2t}, \tag{13.12}$$

$$x_{2t} = l_t, \quad x_{2t+1} = h_t + MC(x_{2t}, mv^{+}_{2t+1}). \tag{13.13}$$

Figure 13.10 presents the temporal wavelet analysis and synthesis scheme for modified Haar filters, with the update step referring to the previous image (Pau *et al.*, 2004).

For the presented filtering structure, the analysis and synthesis equations are respectively

$$h_t = x_{2t+1} - MC(x_{2t}, mv^{+}_{2t+1}), \quad l_t = x_{2t} + \frac{1}{2}MC^{-1}(h_{t-1}, mv^{-}_{2t+1}), \tag{13.14}$$

$$x_{2t} = l_t - \frac{1}{2}MC^{-1}(h_{t-1}, mv^{-}_{2t+1}), \quad x_{2t+1} = h_t + MC(x_{2t}, mv^{+}_{2t+1}). \tag{13.15}$$

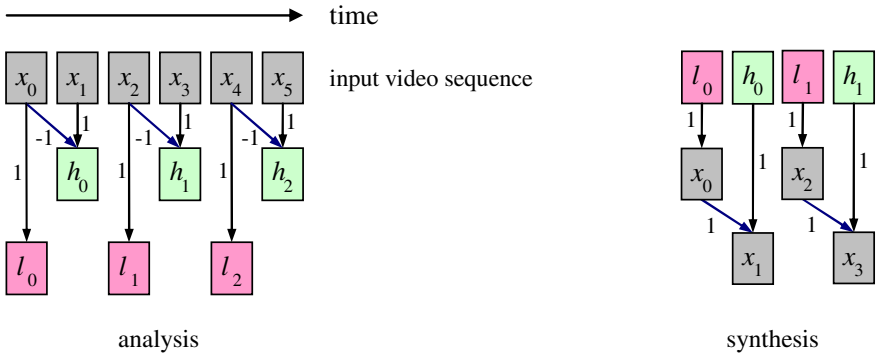


Fig. 13.9. Filtering scheme for modified Haar filters with the update step removed

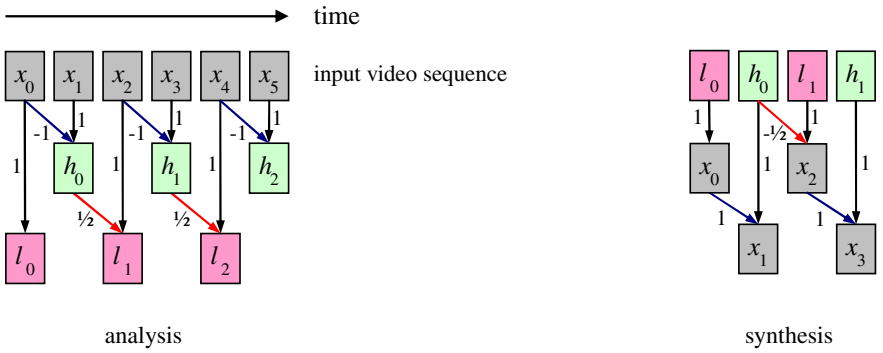


Fig. 13.10. Filtering scheme for modified Haar filters with the update step relating to the previous image

The scheme can also be interpreted as a 5/3 filter with both prediction and update steps removed, referring to the future pictures. It is important to note that this scheme requires two sets of motion vectors to be calculated: one for the prediction step and one for the update step.

Figure 13.11 presents the temporal wavelet analysis and synthesis scheme for 5/3 filters, with the prediction step removed, referring to the future picture.

For the presented filtering structure, the analysis and synthesis equations are respectively

$$\begin{aligned} h_t &= x_{2t+1} - MC(x_{2t}, mv^{+}_{2t+1}), \\ l_t &= x_{2t} + \frac{1}{4}\big(MC^{-1}(h_{t-1}, mv^{-}_{2t-1}) + MC^{-1}(h_t, mv^{+}_{2t+1})\big), \end{aligned} \tag{13.16}$$

$$\begin{aligned} x_{2t} &= l_t - \frac{1}{4}\big(MC^{-1}(h_{t-1}, mv^{-}_{2t-1}) + MC^{-1}(h_t, mv^{+}_{2t+1})\big), \\ x_{2t+1} &= h_t + MC(x_{2t}, mv^{+}_{2t+1}). \end{aligned} \tag{13.17}$$

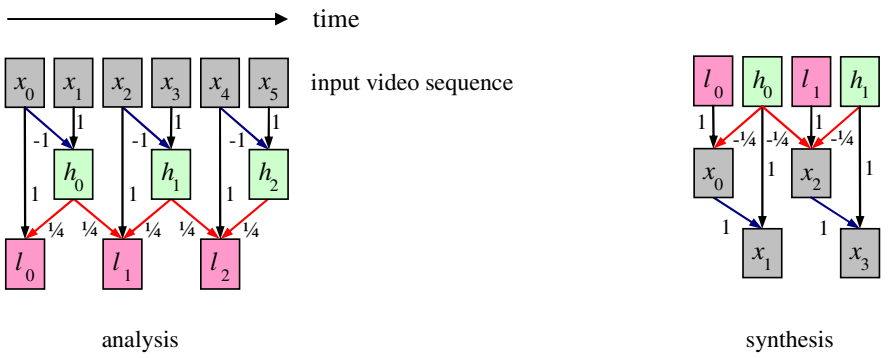


Fig. 13.11. Filtering scheme for modified 5/3 filters with the prediction step removed, relating to the future picture

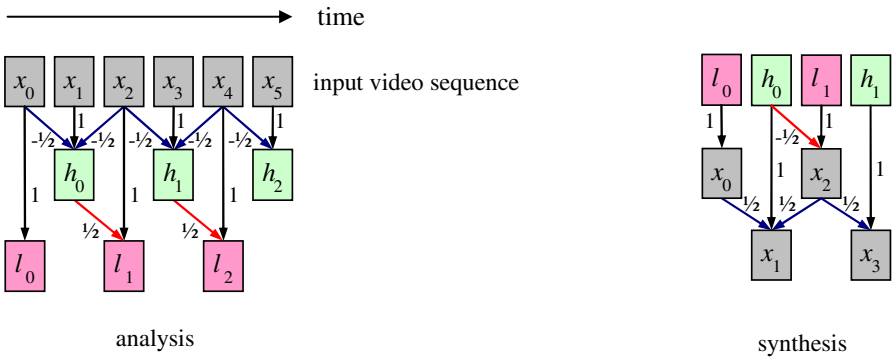


Fig. 13.12. Filtering scheme for modified 5/3 filters with the update step removed, relating to the future picture

Figure 13.12 presents the temporal wavelet analysis and synthesis scheme for 5/3 filters, with the update step removed, referring to the future picture. For the presented filtering structure, the analysis and synthesis equations are respectively

$$\begin{aligned} h_t &= x_{2t+1} - \frac{1}{2}\big(MC(x_{2t}, mv^{+}_{2t+1}) + MC(x_{2t+2}, mv^{-}_{2t+1})\big), \\ l_t &= x_{2t} + \frac{1}{2} MC^{-1}(h_{t-1}, mv^{-}_{2t-1}), \end{aligned} \tag{13.18}$$

$$\begin{aligned} x_{2t} &= l_t - \frac{1}{2} MC^{-1}(h_{t-1}, mv^{-}_{2t-1}), \\ x_{2t+1} &= h_t + \frac{1}{2}\big(MC(x_{2t}, mv^{+}_{2t+1}) + MC(x_{2t+2}, mv^{-}_{2t+1})\big). \end{aligned} \tag{13.19}$$

By analysing the above schemes of temporal wavelet filtering for a single temporal decomposition level, there were identified six modified temporal filtering schemes. The reduction of coding delay in these schemes is achieved by eliminating the prediction

or update steps and breaking the reference to future pictures of the sequence. In this chapter we will assign two-symbol names to particular schemes:

- SH – filtering with original Haar filters,
- SU – filtering with Haar filters with the update step removed,
- SB – filtering with Haar filters with the update step referring to the previous picture,
- SP – filtering with 5/3 filters with the prediction step removed, referring to the future picture,
- S3 – filtering with 5/3 filters with the update step removed, referring to the future picture,
- S5 – filtering with original 5/3 filters.

After initial experiments we decided to eliminate from further research the SP scheme, since PSNR assessment showed that removing the prediction step is significantly less advantageous than removing the update step (S3 scheme) and both solutions are of the same coding delay.

The remaining schemes were used to construct 21 variants of wavelet analysis for 8 values of coding delay in four temporal decomposition levels.

On the basis of the previous research (Popławski, 2006), the picture group size was assumed to be 16, as assuring the best compression efficiency for most of the test sequences, various spatial and temporal resolutions and transmission speeds.

13.3.2. Experimental results

The research was carried out for ten transmission speeds, and motion vectors were calculated with the accuracy of a 1/4 spatial sampling interval. For the experiments, nine test sequences in the formats CIF 15 Hz, CIF 30 Hz, 4CIF 30 Hz were taken. For easier identification, the coding variants were given four-symbol names, depending on the filtering scheme on the subsequent decomposition level. For example symbol 53 HU means filtering with the use of the following schemes:

- S5 – on the first temporal decomposition level,
- S3 – on the second temporal decomposition level,
- SH – on the third temporal decomposition level,
- SU – on the fourth temporal decomposition level.

The examined delays were 0, 1, 3, 5, 7, 9, 13 and 15 pictures (temporal sampling intervals). The values selected result from the temporal filtering schemes applied.

For a given coding delay it is possible to use different variants of temporal wavelet analysis. For example, for a delay of nine pictures there can be applied 533U, 53HU and 55UU schemes (see Table 13.5), presented in Figs. 13.13–13.15.

The main aim of the research was to find the most effective (in terms of compression effectiveness) scheme of temporal wavelet coding for a given coding delay value.

The reduction of coding delay, implying the use of modified filtering schemes, has significant influence on coding efficiency. The reduction of coding delay results in an

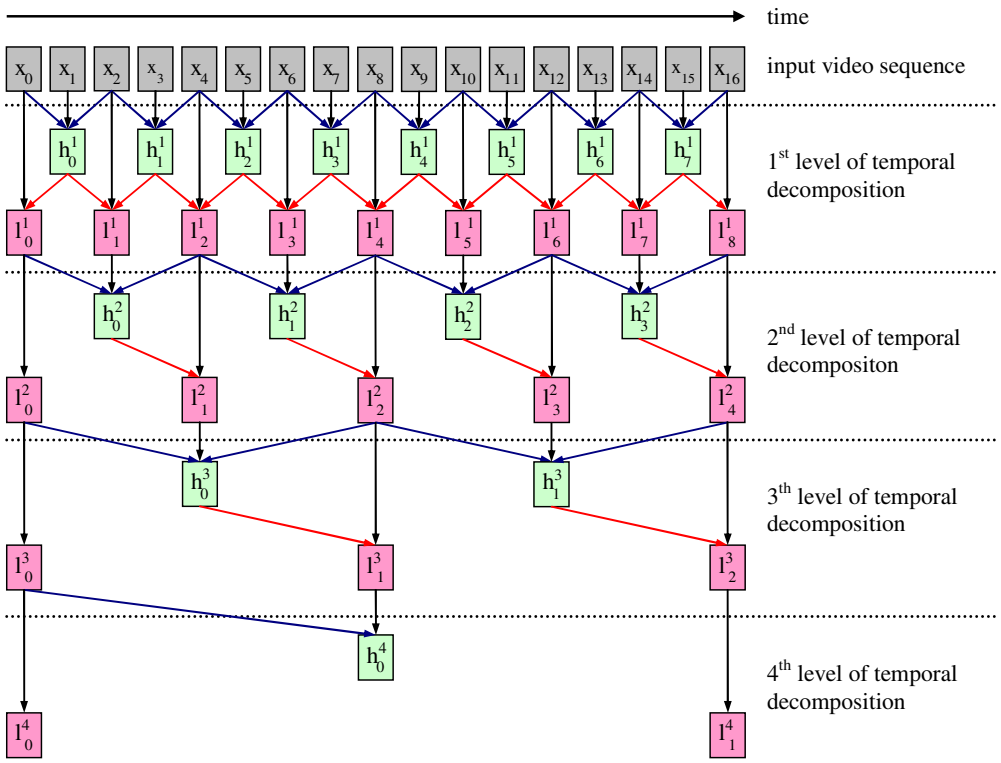


Fig. 13.13. Wavelet analysis 533U scheme with the coding delay of 9 pictures

accumulative PSNR decrease. Table 13.3 presents average results of PNSR decrease (for ten examined transmission speeds) of the best filtering schemes and for given coding delay versus filtering with no coding delay restrictions (5555 scheme, shown in Fig. 13.16 – unmodified 5/3 filters on all decomposition levels, coding delay of 45 pictures).

The least decrease in coding efficiency occurred for the *Footbal*l sequence, the average PSNR decrease (for ten transmission speeds) was 0,79 dB for zero coding delay. The biggest average decrease was observed for the *Mobile* sequence (4,33 dB). For all test sequences the worst coding efficiency was observed for no coding delay (Fig. 13.17).

The results discussed above were presented from the viewpoint of a loss of coding efficiency versus the reduction of coding delay, and the measure was the PSNR. It is interesting to compare the results in some other way: for the constant compression quality (measured with the PSNR) we will examine a transmission speed increase (in percent), regardless the coding delay. As a reference we take the PSNR of the scheme with no coding delay constrains (5555 scheme), for each test sequence separately:

– 256 kb/s for CIF 15Hz,

– 512 kb/s for CIF 30Hz,

– 1152 kb/s for 4CIF 30Hz.

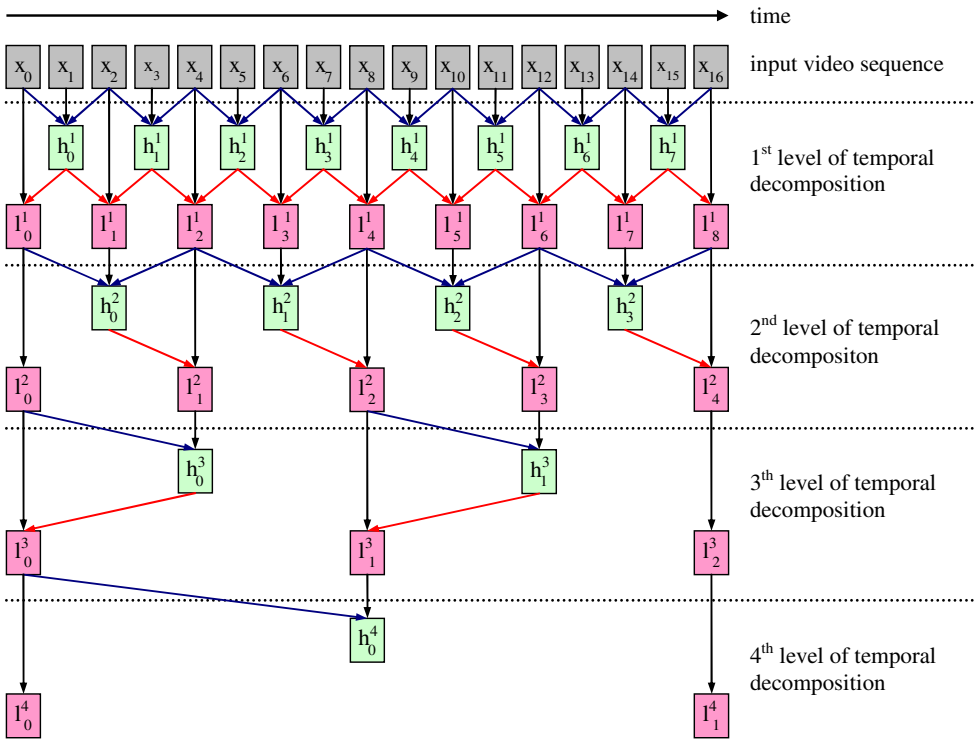


Fig. 13.14. Wavelet analysis 53HU scheme with the coding delay of 9 pictures

Table 13.3. Average PSNR loss (for the assumed coding delays) vs. the solution with no delay constraints for the CIF resolution of 30 Hz

Test sequence	PSNR loss [dB] / Coding delay							
	15	13	9	7	5	3	1	0
City	0,71	1,00	1,19	1,32	1,95	2,16	2,98	3,96
Crew	0,35	0,00	0,14	0,30	0,26	0,41	0,77	1,47
Harbour	0,89	0,85	1,07	1,28	1,66	1,86	2,61	3,36
Ice	0,57	0,41	0,56	0,75	0,87	1,06	1,61	2,55
Soccer	0,48	0,37	0,49	0,63	0,88	1,00	1,60	2,45
Football	0,20	0,02	0,12	0,21	0,11	0,21	0,28	0,79
Silent	0,44	0,51	0,63	0,76	1,02	1,15	1,66	2,29
Mobile	0,65	1,13	1,19	1,54	2,21	2,33	3,35	4,33
Foreman	0,60	0,54	0,72	0,91	1,04	1,21	1,80	2,50
Average	*0,55*	*0,54*	*0,68*	*0,86*	*1,11*	*1,26*	*1,85*	*2,63*

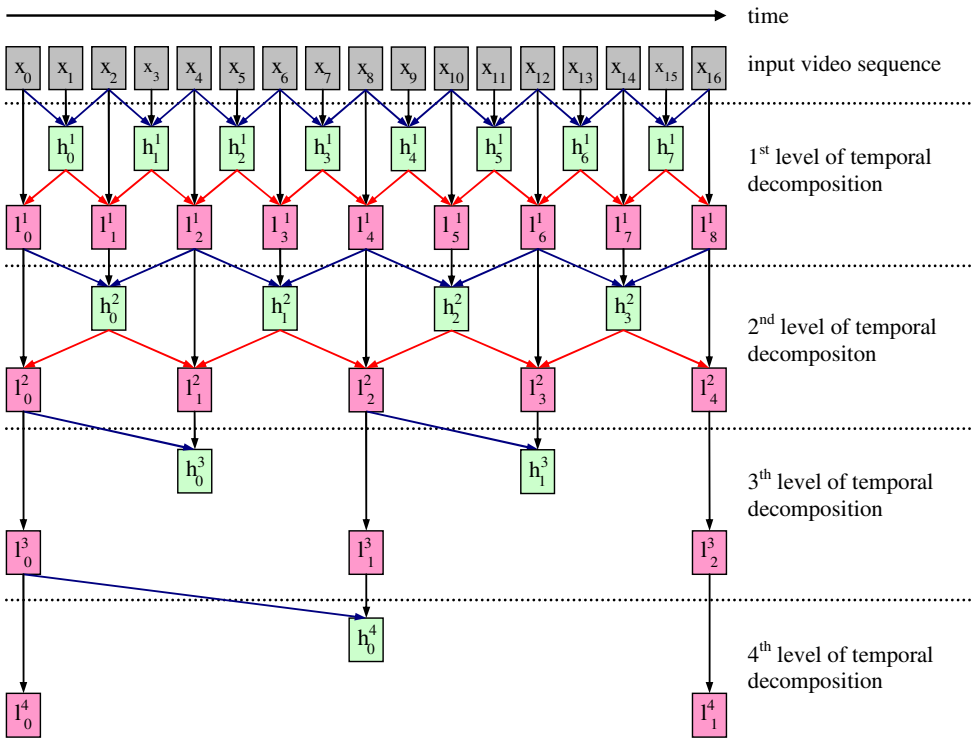


Fig. 13.15. Wavelet analysis 53UU scheme with the coding delay of 9 pictures

Transmission speed for a given PSNR for subsequent delays was obtained by linear interpolation, based of two PSNR values closest to the reference point. The results are presented in Table 13.4.

Limiting coding delay leads to a bitstream size increase to keep the compression quality at the required level. The percentage of this increase is rather smooth for delays between 15 and 3 pictures. For a delay of 0 or 1 the increase in the require transmission speed is more intense. Figure 13.18 presents the diagram of an average percentage of the transmission speed increase for a constant PSNR and a given coding delay.

13.4. Conclusions

The research results allow selecting the coding schemes of the best coding efficiency for a predefined coding delay. Table 13.5 presents a comparison of compression effectiveness of the temporal analysis schemes mentioned above. The most effective solutions are collected in Group I and the less effective in Group III.

There can be formed a general rule of selecting temporal wavelet analysis schemes to obtain high compression efficiency. The first attempt is to use the S5 scheme, as having the higher coding efficiency. If the coding delay of the S5 scheme is too high, there should be used the S3 scheme. If the coding delay is still unacceptable, there

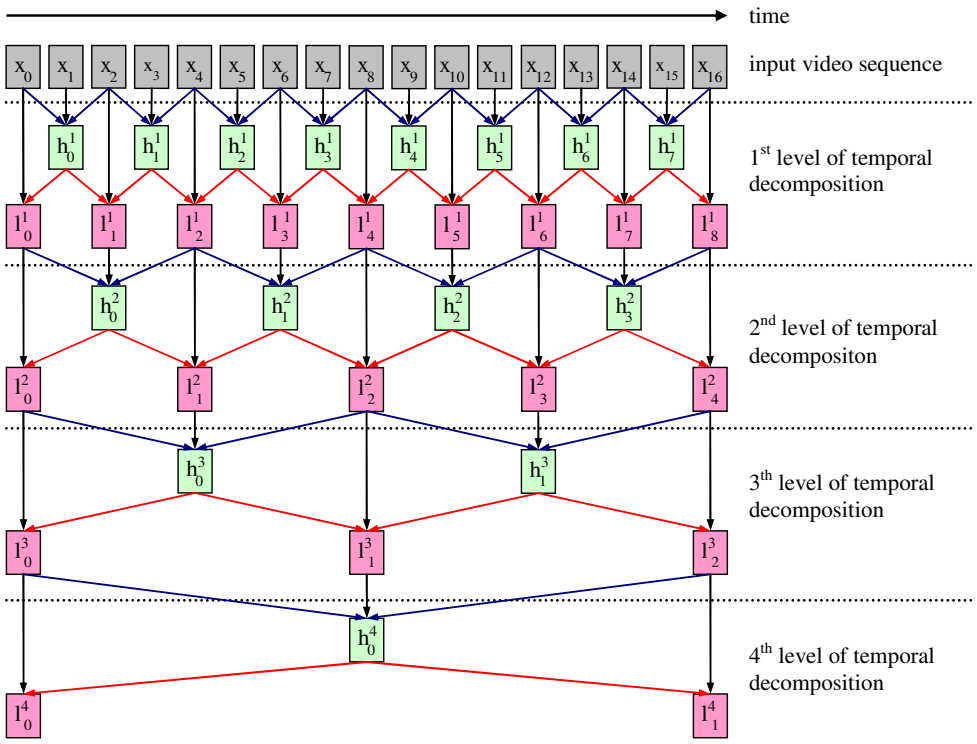


Fig. 13.16. Wavelet analysis 5555 scheme with the coding delay of 45 pictures

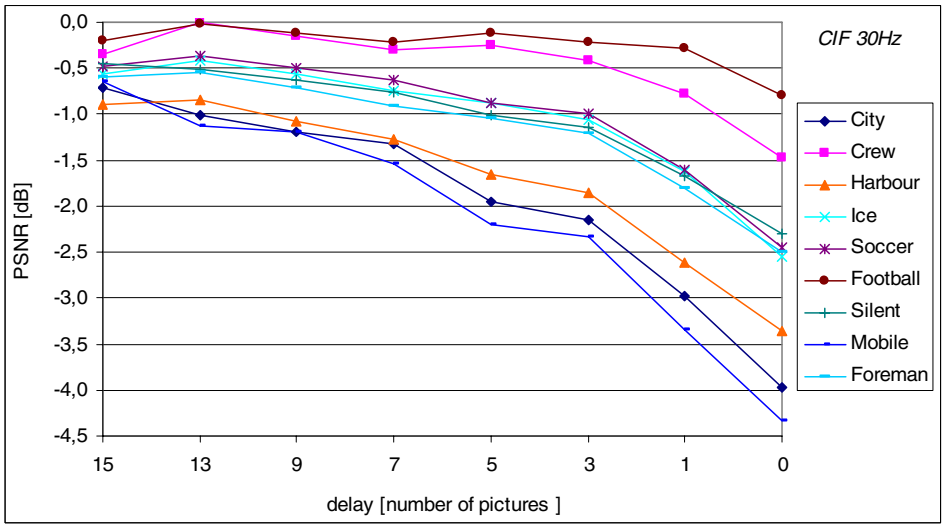


Fig. 13.17. Average PSNR loss (for the assumed coding delays) vs. the solution with no delay constrainst for the CIF resolution of 30 Hz

Table 13.4. Transmission speed increase (in percent) for a constant PSNR value and the assumed coding delay for the CIF resolution of 30 HZ

Test sequence	Transmission speed increase [%] / Coding delay							
	15	13	9	7	5	3	1	0
City	12,0%	21,3%	24,6%	27,7%	45,7%	51,4%	80,6%	128,4%
Crew	8,9%	0,3%	3,6%	7,3%	7,5%	11,0%	22,6%	47,2%
Harbour	22,2%	22,9%	28,9%	34,7%	48,4%	55,2%	88,7%	128,1%
Ice	10,2%	7,9%	10,3%	12,8%	15,0%	17,9%	28,5%	49,3%
Soccer	9,4%	7,7%	10,0%	13,1%	18,8%	21,5%	36,4%	61,8%
Football	4,7%	0,8%	3,1%	5,0%	3,3%	5,5%	8,2%	22,4%
Silent	6,6%	8,6%	10,2%	11,9%	16,7%	18,8%	28,2%	42,9%
Mobile	17,6%	31,6%	33,4%	43,3%	67,1%	70,1%	111,4%	170,6%
Foreman	12,5%	12,4%	16,0%	20,4%	23,6%	28,2%	46,6%	73,8%
Average	*11,6%*	*12,6%*	*15,6%*	*19,6%*	*27,4%*	*31,1%*	*50,1%*	*80,5%*

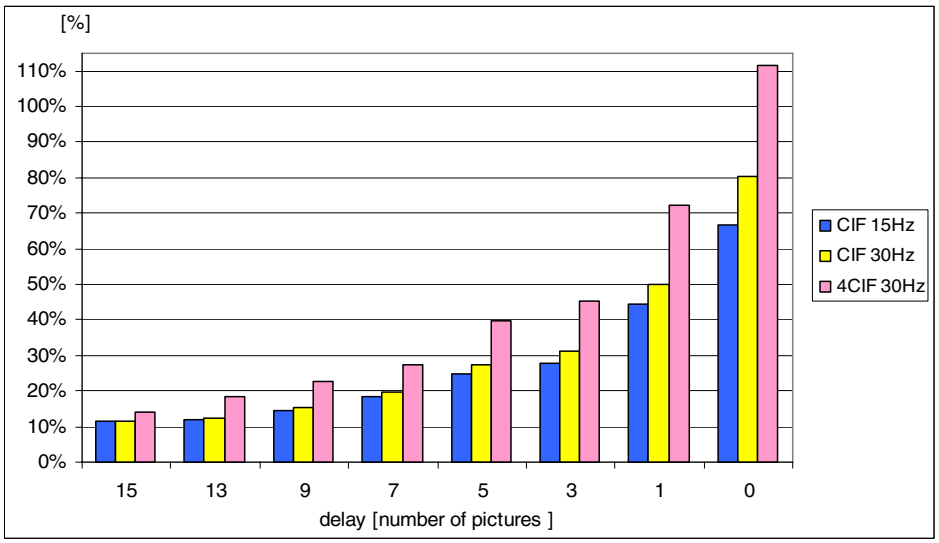


Fig. 13.18. Transmission speed increase (average over all tested sequences) for a constant PSNR and the assumed coding delay vs. the solution with no coding delay constraints

should be used the SU scheme. The SB and SP schemes are of significantly worse compression effectiveness and they are not recommended.

It would be interesting to note that there is a relatively small difference in coding effectiveness of the schemes:

– 53UU and 5HUU (5 pictures delay),

Table 13.5. Compression effectiveness of the investigated temporal filtering schemes

Coding delay [number of pictures]	Group I (most effective schemes)	Group II	Group III (least effective schemes)
0	UUUU	UBBB	BBBB
1	3UUU	HUUU	—
3	33UU	5UUU	HHUU
5	53UU	5HUU	—
7	333U	33HU	HHHU
9	533U	53HU	55UU
13	553U	55HU	—
15	3333	33HH	HHHH

– 333U and 33HU (7 pictures delay),

– 533U and 53HU (9 pictures delay),

– 553U and 55HU (13 pictures delay).

Schemes of lower coding effectiveness have lower numerical cost (there are fewer sets of motion vectors to calculate), and that feature can be useful in some cases.

Figure 13.17 illustrates the influence of coding delay constrains (on average over the range of transmission speeds) on the coded sequence quality when the best wavelet scheme is applied. Table 13.4 shows the relative increase in the required transmission speed, implied by the limitation of an acceptable coding delay.

The conclusion of the results is presented in Fig. 13.18, showing the relative increase in the required transmission speed, average over nine sequences and ten transmission speeds. These results can be used to estimate the unavoidable increase in the required transmission speed implied by the limitation of an acceptable coding delay.

References

Ahmed N., Natarajan T. and Rao K.R. (1974): *Discrete cosine transform.* — IEEE Trans. Computer, Vol. C-23, No. 1, pp. 90–93.

Chen P. (2003): *Fully scalable subband/wavelet coding.* — doctoral thesis, Rensselaer Polytechnic Institute, Troy, New York.

Choi S.J. and Woods J.W. (1999): *Motion-compensated 3-D subband coding of video.* — IEEE Trans. Image Processing, Vol. 8, No. 2, pp. 155–167.

Claypoole R.L., Davis G.M., Sweldens W. and Baraniuk G. (2003): *Nonlinear wavelet transforms for image coding via lifting.* — IEEE Trans. Image Processing, Vol. 12, No. 12, pp. 1449–1459.

Daubechies I. and Sweldens W. (1996): *Factoring wavelet transforms into lifting steps.* — Bell Laboratories, Lucent Technologies.

Domański M. (1998): *Advanced Image and Video Compression Techniques.* — Poznań: Technical University Press, (in Polish).

Hsiang S.-T. and Woods J.W. (2000): *Embedded image coding using zeroblocks of subband/wavelet coefficients and contest modelling.* — Proc. IEEE Int. Symp. *Circuits and Systems*, Geneva, Switzerland, Vol. 3, pp. 662–665.

Hsiang S.-T. and Woods J.W. (2001): *Embedded video coding using invertible motion compensated 3-D subband/wavelet filter bank.* — Signal Processing: Image Communication, Vol. 16, No. 8, pp. 705–724.

Huang L.M., Mei S.S. and Honda Y. (2003): *Results on Scalable Video Coding in Low Delay Mode.* — ISO/IEC JTC1/SC29/WG11, Doc. M9843, MPEG 2003.

ISO/IEC International Standard 13818 (1994), Information Technology – Generic Coding of Moving Pictures and Associated Audio Information.

ISO/IEC 14496-10 AVC / ITU-T Rec. H.264, Text for ISO/IEC 14496-10:2005 (AVC 3rd Edition), Information Technology – Coding of audio-visual objects – Part 10: Advanced Video Coding, MPEG 2005.

ITU-T Rec. H.261 (1990), Video codec for audiovisual services at p×64 kbit/s.

ITU-T Rec. H.263 (1996), Video coding for Low Bit Rate Communication.

ITU-R Rec. BT.470-3 (1994), Television systems.

ITU-R Rec. BT.500-6 (1994), Methodology for the subjective assessment of the quality of television pictures.

ITU-R Rec. BT.813 (1992), Methods for objective picture quality assessment in relation to impairments from digital coding of television signal.

Jayant N. and Noll P. (1984): *Digital Coding of Waveforms.* — Englewood Cliffs, NJ: Prentice-Hall.

LeGall D. and Tabatabai A. (1988): *Subband coding of digital images using symmetric short kernel filters and arithmetic coding techniques.* — Proc. IEEE, Int. Conf. *Acoustics, Speech and Signal Processing*, New York, USA, pp. 761–765.

Li Z.G., Lim K.P., Lin X. and Rahardja S. (2005): *Customer oriented low delay scalable video coding.* — ISO/IEC JTC1/SC29/WG11, Doc. M11544, MPEG 2005.

Ohm J.-R. (1992): *Temporal domain subband video coding with motion compensation.* — Proc. IEEE Int. Conf. *Acoustics, Speech and Signal Processing*, San Francisco, CA, USA, Vol. 3, pp. 229–232.

Ohm J.-R. (1993): *Three-dimensional motion-compensated subband coding.* — Proc. *SPIE Video Communications and PACS for Medical Applications*, Vol. 1977, pp. 188–197.

Ohm J.-R. (1994): *Three-dimensional subband coding with motion compensation.* — IEEE Trans. Image Processing, Vol. 3, No. 5, pp. 559–571.

Ohm J.-R. (2002): *Complexity and delay analysis of MCTF interframe wavelet structures.* — ISO/IEC JTC1/SC29/WG11, Doc. M8520, MPEG 2002.

Pau G., Pesquet-Popescu B., van der Schaar M. and Viéron J. (2004): *Delay-performance trade offs in motion-compensated scalable subband video compression.* — ISO/IEC JTC1/SC29/WG11, Doc. 11084, MPEG 2004.

Pau G., Viéron J. and Pesquet-Popescu B. (2005): *Video coding with flexible MCTF structures for low end-to-end delay.* — Proc. Int. Conf. *Image Processing*, Genova, Vol. 3, pp. 241–244.

Popławski A. (2006): *Low delay three-dimensional wavelet coding of video sequences.* — Doctoral thesis, Faculty of Electrical Engineering, Poznań University of Technology, (in Polish).

Rabbani M. and Jones P.W. (1991): *Digital Image Compression.* — Bellingham, Washington: SPIE Opt. Eng. Press.

Rusert T., Hanke K. and Wien M. (2004): *Optimization for locally adaptive MCTF based on 5/3 lifting.* — Proc. Symp. *Picture Coding*, San Francisco, CA, USA, pp. 210–220.

Schwarz H., Marpe D. and Wiegand T. (2004a): *SVC Core Experiment 2.2: Influence of the update step on the coding efficiency.* — ISO/IEC JTC1/SC29/WG11, Doc. M11048, MPEG 2004.

Schwarz H., Shen J., Marpe D. and Wiegand T. (2004b): *Technical description of the HHI proposal for SVC CE3.* — ISO/IEC JTC1/SC29/WG11, Doc. M11246, MPEG 2004.

Seran V. and Kondi L.P. (2005): *3D based video coding in the overcomplete discrete wavelet transform domain with reduced delay requirements.* — Proc. Int. Conf. *Image Processing*, Genova, Vol. 3, pp. 233–236.

Strang G. and Nguyen T. (1996): *Wavelets and filter banks.* — Wellesley: Cambridge Press.

Sweldens W. (1996): *The lifting scheme: A custom-design construction of biorthogonal wavelets.* — Appl. Comput. Harmon. Anal., Vol. 3, No. 2, pp. 186–200.

Sweldens W. and Schröder P. (1996): *Building your own wavelets at home.* — Wavelets in Computer Graphics, Vol. 1, ACM SIGGRAPH Course Notes, pp. 15–87.

Topiwala P.N. (1998): *Wavelet Image and Video Compression.* — Boston, MA: Kluwer Academic Publishers.

Vaidyanathan P.P. (1993): *Multirate Systems and Filter Banks.* — Englewood Cliffs, NJ: Prentice Hall.

Vetterli M. and Kovačević J. (1995): *Wavelets and Subband Coding.* — Englewood Cliffs, NJ: Prentice-Hall.

Viéron J., Boisson G., François E., Pau G. and Pesquet-Popescu B. (2005): *Time and level adaptive MCTF architectures for low delay video coding.* — ISO/IEC JTC1/SC29/WG11, Doc. M11673, MPEG 2005.

Woods J.W. (1991): *Subband Image Coding.* — Norwell, MA: Kluwer Academic Publishers.

Xu J., Xiong Z., Li S. and Zhang Y. (2001): *Three-dimensional embedded subband coding with optimized truncation (3D ESCOT).* — Appl. Computat. Harmonic Anal., Vol. 10, No. 3, pp. 290–315.

Chapter 14

SAFE RECONFIGURABLE LOGIC CONTROLLERS DESIGN

Marian ADAMSKI*, Marek WĘGRZYN*, Agnieszka WĘGRZYN*

14.1. Introduction

To describe digital systems, designers frequently adapt a concurrent and distributed view of the modeled behavior. Petri Nets (PNs) provide a mechanism which is suited for representing parallelism and hierarchy in complex digital processes (Cortadella *et al.*, 2002; Murata, 1989). The control part of the system (concurrent controller, parallel controller (Adamski, 1991)) is described and verified using the well-developed Petri net theory, and its specification can be translated through automated processes into a format accepted by selected FPGA (Field Programmable Gate Array) and CPLD (Complex Programmable Logic Device) synthesis tools. Presently Petri nets are used both as specification and synthesis models for reconfigurable logic controllers design, which are frequently embedded inside modern, reactive microsystems.

The main aim of this chapter is to demonstrate a practical, direct method of mapping concurrent digital systems into Field Programmable Logic (FPL) during the design process of logic controller design. The paper gives also an overview of selected papers related to hardware implementation of Petri nets. The experimental results have shown that the presented novel approach may produce economical FPL implementations of application-specific reconfigurable logic controllers.

A Petri net is considered as a formal model for logic rules (logic relations), as well as a model for HDL (Hardware Description Language) description. The net can be first expressed graphically, or it is given directly as a set of rules in Petri net specification formats, and then automatically translated into an equivalent program in VHDL (Very high speed integrated circuits Hardware Description Language) or Verilog (Węgrzyn M., 2003; Wolański *et al.*, 1997).

Behavioral rule-based textual descriptions of Control Interpreted Petri Nets (CIPN) may be formally transformed into a proper format (structured template in

* Institute of Computer Engineering and Electronics
e-mails: {M.Adamski, M.Wegrzyn, A.Wegrzyn}@iie.uz.zgora.pl

XML (Extensible Markup Language)), which can be accepted as a shell for standard hardware description languages, like VHDL or Verilog. The automatic design process is realized by means of related CAD (Computer Aided Design) tools on the Register Transfer Level (RTL), which makes it possible to obtain compact and reliable implementation.

14.1.1. Background

In general, field programmable logic can be re-configured by the user to perform particular combinational or registered logic functions. The design process is greatly simplified by FPGA and CPLD compilers. The effective simulation allows the logic controller to be debugged before the device is programmed. If a design change is needed, it is a simple matter to re-edit the original specification and then re-program or exchange the old device. FPGA can be dynamically reconfigured to perform many different logic control programs, serving as an adaptive concurrent (parallel) state machine with a data path (Nascimento *et al.*, 2004; Węgrzyn M., 2006).

While software implementation of logic controllers can be applied only to comparatively slow targets, hardware implementation of a Petri net is recommended for high-speed, parallel controllers, interacting with several concurrent processes. Other advantages of the method are reusability, fast prototyping and testability, which are ensured because the reconfigurable controller fully implements the structure of a discrete algorithm and its desired properties (Adamski, 1999; Milik and Hrynkiewicz, 2001).

In industrial control projects generally engineers of different disciplines have to work together. This results in the need for integrating heterogeneous descriptions of engineering notations, like Sequential Function Chart (SFC) 1131-3 (Adamski, 1999; Węgrzyn M., 2003), as well as those mainly used by computer scientists, like Petri nets (Murata, 1989).

A control interpreted Petri net (David and Alla, 1992) is an extended safe Petri net supplemented by the use of Boolean conditions associated with transitions. The transition fires only when each input place of transition contains a token, and the Boolean condition of the given transition is satisfied. Some Boolean variables are associated with places. They are satisfied when such places are marked.

The chapter is concentrated on behavioral specification of reconfigurable logic controller programs, written later in HDL (Adamski, 2006a; Fernandes *et al.* 1997; Pardey and Bolton, 1992; Węgrzyn M., 2006; Wolański *et al.*, 1997). Petri net hardware synthesis from VHDL has been also developed in the Linkoping University, Sweden (Eles *et al.*, 1998), and some other universities (Yakovlev *et al.*, 2000).

The book (Mandado *et al.*, 1996) makes the connection between digital electronic design with Programmable Logic Devices (PLDs) and Programmable Logic Controllers (PLCs). The design of Petri net-based controllers is summarized in (Chang *et al.*, 1998; Yakovlev *et al.*, 2000). Several aspects related to hardware design with Petri nets can be found in the books (Adamski *et al.*, 2005; Cortadella *et al.*, 2002; Zakrevskij, 1999). The fundamentals of concurrent (parallel) controllers design are based on previous research reported, among others, in (Adamski 1991; Adamski *et al.*, 2005; Biliński *et al.*, 1994). The methodology for digital design of concurrent (parallel) controllers from Sequential Function Charts has been developed for several years

and is presented in the papers (Adamski and Monteiro, 1996; Adamski, 1999; 2006a; 2006b; Węgrzyn M., 2003).

To model and design a concurrent (parallel) controller as a digital system implemented in hardware with a safe Place/Transition (P/T) net, control interpreted Petri nets are adapted (Adamski, 1993; Biliński, 1996; Fernandes *et al.*, 1997; Kozłowski *et al.*, 1995; Pardey and Bolton, 1992). Logic expressions are assigned to transitions as guards (predicates). Output signals are attached to places, to represent the unconditional controller action (Moore type outputs) as well as conditional control actions (Mealy type outputs). Transition firings are synchronized with the active edge of a global clock. The proper local state encoding (place encoding) guarantees that all enabled transitions can fire independently, in any order, not necessary with the same edge of the clock. An undesirable situation in an interpreted Petri net occurs when two or more transitions attempt to simultaneously unmark the same shared input place. It is considered that the behavior of the net is deterministic, since all such possible conflicts are previously eliminated by consistent labeling of the transitions by the guards. A hierarchical specification mechanism is introduced by means of macroplaces to permit the encapsulation of subnets as macronodes, which decreases the size of the specification, improves its readability and introduces modularity (Adamski *et al.*, 2005). It should be noted that the modular Petri net divides control tasks into nested, concurrent, nearly independent control procedures (Adamski, 2006a).

A marking of the Petri net is regarded as a description of a global state of the modeled system and can be treated as a global state of the logic controller. The transformation of the marking corresponds to a global state change, which is depicted as an edge in the reachability graph and is labeled by a single transition. The transition describes the selected local state change among local states, which are expressed as input places of the transition (current local states) and output places of transitions (next local states).

It should be stressed that the interpreted Petri net is directly mapped into Boolean equations as well as VHDL statements without explicit enumeration of all possible global states and global state changes. The codes of global states are implicitly formed in the global state register as a superposition of the current local state codes, attached to the subsets of places, which are marked simultaneously. The VHDL style and template type, introduced by Bolton, was continued and modified by several researchers (Fernandes *et al.*, 1997; Pardey and Bolton, 1992; Wolański *et al.*, 1997). To keep a very strict correspondence between the initial specification as a Petri net and hardware description languages, such as VHDL, the rule-based textual form is considered (Adamski, 1991). It was developed as a bridge between PN and its VHDL models (Fernandes *et al.*, 1997; Kozłowski *et al.*, 1995; Wolański *et al.*, 1997). Petri net description in Verilog is presented in the papers (Adamski, 2006a; Węgrzyn M., 2003).

The work introduced at University of Zielona Góra was extended at several universities abroad (Adamski, 1986; 1990b). The result of collaboration with the University of Bristol, UK, is reported, for example, in the papers (Pardey *et al.*, 1992; 1994; Biliński *et al.*, 1994; Kozłowski *et al.*, 1995). Some parts of the work have been realized at the University of Minho, Braga, Portugal (Adamski and Monteiro, 1995; Fernandes *et al.*, 1997; Węgrzyn *et al.*, 1996; 1998), the Fern University in Hagen

(Halang and Adamski, 1997), the Technical University of Ilmenau, Germany (Fengler *et al.*, 1996) and the Academy of Science of Byelorussia (Adamski and Zakrevskij, 2001).

The current research work is an extension of the previous one, and it is especially related to Petri net-based structured state assignment for concurrent state machines, different kinds of Petri net decomposition, symbolic exploration of the Petri net state space, etc. (Adamski, 2006b; Andrzejewski, 2002; Łabiak, 2003; Miczulski and Adamski, 2006; Węgrzyn A., 2003; 2006; Węgrzyn M., 2006).

14.2. Logic controller and the binary control system

Logic controllers are related mainly to relatively simple embedded discrete systems, whose behavior is defined by the interaction with their environment. As synthesis tools become more advanced and user friendly, the entry point in the design process is moving towards higher levels of specification. The proposed structured design is used in many formalisms used to specify logic controllers programs, such as the interpreted Petri net or the Sequential Function Chart. VHDL is used for the synthesis, verification, and documentation of design. The logic controller model (concurrent state machine with a data path) may be implemented explicitly in the hardware description language, or from the front-end entry (shell).

Application specific logic controllers are reconfigurable dedicated devices, implemented in programmable logic. Embedded systems require real-time operations and concurrent processing. A discrete HDL model of the logic controller is derived directly from the control interpreted Petri net and it is implemented as an FPGA-based control unit, for example, in array logic, nested inside a modern integrated digital microsystem (Fig. 14.1). An industrial logic control system consists of three parts: a logic controller, an operation unit, and an environment, which involves a human operator. The system's functionality is usually fixed and primarily determined by subsystems interactions, including the influence of the environment.

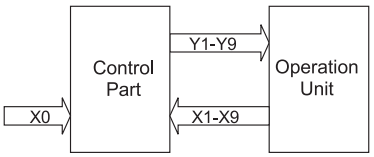


Fig. 14.1. Industrial logic control system

The applicability of the approach is demonstrated by a solution to the tank filling problem (Adamski, 1987). It has been adapted by many authors as an illustration of several different design methodologies. The controlled part is shown in Fig. 14.2.

The reactor R is fed after the start signal *x0* with two kinds of liquid from the measuring vessels *MV1* and *MV2*, which feed from the storage vessels *SV1* and *SV2*. After the reaction between the liquids is completed, the reactor is discharged into the catch vessel *CV*. When the reactor is empty, the process product is transported to the storage vessel *SV3* using the carriage *C*. To ensure a complete reaction, the process liquid in the reactor is agitated by the stirrer *ST*.

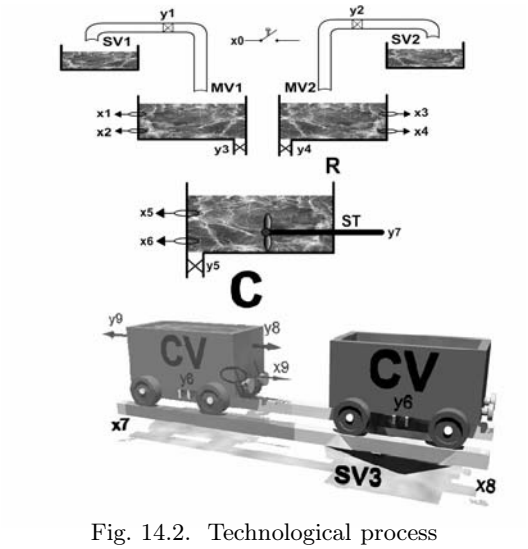


Fig. 14.2. Technological process

When the push-button *x0* is pressed, the valves *y1* and *y2* are opened, and the measuring vessels *MV1* and *MV2* are refilled until the high level conditions *x1* and *x3* are sensed. The start signal *x0* also forces the carriage C with the carriage vessel CV to go towards the initial left position (*y9*). After that valves *y1* and *y2* are closed, the reactor R is fed from the measuring vessels *MV1* and *MV2* through the valves *y3* and *y4*. The reactor stirrer (ST) should mix (*y7*) when the level in the reactor is higher than *x5*. When a low level (*!x2* in *MV1* and *!x4* in *MV2*) is sensed, the reactor charge valves *y3* and *y4* are closed and the reactor is emptied through the discharge valve *y5*.

After discharging the reactor (*!x6*) to the carriage vessel CV, the product is transported right by using a carriage (*y8*). After that (*x8*), the product of the reaction is placed in the container *SV3* through *y6* until *!x9*. When a full technological cycle is completed, the system waits in the initial, idle state.

It is necessary to identify the inputs and outputs (Table 14.1) and the unique local states (denoted by places) of the controller (Table 14.2). The behavior of the controller is represented as a related Petri net (Chapter 14.3).

14.3. Petri net as a specification of a concurrent state machine

14.3.1. Petri nets and logic controllers

Designing the discrete controller as a digital subsystem involves the generation of Petri net-based behavioral specification by analyzing the properties of the controlled objects and desired functionality (Fig. 14.3). The sequence control problems are represented in a structural manner, showing the various actions (y) to be taken in each discrete step (P) and indicating the conditions (x) which need to be satisfied to advance to the next step.

A Petri net represents the behavior of a discrete controller as sequences of places and transitions. Each place is related to an action that is either active or inactive

Table 14.1. Description of inputs and outputs of the controller

Signal name		Description
Inputs	*xo*	Start button
	x1, x3	Max level in vessel *MV1, MV2*
	x2, x4	Min level in vessel *MV1, MV2*
	x5	Min level for stirrer *ST*
	x6	Min level in reactor *R*
	x7, x8	Left, right side position of carriage *C*
	x9	Min level in carriage vessel *CV*
Outputs	*y1, y2*	Feed valves of vessel *MV1, MV2*
	y3, y4	Discharge valves of vessel *MV1, MV2*
	y5	Discharge valves of reactor *R*
	y6	Discharge valves of carriage vessel *CV*
	y7	Stirrer (agitator) *ST*
	y8, y9	Right, left direction of carriage *C*

Table 14.2. Local states description

Local state	Description	Local state	Description
P1	Initial state (*Start*)	*P9*	Filling tank R from tank *SV1*
P2	Filling tank *SV1*	*P10*	Filling tank R from tank *SV2*
P3	Filling tank *SV2*	*P11*	Waiting for empty tank *SV1*
P4	Waiting for full tank *SV1*	*P12*	Waiting for empty tank *SV2*
P5	Waiting for full tank *SV2*	*P13*	Waiting for filling *CV*
P6	Carriage C is going left	*P14*	Tank R is emptied to *CV*
P7	Stirrer *ST* is turned on	*P15*	Carriage *C* is going right
P8	Idle state of stirrer *ST*	*P16*	Tank *CV* is emptied

($y = 1$ or $y = 0$). If the transition label (guard, predicate) is true, the active (marked) input places of the transition can become inactive and the next output places can became active. The required sequence is shown by directed edges, pointing the flow of control. The interpreted Petri net encapsulates concurrent input and output sequences that the controller can accept and produce.

Petri nets have shown to be a powerful tool to specify and model the behavior of parallel systems and, particularly, parallel (concurrent) digital controllers (Fig. 14.1). A detailed analysis of the model is possible, based on a set of formal validation procedures, and permitting the detection of a large number of design errors prior to system implementation.

Safe PNs can be viewed as a natural extension to the well-known Finite State Machine (FSM) specifications. Decomposing a discrete system into a controlled path

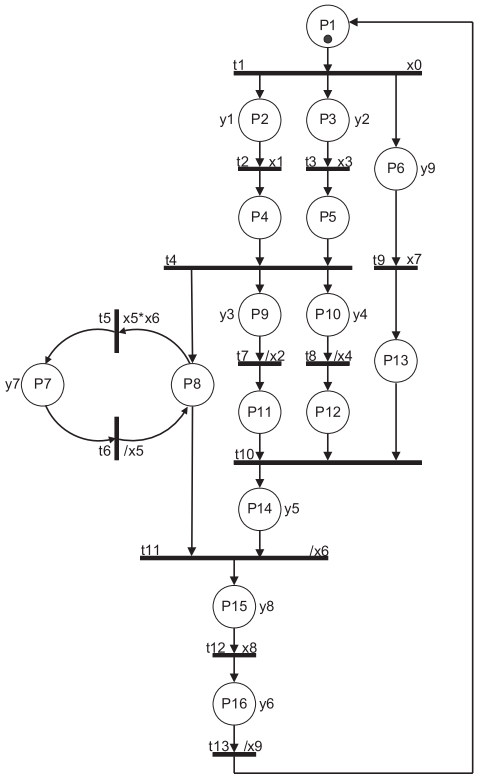


Fig. 14.3. Petri net model

and a controller is a widely accepted step of the classical design methodology. To realistically model any parallel controller with Place/Transition PNs, some modifications have been made. The first well-known modification is the association of logical expressions, which are called guards, with transitions. Additionally, to represent the controller actions, the output signals have been associated with the places or transitions. Each place of the Interpreted Petri Net (IPN) is viewed as a control state. The global state of the controller is given by the PN marking (distribution of tokens by the places). To achieve and maintain the advantages of the linked state machine models and Grafcet-like models (David and Alla, 1992), test arcs (enabling and inhibitor arcs) would be considered.

A synchronous (clocked) transition fires with the active edge of the system clock. The enabled transitions at a given moment do not fire instantaneously but wait for a clock pulse. Then the enabled transitions fire and a new marking is obtained. The resulting PN type is a Synchronous Interpreted PN (SIPN) (Kozłowski *et al.*, 1995).

14.3.2. Concurrent state machine

A Concurrent State Machine (CSM) has a set of inputs or outputs, which allow the data to be exchanged with the external environment. Virtual Sequential State

Machines (SSMs), which are included in the CSM, interact with others (Fig. 14.4) by means of a shared memory (internal state register).

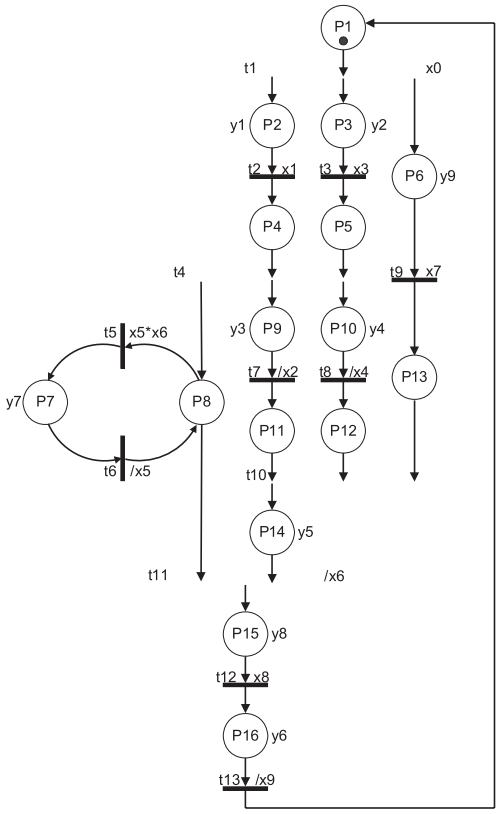


Fig. 14.4. SM components in a Petri net model

The Petri net is directly mapped into Boolean equations (decision rules) without explicit enumeration of all possible global states and all possible global state changes. The specification is given in terms of local state changes (related to Petri net transitions). Input signals are associated with transitions as labels, called also guards. Moore type output signals are generated by places. Mealy type outputs are activated if the particular place is marked and the relevant label (logic expression formed from input signal names) is true. Sometimes it is useful to attach a Mealy output directly to the transition. In such a way the Mealy output is active if the given transition is enabled.

A typical state machine application involves the control of a fast device under the direction of a higher level controller. New trends go towards the implementation of a concurrent, dedicated digital microsystem as a single FPGA chip.

The concurrent logic controller can be presented using a concurrent view of the modeled system behavior. The logic controller model should retain the natural partitioning of the behavior imposed by the designer. The functionality is very often

represented as a set of concurrent blocks of a manageable size that communicate using few signals.

The initial partitioning generates interacting finite state machines with separate state registers. They form a conservative interpreted colored State Machine Petri Net (SM-PN) (Adamski and Węgrzyn, 1994). A Petri net can be expressed graphically, then decomposed (Augin *et al.*, 1978) into Linked State Machines (LSMs) (Belhadj *et al.*, 1993), and finally translated into an equivalent set of Verilog models (Węgrzyn M., 2006).

The collaborated state machines can be implicitly given by means of CIPNs or industrial formats (IEC-SFC). The generic architecture of interactive FSMs is a set of interconnected FSMs which exchange data (local internal state signals or output signals) through input and output ports. Each component is characterized by input and output ports that connect it with other components and external controller ports. The set of communicating FSMs is called a concurrent state machine (concurrent controller) iff two or more FSMs are not exclusively active.

A colored Petri net is also represented as a graph with two kinds of nodes – places and transitions. The marking of the net is represented by several colored tokens. Directed implicitly colored arcs connect explicitly colored places and implicitly colored transitions. Transitions are allowed or prevented from occurring with respect to a particular color if the attached colored Boolean expression is respectively true or false.

By using a Colored Petri Net (CPN), each color can represent a sequential process, i.e. a particular color can be related only to one process. The total number of the colored tokens indicates the number of concurrent processes being active at any particular global state. Such a colored Petri net can be decomposed on several SM-PNs. Figure 14.5 shows a colored Petri net model of the net from Fig. 14.3.

14.3.3. Textual specification of Petri nets

The digital system is considered as an abstract reasoning system (rule-based system) implemented in hardware. The mapping between the inputs and outputs of the system is described in a formal manner by means of logic rules (represented as sequents) with some temporal operators, especially the operator 'next'@. Sequents may be roughly treated as more general forms of clauses with conjunctive antecedents and disjunctive consequents and they represent assertions (Adamski, 1990a).

In the paper (Kozłowski *et al.*, 1995), the Petri Net Specification Format (PNSF) for VLSI design was introduced. It has been later extended into the newer version – PNSF2 (Fig. 14.6) (Węgrzyn, 1998). The format PNSF2 supports structured, hierarchical designs with Petri nets and FPGAs. The CONPAR specification format (Fernandes *et al.*, 1997) is consistent with previously introduced rule-based specification languages and was also created mainly as a bridge between the textual logic description of Petri nets and their VHDL models. Transition rules in PARIS and CONPAR are treated as production rules ('if-then' non-procedural statements). The rule-based description (Adamski, 1990b), supported by means of logic deduction techniques (Gentzen natural logic calculus), has been recently refreshed in a programmable logic controller design context.

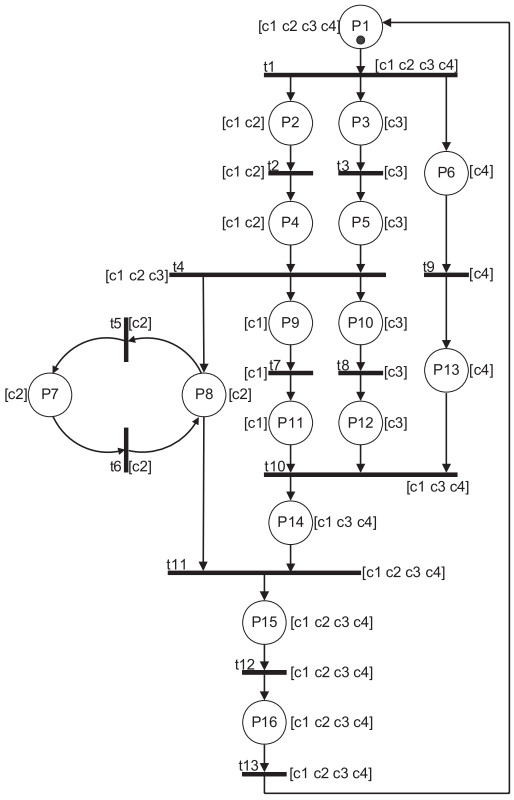


Fig. 14.5. Colored flat Petri net

Petri nets can be also specified in the newer textual formats – PNSF3 (Petri Net Specification Format v.3) (Węgrzyn A., 2003) and CCPNML (Concurrent Control PNML) (Węgrzyn A., 2006). PNSF3 represents interpreted, synchronous, hierarchical and colored Petri nets. PNSF3 is specified in the XML language. PNSF3 describes the elements of the structure of the Petri net, i.e. places, transitions, and connections between them. The format keeps also information about clocks, inputs and outputs of controllers. Notwithstanding, it does not store information about the position of each element on a net picture. The CCPNML format is a version of the PNML format (Billington *et al.*, 2003) adapted for concurrent controller specification. It introduced addition tags that represent some elements suitable for controller specification. In contrast to PNSF3, it contains information about the positions of elements in a graphical representation of nets.

14.3.4. Hierarchical interpreted Petri nets

A straightforward approach to the design of the hierarchical digital controller is to model the system by using a Petri net and then the well-known bottom-up reduction techniques for its decomposition and validation. This approach is limited by the initial

```
.clock CLK
.inputs x0 x1 x2 x3 x4 x5 x6 x7 x8 x9
.comb_outputs y1 y2 y3 y4 y5 y6 y7 y8 y9
.places P1 P2 P3 P4 P5 P6 P7 P8 P9 P10 P11 P12 P13 P14 P15 P16
.transitions t1 t2 t3 t4 t5 t6 t7 t8 t9 t10 t11 t12 t13
.net
  t1: P1 * x0 |- P2 * P4 * P6;
  t2: P2 * x1 |- P4;
  t3: P3 * x3 |- P5;
  t4: P4 * P5 |- P8 * P9 * P10;
  t5: P8 * x5 * x6 |- P7;
  t6: P7 * !x5 |- P8;
  t7: P9 * !x2 |- P11;
  t8: P10 * !x4 |- P12;
  t9: P6 * x7 |- P13;
 t10: P11 * P12 * P13 |- P14;
 t11: P8 * P14 * !x6 |- P15;
 t12: P15 * x8 |- P16;
 t13: P16 * !x9 |- P1;
.MooreOutputs
  P2 |- y1;
  P3 |- y2;
  P6 |- y9;
  P7 |- y7;
  P9 |- y3;
 P10 |- y4;
 P14 |- y5;
 P15 |- y8;
 P16 |- y6;
.marking P1
.e
```

Fig. 14.6. Petri net specification in PNSF2

complexity of the net, mainly from the human interaction point of view. Fortunately, most of the results taken from reduction methodology can be adopted for the synthesis of a large Petri net from the initial specification (Węgrzyn and Adamski, 1999).

A given Petri net is transformed into a hierarchical macronet, i.e. a net having structured macroplaces, which represent Petri net subnets, particularly state machine subnets (Figs. 14.3 and 14.4). The first level macronet (Fig. 14.7) consists of the macroplaces *M1–M6*, (Table 14.3), and especially the exposed places *P7*, *P8* for a detailed analyses purpose (Chapter 14.4), and the isolated place *P14*.

Table 14.3. Description of the first-level macronet

Macroplace	Internal Places
M1	*P1, P15, P16*
M2	*P2, P4*
M3	*P3, P5*
M4	*P9, P11*
M5	*P10, P12*
M6	*P6, P13*

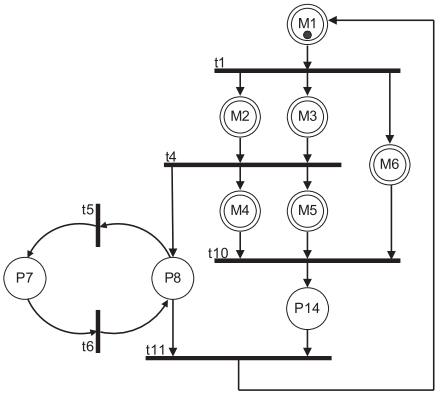


Fig. 14.7. First level of a macronet

The second-level macronet (Fig. 14.8) is built from the macroplaces *M1*, *M6–M8* and the places *P7*, *P8*, *P14*, where *M7 = {M2, M3}* and *M8 = {M4, M5}*.

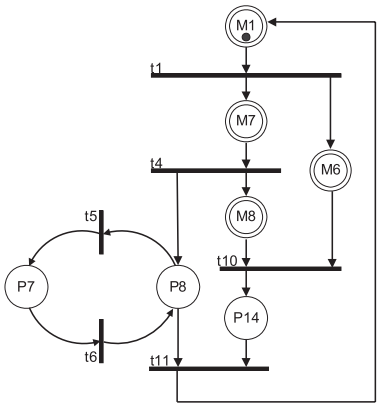


Fig. 14.8. Second level of a macronet

The third-level macronet (Fig. 14.9) is built from a macroplaces *M1*, *M6–M9* and the place *P14*, where *M9 = {P7, P8}*.

At the next step of the analysis, the reachability graph of the macronet is obtained (Fig. 14.10, Table 14.4).

14.3.5. Relation of concurrency

Every two simultaneously marked macroplaces *Mi, Mj* are represented by vertices (*Mi, Mj*) connected by edges in the *concurrency graph* (*GC*). The complement of the concurrency graph *GC* forms a *non-concurrency graph* (*GN*). In the non-concurrency graph, edges connect pairs of the macroplaces, which are not simultaneously marked. The two graphs are frequently represented as adjacency matrices. The adjacency matrix of the graph *GC*, which is supplemented with the numbers 1 in the main

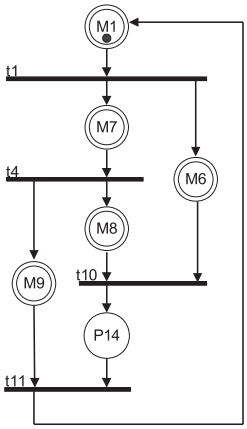


Fig. 14.9. Third level of a macronet

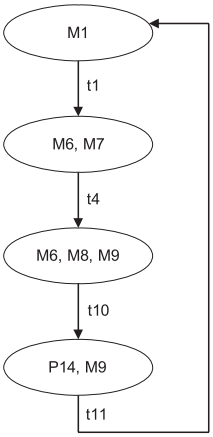


Fig. 14.10. Reachability graph

Table 14.4. Reachability graph tabular description for the third level of hierarchy

Marked macroplaces	Transition	New marked macroplaces
M1	*t1*	*M6, M7*
M6, M7	*t4*	*M6, M8, M9*
M6, M8, M9	*t10*	*P14, M9*
P14, M9	*t11*	*M1*

diagonal, is called a *concurrency matrix* (Amroun and Bolton, 1989; Pardey *et al.*, 1992).

A relatively efficient method for *GC* or *GN* graph representation is *successor listing*. After assigning the vertices, in any order, the numbers $1, 2, \ldots, n$, each vertex k is represented by a linear array, whose first element is k and whose remaining

Table 14.5. Concurrency table

(Macro)place	List of its concurrent (macro)places
M1	—
M6	*M7, M8, M9*
M7	*M6*
M8	*M6, M9*
M9	*M6, M8*
P14	*M9*

Table 14.6. Non-concurrency table

(Macro)place	List of its non-concurrent (macro)places
M1	*M6, M7, M8, M9, P14*
M6	*M1, P14*
M7	*M1, M8, M9, P14*
M8	*M1, M7, P14*
M9	*M1, M7*
P14	*M1, M6, M7, M8*

elements are vertices that are adjacent to k. The tabular version of successor lists is called the *concurrency table* for the graph *GC* (Table 14.5) and the non-concurrency table (Table 14.6) for the graph *GN* (Adamski, 1986; Węgrzyn, 1998).

The degree of the vertex k is the number of edges incident to it. Any vertex adjacent to the vertex k is said to be dominated by k, whilst any other vertex is independent of k.

The macroplaces in the list of successors for the graph *GN* are reordered according to the degree of incidence starting from the vertex with a maximal degree. The first vertex gets a code and is removed from the list. The encoding process is continued until the list is empty.

It has been previously mentioned that places in a Petri net are marked sequentially or concurrently with respect to each other. If the local state space of a Petri net or Petri macronet is explicitly given (Fig. 14.10), it is straightforward to construct the concurrency graph (Fig. 14.11(a)). This is performed by means of the inspection of cliques related to vertices in the reachability graph. Figure 14.11(b) shows the extended version of the concurrency graph, where the macroplaces *M7* and *M8* are substituted by their internal concurrent macroplaces.

An entry *Cij* in the concurrency matrix C equals 1 if places corresponding to the row i and the column j may hold tokens simultaneously (they belong to the same marking of the net), otherwise it equals 0. It should be noted that the main diagonal of the matrix contains the numbers 1 (Table 14.7). The concurrency matrix may be used for several analysis or synthesis techniques, including hierarchical, sequential, parallel

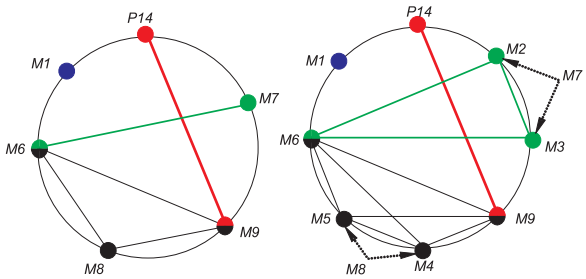


Fig. 14.11. Concurrency graphs

decomposition and place encoding. The complementary matrix, which represents the relation of non-concurrency, is referred to as a *non-concurrency matrix* (Table 14.8).

Table 14.7. Concurrency matrix

	M1	*M6*	*M7*	*M8*	*M9*	*P14*
M1	1	0	0	0	0	0
M6	0	1	1	1	1	0
M7	0	1	1	0	0	0
M8	0	1	0	1	1	0
M9	0	1	0	1	1	1
P14	0	0	0	0	1	1

Table 14.8. Non-concurrency matrix

	M1	*M6*	*M7*	*M8*	*M9*	*P14*
M1	0	1	1	1	1	1
M6	1	0	0	0	0	1
M7	1	0	0	1	1	1
M8	1	0	1	0	0	1
M9	1	0	1	0	0	0
P14	1	1	1	1	0	0

The concurrency and non-concurrency matrices can be generated in many simpler ways, especially for some restricted classes of Petri nets.

14.4. Verification and decomposition methods

Formal analysis of Petri nets, which represent the behavior of logic controllers, is important from a practical point of view, so many advanced methods of Petri net analysis have been developed. The properties of the discrete model of the controller

are verified on the basis of the well-known Petri net theory. Almost all methods show merely if the net is live, bounded, persistent etc., without presenting exactly the reason for the potential design error in the desired, distributed state space of an equivalent concurrent state machine.

After finding all deadlocks and traps in a Petri net and checking the dependencies between the set of deadlocks and the set of traps, it is possible to solve the problem if the Petri net is bounded, live and persistent. Apart from that, using the new proposed symbolic method it is possible to decompose the verified Petri net into concurrent subnets, nearly without any additional effort. In such a way the concurrent state machine can be transformed into a collection of collaborating sequential state machines. The calculation of deadlocks and traps is one of the most important analysis tasks, because a well-designed system should not contain dead events, which can never occur.

In the presented approach, Thelen method (1981), based on the mathematical logic, is applied for checking some structural conditions for the liveness of the Petri net. This method permits efficient calculation of prime implicants of a Boolean function, which is represented in a clausal form. In such a way it allows obtaining separated sets of deadlocks and traps, represented compactly in the form of ternary vectors $(0, 1, -)$. The method is based on generating a decision tree for searching Conjunctive Normal Form (CNF.) Particular deadlocks and traps of the Petri net correspond to logical equations in a conjunctive normal form.

A Horn formula is a conjunction of the basic Horn formulae. The basic Horn formula (Horn clause) is a disjunction of literals, with at most one positive literal. A literal is either a propositional letter P (positive literal) or the negation $/P$ of a propositional letter P (negative literal).

In Fig. 14.12, complete Horn formulae, representing deadlocks and traps in the discussed macronet from Fig. 14.8, are presented. For such formulae, two Thelen trees are generated, and sets of deadlocks and traps are received (Fig. 14.13).

```
HF = (/M1 + M7) * (/M1 + M6) * (/M7 + M8) * (/M7 + P8)
       * (/P8 + P7) * (/P7 + P8) * (/M6 + /M8 + P14)
       * (/P8 + /P14 + M1)

HF' = (M1 + /M7 + /M6) * (M7 + /M8 + /P8) * (/P7 + P8)
        * (/P8 + P7) * (M8 + /P14) * (M6 + /P14)
        * (P8 + /M1) * (P14 + /M1)
```

Fig. 14.12. Horn formulae representing deadlocks and traps

For liveness checking of the Petri net, Commoner's property can be used (Cortadella *et al.*, 2002; Murata, 1989; Pastor *et al.*, 2001). It is tested if each previously obtained deadlock contains a marked trap. It is also considered that each analyzed deadlock contains a minimal deadlock or it is a minimal deadlock. If each minimal deadlock contains a marked trap, then each deadlock contains a marked trap. Therefore, for liveness checking of the Petri net, only minimal deadlocks should be considered. Similarly, during the analysis of the relations between the set of deadlock and the set of traps, only minimal traps are considered. From the above deliberations it

```
RESULT_HF = (P7 * P8 * /P14 * /M1 * /M6 * M8)
             + (P7 * P8 * /P14 * /M1 * /M7 * /M8)
             + (P7 * P8 * /P14 * /M1 * /M6 * /M7)
             + (/P7 * /P8 * P14 * /M1 * /M7)
             + (/P7 * /P8 * /M1 * /M7 * /M8)
             + (/P7 * /P8 * /M1 * /M6 * /M7)
             + (P7 * P8 * P14 * M1 * M6 * M7 * M8)

RESULT_HF' = (/P7 * /P8 * /P14 * /M1 * /M6)
              + (P7 * P8 * /P14 * /M1 * /M6 * /M8)
              + (P7 * P8 * /P14 * /M1 * /M6 * M7)
              + (P7 * P8 * P14 * M1 * M6 * M7 * M8)
              + (/P7 * /P8 * /M1 * M6 * /M7 * M8)
              + (/P7 * /P8 * /P14 * /M1 * /M7)
              + (P7 * P8 * /P14 * /M1 * /M7 * /M8)
```

Fig. 14.13. Expression representing sets of deadlocks and traps

appears that during the checking of liveness for Free Choice (FC) or Extended Free Choice (EFC) Petri nets, it should be necessary to see only whether minimal, marked (in initial marking) traps are contained inside minimal deadlocks.

Liveness checking can be performed as follows:

i) If there is minimal deadlock not marked in the initial marking, then the Petri net is not live.

ii) If each deadlock is marked, then it is checked if each minimal deadlock contains a minimal marked trap. If each deadlock contains traps, then the net is live. If at least one deadlock does not contain any trap, then the net is not live.

In the example considered, the Petri net is live, because each minimal deadlock contains a minimal initially marked trap. Using previously obtained sets of deadlocks and traps it is possible to check if an analyzed Petri net is bounded or not. For checking such a property, the dependency between sets of deadlocks and traps are considered. These sets is compared, and vectors which represent both deadlocks and traps at the same time are received.

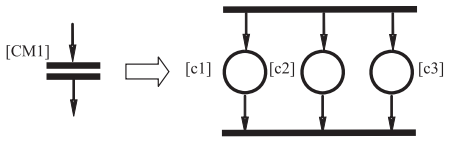
There are obtained three minimal P-invariants: {*P14,M1,M6*}, {*P7, P8, M1, M7*}, {*P14, M1, M7, M8*}, which correspond to the vectors $[1,1,0,0,0,0,1]$, $[1,0,1,0,1,1,0]$, $[1,0,1,1,0,0,1]$. Each element of vector represents a place in Petri net, i.e. the first element corresponds to the macroplace *M1*, the second element – to the macroplace *M6*, and so on. If the whole Petri net is cover by P-invariants, then the Petri net is bounded. Otherwise, it could not be decided if the Petri net is bounded or not. In the presented example of a Petri net, it is completely covered by P-invariants, and this means that the net is bounded. Such a P-invariant could correspond to a one-sequence automaton (state machine) and it is distinguished by a selected individual color in the complete Petri net. The total number of colors depends on the number of P-invariants for a flat Petri net, so some colors in the should be split off.

For example, the complex color [*CM1*] of the multi-active macroplace in Fig. 14.14 is replaced by three individual colors [*c1 c2 c3*] during the expansion of this place into three parallel separated places. During parallel decomposition, a Petri net is divided

Table 14.9. Set of deadlocks and traps

Deadlocks	Traps
{*P14, M1, M6*}	{*P7, P8, P14, M1, M6*}
{*P14, M1, M6, M7*}	{*P7, P8, P14, M1, M6, M7*}
{*P14, M1, M7, M8*}	{*P7, P8, P14, M1, M6, M7, M8*}
{*P14, M1, M6, M7, M8*}	{*P7, P8, P14, M1, M6, M8*}
{*P7, P8, M1, M7, M8*}	{*P14, M1, M6*}
{*P7, P8, P14, M1, M7, M8*}	{*P14, M1, M6, M8*}
{*P7, P8, P14, M1, M6, M7, M8*}	{*P14, M1, M6, M7, M8*}
{*P7, P8, M1, M6, M7, M8*}	{*P7, P8, M1, M7*}
{*P7, P8, M1, M7*}	{*P7, P8, P14, M1, M7*}
{*P7, P8, M1, M6, M7*}	{*P7, P8, P14, M1, M7, M8*}
{*P7, P8, M1, M7, M8*}	{*P14, M1, M7, M8*}
{*P7, P8, M1, M6, M7, M8*}	
{*P7, P8, P14, M1, M6, M7*}	

into a set of subnets. These subnets have to satisfy some restrictions, e.g. a subnet must include only places which are sequential to each other and it cannot contain multi-input or multi-output transitions (Augin *et al.*, 1978). The decomposition of a Petri net can be based on the coloring of the net. If it is possible to color a Petri net according to the rules given in the papers (Adamski and Węgrzyn, 1994), then the net could be decomposed into one-sequence Petri net subnets. In such a case, P-invariants correspond to concurrent automata subnets.



[CM1] → [c1] [c2] [c3]

Fig. 14.14. Expansion of a macroplace and the splitting of a complex color

The three P-invariants of the macronet in Fig. 14.15 represent three colors covering whole hierarchical Petri net. It means that a Petri net is bounded. Each color presents one automaton. The first vector corresponds to the color *CM1*, second vector corresponds to the color *CM2*. The places *M1, M7, P7, P8* are recognizable by the color *CM1*, the places *M1, M7, M8, P14* are distinguished by the color *CM2* and places *P8, P9, P10, P11* are colored by color *CM3*. The places *M1, M6, P14* are notable by the color *CM3*.

After the expansion of the hierarchical Petri net, the complex color *CM1* is replaced by the color *c2*, the color *CM2* is substituted by the color *c1* and *c3*, and the color *CM3* is changed into the color *c4*. Each color corresponds to a particular

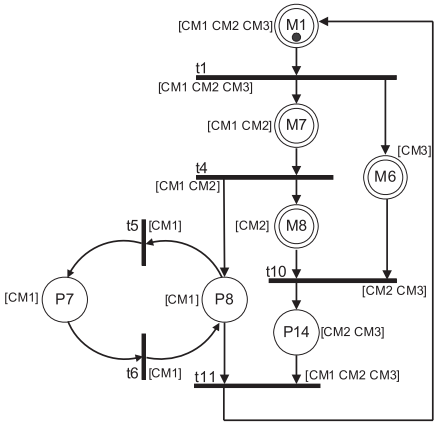


Fig. 14.15. Colored macronet

partial control automaton. In the discussed example, there are four automata (SM-components), named A, B, C, D, which are shown respectively in Fig. 14.16. In the presented case, the color *c1* represents the automaton A, the color *c2* represents the automaton B, the color *c3* represents the automaton C and the color *c4* represents the automaton D. Because some places are shared among the four automata and it is necessary to avoid duplications of them, the additional initial idle places S2, S3, S4 have to be introduced.

14.5. Controller synthesis

14.5.1. Concurrent local state assignment

The direct mapping of a Petri net into field programmable logic is based on the correspondence between a transition and a simple combinational circuit and the correspondence between a place and a clearly defined subset of the state register (Adamski, 1986; 1987; 1991; 1993). In dealing with concurrency, the designer is confronted with some problems that will not arise in logic synthesis of sequential systems. One of them is concurrent local state encoding. One-hot encoding of a Petri net is treated as the simplest case of a more general mapping. The one-hot method (Fernandes *et al.*, 1997; Pardey and Bolton, 1992; Patel, 1990; Węgrzyn, 1998) produces fast designs with a simple combinational part, especially for implementations in FPGA. It is not assumed that all flip flops, except one, are set to 0 since several places can be marked simultaneously.

It is possible to drastically reduce the global number of flip-flops, but together with increasing the complexity of the combinatorial circuits per particular flip-flop. Adding additional state variables, for the encoding of a particular place, multiplies the number of expressions in flip-flop excitation functions to be realized. This is a strongly recommended arrangement for CPLD with a limited number of output and buried cells, but not necessarily a worthwhile strategy for FPGA-based designs. The

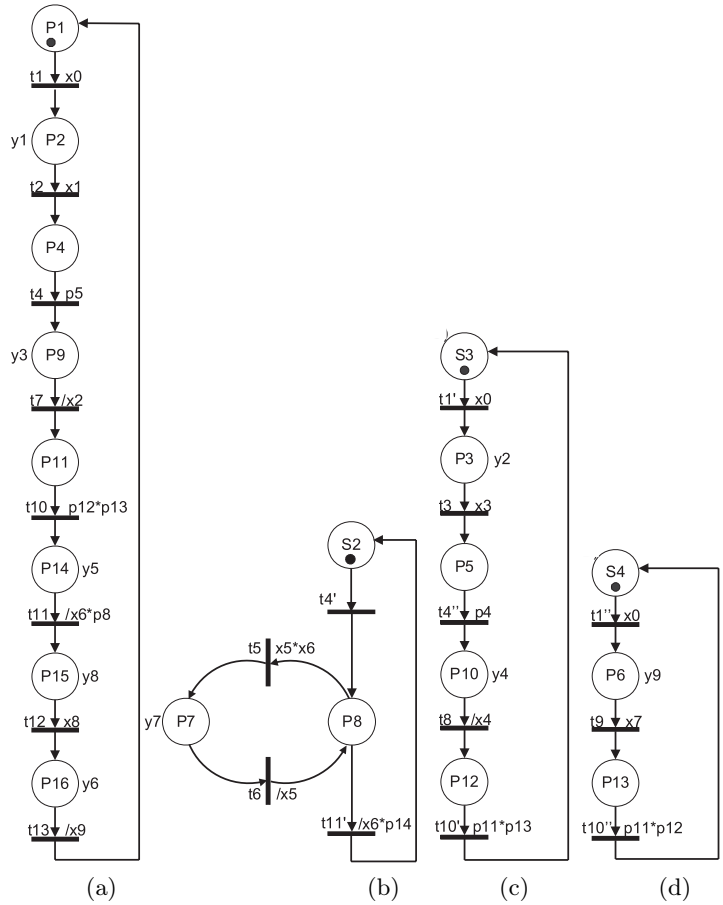


Fig. 14.16. Petri net decomposed into concurrent linked nets

basic methods (Adamski, 1987; 1991; 1993) were improved and developed by Amroun and Bolton (1989), Pardey and Bolton (1992), Kozłowski *et al.* (1995) and Biliński (1996).

The concurrent one-hot encoding is a modification of the popular one-hot-one state assignment of the sequential (non-concurrent) state machine, in which one flip-flop is used for each global state. After such local state encoding of the concurrent state machine, all flip-flops related with simultaneously marked places are set to one at the same time. The total number of flip-flops is equal to the number of places (Patel, 1990):

$$\begin{aligned} code(P1) &= Q1, \\ code(P2) &= Q2, \\ &\vdots \\ code(P16) &= Q16. \end{aligned}$$

The hybrid concurrent one-hot is a mixture of the well-known binary encoding and concurrent one-hot encoding. The advantage is that distributed macroplace encoding remains one-hot like, with fewer flip-flops needed for their internal place encoding. To minimize the number of the output pins of PLD or cells and outputs in FPGA, it is possible to intensively use for that purpose some of the output signals by making them registered. In such a way, the macronet structure is directly mapped into the implementation by means of heuristic, very economical encoding. An example of such kind encoding is as follows (Adamski and Węgrzyn, 1999):

$$
\begin{aligned}
code(P1) &= /Q1 * /Y6 * /Y8,\\
code(P2) &= Q1 * /Q2 * /Q3 * Y1,\\
code(P3) &= Q1 * /Q2 * /Q3 * Y2,\\
code(P4) &= Q1 * /Q2 * /Q3 * /Y1,\\
code(P5) &= Q1 * /Q2 * /Q3 * /Y2,\\
code(P6) &= Q1 * /Q3 * Y9,\\
code(P7) &= Q1 * Q2 * Y7,\\
code(P8) &= Q1 * Q2 * /Y7,\\
code(P9) &= Q1 * Q2 * /Q3 * Y3,\\
code(P10) &= Q1 * Q2 * /Q3 * Y4,\\
code(P11) &= Q1 * Q2 * /Q3 * /Y3,\\
code(P12) &= Q1 * Q2 * /Q3 * /Y4,\\
code(P13) &= Q1 * /Q3 * /Y9,\\
code(P14) &= Q1 * Q3 * Y5,\\
code(P15) &= /Q1 * /Y6 * Y8,\\
code(P16) &= /Q1 * Y6 * /Y8.
\end{aligned}
$$

Some advanced techniques of concurrent state encoding developed by Adamski, Cheremisinova, Pottosin and Zakrevskij are presented in the book (Adamski *et al.*, 2005).

14.5.2. Mapping of the concurrent state machine into programmable logic

In direct implementation, the control algorithm represented by a Petri net is fixed usually at the design stage and it is mapped into a network of interconnected macrocells. A recent overview of a Petri net based direct implementation of logic controllers is given in (Yakovlev *et al.*, 2000).

The direct implementation of concurrent controllers in FPGA is similar to the realizations of logic controllers based on FSMs. The main essential difference is concurrent state assignment (place encoding of a Petri net). The logic controller contains a concurrent local states register, serving also as a global state register. The combination of the code words of individual local states produces unique configuration encoding. The superposition of codes of any two concurrent local states can share logic variables,

but must be represented by words (ternary Boolean vectors), with non-overlapping, complete independent parts. Local states, which can never be concurrent, may also share a part of logic variables, but they must have a common overlapping part, with different values of logic variables.

When a Petri net is used to model a concurrent state machine, places represent its local states. A maximal subset of simultaneously marked places determines the global state of the controller. Transitions describe local state changes, mostly forced by external inputs. For simplicity, it will considered that the CSM is implemented as a sequential circuit with a common internal clock.

Usually the controller, whose output depends on both internal state and external inputs, is modeled as a Mealy state machine. In practice, most of the controllers must be formally classified as Mealy CSMs, because they have at least one Mealy type output that depends on the input as well as the internal state. Very frequently most of the outputs of the same concurrent state machine depend directly only on local internal states. This traditional approach creates several conceptual problems when the state machine is implemented with programmable devices and reconfigurable output cells. As an initial model, the universal concurrent state machine with distinguished Mealy type outputs and Moore type outputs will be considered. The Moore type output by definition is implemented by a combinational cell as a function of state variables. On the other hand, the Moore type output may be produced in advance in a registered output cell, because it must be stable for the entire clock period. In general, Mealy outputs are produced by combinational output cells, but they also may be delayed and synchronized with a clock. Registered outputs can be eventually used for local state encoding.

Proper local state encoding (place encoding) guarantees that all enabled transitions can fire independently, in any order, not necessarily with the same edge of the clock.

14.5.3. HDL modeling and synthesis of SM-components

The main simulation and synthesis tools for concurrent state machines are VHDL (Adamski *et al.*, 1997; Wolański *et al.*, 1997) or Verilog (Węgrzyn M., 2003).

As an example of design methodology, FPGA implementation of logic controller based on parallel decomposition of a Petri net is presented, and the modeling of automata in Verilog is focused on. Figure 14.17 shows a part of the Verilog model realized by using two processes (two "always" statements) (Węgrzyn M., 2003). There is described only the first SM-component (i.e. the biggest automaton A from Fig. 14.16). The standard one-hot method has been applied, which is recommended for using FPGA devices as a final implementation technology. A sequential process is disconnected from the combinational process. Such a description in Verilog is easily readable, because outputs changes are separated from sequences of the states. The assignment of a new current state according to the next state expression is done in the first process. The first process has a clock signal (CLK) that synchronizes the designed system, and a reset signal ($RESET$) to set up the automaton into the initial state on its sensitivity list.

The second process has on its sensitivity list only the current state (place). The new automaton outputs are assigned after the state change.

```
parameter [8:1] P1 =  8'b00000001,
                P2 =  8'b00000010,
                P4 =  8'b00000100,
                P9 =  8'b00001000,
                P11 = 8'b00010000,
                P14 = 8'b00100000,
                P15 = 8'b01000000,
                P16 = 8'b10000000;

always @(posedge CLK, posedge RESET)
begin
   if(RESET==1) PLACE_SM1 <= P1;
   else if(CLK==1)
      case (PLACE_SM1)
         P1:  if (x[0]==1) PLACE_SM1 <= P2;
         P2:  if (x[1]==1) PLACE_SM1 <= P4;
         P4:  if (PLACE_SM3==P5) PLACE_SM1 <= P9;
         P9:  if (x[2]==0) PLACE_SM1 <= P11;
         P11: if (PLACE_SM3==P12 && PLACE_SM4==P13)
                        PLACE_SM1 <= P14;
         P14: if (x[6]==0 && PLACE_SM2==P8)
                        PLACE_SM1 <= P15;
         P15: if (x[8]==1) PLACE_SM1 <= P16;
         P16: if (x[9]==0) PLACE_SM1 <= P1;
         default PLACE_SM1 <= P1;
      endcase
end

always @(PLACE_SM1)
   case (PLACE_SM1)
         P2:     Y_SM1 <= 5'b00001;
         P9:     Y_SM1 <= 5'b00010;
         P14:    Y_SM1 <= 5'b00100;
         P15:    Y_SM1 <= 5'b10000;
```

Fig. 14.17. Verilog model of the first SM-component

Figure 14.18 shows an outline of the proposed design methodology. A Petri net model is converted into an intermediate textual rule-based description PNSF2, which is transformed into a selected Hardware Description Language (HDL), or directly into a netlist (Węgrzyn, 1998). The chosen Verilog provides a path to commercial simulation and synthesis tools. The described subsystem is implemented in Java (Węgrzyn A., 2006; Węgrzyn M. 2006; Węgrzyn *et al.*, 1997).

The PenCAD subsystem contains, among others:

- visualization application;
- Petri net editor, with a translator into HDLs (especially Verilog);
- verification application, supported with an additional module for Petri nets decomposition into FSMs.

Data for transformation, verification, decomposition and visualization are recorded in a data base.

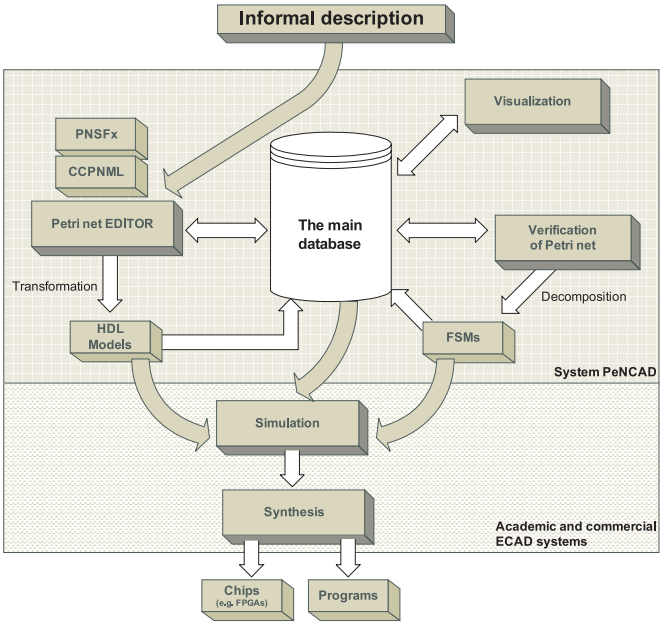


Fig. 14.18. Outline of design methodology

14.6. Conclusions

The chapter focused on behavioral specification of RLC programs, given initially as Petri nets and rewritten later in hardware description languages. Some engineering notations, like the Sequential Function Chart 1131-3, as well as those mainly used by computer scientists, like Petri nets, are integrated inside a unified design methodology. By formally verifying the structural properties of a Petri net, the behavioral properties of control program such as reversibility, liveness and safeness are tested.

Reconfigurable hardware implementation of a Petri net is recommended for high-speed, parallel safety-critical controllers, interacting with several concurrent processes. It should be noted that the modular Petri net divides control tasks into nested, concurrent, nearly independent control procedures. The reconfigurable controller fully and directly implements the structure of a discrete algorithm and its desired properties. Other advantages of the method, such as reusability, fast prototyping and testability, can be assured.

The hierarchical specification mechanism permits the encapsulation of subnets as macroplaces, which decreases the size of the specification, improves its readability and introduces modularity. It should be noted that the interpreted Petri net is directly mapped into Boolean equations as well as HDL statements without explicit enumeration of all possible global states and global state changes. The codes of global states are implicitly formed in the global state register as a superposition of current local state codes, attached to the subsets of places, which are simultaneously marked.

Informal specification of the logic controller consists of a graphical description of the controlled part, and a list specifying the signals from sensors and actuators. It

also includes a short verbal description of the industrial discrete process. Application-specific logic controllers provide precisely the function needed for a specific task because they can be precisely tuned to the applications, resulting in array-based solutions that are faster, compact, and power efficient. Hardware description languages, such as VHDL or Verilog, are used for intermediate representation of controller behavior on the top of existing commercial synthesis tools.

The rule-based textual language input makes it possible to integrate the design system with the existing formal logic based computer-based theorem proovers. Petri net description in HDL provides the opportunity to integrate the existing Petri net software with several commercial systems. A more advanced research, among other topics, would concentrate on:

- unified design of concurrent logic controllers with a data path;
- effective structured state assignment and decomposition techniques devoted to the mapping of Petri net subnets into FPL, embedded in modern microsystems;
- extensive application of symbolic methods for the analysis and synthesis of concurrent state machines, implemented in dynamically reconfigurable arrays.

References

Adamski M. (1986): *Heuristic method of structural encoding of Petri net places.* — Zeszyty Naukowe WSI, No. 78, (in Polish).

Adamski M. (1987): *Direct implementation of Petri net specification.* — Proc. 7th Int. Conf. *Control Systems And Computer Science, CSCS*, Bucharest, Romania, Vol. 3, pp. 74–85.

Adamski M. (1990a): *Digital system design by formal transformation of specification.* — Proc. 35th Int. *Scien. Colloquium, IWK*, Ilmenau, Germany, Heft 3, pp. 62–65.

Adamski M. (1990b): *Digital Systems Design by Means of Rigorous and Structural Method.* — Technical University of Zielona Góra Press, Monograph No. 49.

Adamski M. (1991): *Parallel controller implementation using standard PLD software*, In: FPGAs (W.R. Moore, W. Luk, Eds.). — Abingdon EE&CS Books, Abingdon, UK, pp. 296–304.

Adamski M. (1993): *Petri nets in ASIC design.* — Applied Mathematics and Computer Science, Vol. 3, No. 1, pp. 169–180.

Adamski M. (1999): *Application specific logic controllers for safety critical systems.* — Proc. *Triennial IFAC World Congress*, Beijing, China, Vol. Q, pp. 519–524, Pergamon Press.

Adamski M. (2006a): *Behavioural specification of programs for modular reconfigurable logic controllers.* — Proc. Conf. *Mixed Design of Integrated Circuits and Systems, MIXDES,* Gdynia, Poland, pp. 239–244.

Adamski M. (2006b): *Reconfigurable logic controller for embedded applications*, In: Discrete-Event System Design 2006, Proc. Vol. from the IFAC Workshop *DESDes*, (Adamski M., L. Gomes, M. Węgrzyn and G. Łabiak (Eds.)). — pp. 147–152, University of Zielona Góra Press.

Adamski M. and Monteiro J.L. (1995): *Rule-based formal specification and implementation of Logic Controllers programs.* — Proc. IEEE Int. Symp. *Industrial Electronics, ISIE*, Athens, Greece, Vol. 2, pp. 700–705.

Adamski M. and Monteiro J.L. (1996): *Declarative specification of system independent logic controller programs.* — Proc. IEEE Int. Symp. *Industrial Electronics, ISIE*, Warsaw, Poland, pp. 305–310.

Adamski M. and Węgrzyn M. (1994): *Hierarchically structured coloured Petri net specification and validation of concurrent controllers.* — Proc. 39th Int. *Scientific Colloquium, IWK*, Ilmenau, Germany, Band 1, pp. 517–522.

Adamski M. and Węgrzyn M. (1999): *Field programmable implementation of concurrent state machine.* — Proc. 3rd Int. Conf. *on Computer-Aided Design of Discrete Devices, CAD DD*, Minsk, Belarus, Vol. 1, pp. 4–12.

Adamski M. and Zakrevskij A. (2001): *Formal specification of reactive logical control devices.* — Proc. World Multiconf. *Systemics, Cybernetics and Informatics, SCI*, Orlando, USA, Vol. 14, pp. 428–433.

Adamski M., Węgrzyn M. and Wolański P. (1997): *Simulating and synthesising of reconfigurable logic controllers using VHDL.* — Proc. 42nd Int. *Scientific Colloquium, IWK*, Ilmenau, Germany, Band 1, pp. 522–527.

Adamski M., Karatkevich A. and Węgrzyn M. (Eds.) (2005): *Design of Embedded Control Systems.* — New York: Springer.

Amroun A. and Bolton M. (1989): *Synthesis of controllers from Petri net descriptions and application of Ella.* — Proc. IMEC-IFIP Int. Workshop *Applied Formal Methods for Correct VLSI Design*, Leuven, Belgium, pp. 57–74.

Andrzejewski G. (2002): *Program model of interpreted Petri net for digital microsystems design.* — Ph.D. thesis, Szczecin University of Technology, Faculty of Information Technology, (in Polish).

Augin M., Boeri F. and Andre C. (1978): *New design using PLA and Petri Nets.* — Proc. Int. Symp. *Measurement and Control*, Athens, Greece, pp. 55–68.

Belhadj H., Gerbaux L., Bertrand M.-C. and Saucier G. (1993): *Specification and synthesis of communicating finite state machines.* — Proc. IFIP WG10.2/WG10.5 Workshops *Synthesis for Control Dominated Circuits*, Grenoble, France, Amsterdam: North-Holland Publishing, pp. 91–101.

Billington J., Christensen S., van Hee K., Kindler E., Kummer O., Petrucci L., Post R., Stehno C. and Weber M. (2003): *The Petri Net Markup Language: Concepts, Technology, and Tools.* — Proc. 24th Int. Conf. *Applications and Theory of Petri Nets, ICATPN*, Eindhoven, The Netherlands, Lecture Notes in Computer Science, (W.M.P. van der Aalst and E. Best, Eds.), Vol. 2679, pp. 483–505, Springer-Verlag.

Biliński K. (1996): *Application of Petri nets in parallel controllers design.* — Ph.D. thesis, University of Bristol, Electrical and Electronic Engineering Department.

Biliński K., Adamski M., Saul J.M. and Dagless E.L. (1994): *Petri net based algorithms for parallel controller synthesis.* — IEE Proc., Part E: Computers and Digital Techniques, Vol. 141, No. 6, pp. 405–412.

Chang N., Kwon W.H. and Park J. (1998): *Hardware implementation of real time Petri-net-based controllers.* — Control Engineering Practice, Vol. 6, No. 7, pp. 889–895.

Cortadella J., Yakovlev A. and Rozenberg A. (2002): *Concurrency and Hardware Design.* — Advances in Petri Nets, Serie: Lecture Notes in Computer Science, Vol. 2549, Springer, Berlin.

David R. and Alla H. (1992): *Petri Nets and Grafcet.* — New York: Prentice Hall.

Eles P., Kuchciński K. and Peng Z. (1998): *System Synthesis with VHDL.* — Boston: Kluwer Academic Publishers.

Fengler W., Wendt A., Adamski M. and Monteiro J.L. (1996): *Petri net based program design and implementation for controller systems.* — Proc. IFAC *Triennial World Congress*, San Francisco, CA, USA, Vol. J, pp. 425–429.

Fernandes J.M., Adamski M. and Proença A.J. (1997): *VHDL generation from hierarchical Petri net specifications of parallel controllers.* — IEE Proc., Part E: Computers and Digital Techniques, Vol. 144, No. 2, pp. 127–137.

Halang W. and Adamski M. (1997): *A programmable electronic system for safety related control applications.* — Proc. Int. Conf. *on Safety and Reliability, ESREL*, Lisbon, Portugal, pp. 349–356.

Kozłowski T., Dagless E.L., Saul J.M., Adamski M. and Szajna J. (1995): *Parallel controller synthesis using Petri nets.* — IEE Proc., Part E: Computers and Digital Techniques, Vol. 142, No. 4, pp. 263–271.

Łabiak G. (2003): *The Use of Hierarchical Model of Concurrent Automaton in Digital Controller Design.* — Ph.D. thesis, Warsaw University of Technology, Faculty of Electronics and Information Technology, (in Polish).

Mandado E., Marcos J. and Perez S.A. (1996): *Programmable Logic Devices and Logic Controllers.* — London: Prentice Hall.

Miczulski P. and Adamski M. (2006): *Analyses of safeness, liveness and persistence properties of Petri nets by means of monotype logic functions.* — Proc. IFAC Workshop *Discrete-Event System Design, DESDes*, Rydzyna, Poland, (Adamski, M., L. Gomes, M. Węgrzyn and G. Łabiak, Eds.), University of Zielona Góra Press, pp. 137–142.

Milik A. and Hrynkiewicz E. (2001): *Reconfigurable logic controller, architecture, programming, implementation.* — Proc. IFAC Workshop *Programmable Devices and Systems, PDS*, Gliwice, Poland, pp. 163–168.

Murata T. (1989): *Petri nets: Properties, analysis and applications.* — Proc. IEEE, Vol. 77, No. 4, pp. 541–580.

Nascimento P.S.B., Maciel P.R.M., Lima P.R.M., Santana R.E. and Filho A.G.S. (2004): *A partial reconfigurable architecture for controllers based on Petri Nets.* — Proc. 17th ACM Symp. *Integrated Circuits and System Design*, Pernambuce, Brazil, pp. 16–21.

Pardey J. and Bolton M. (1992): *Parallel controller synthesis for concurrent data paths.* — Proc. IFIP Workshop *Control Dominated Synthesis From a Register Transfer Level Description*, Grenoble, France, pp. 16–19.

Pardey J., Kozłowski T., Saul J. and Bolton M. (1992): *State assignment algorithms for parallel controller synthesis.* — Proc. IEEE Int. Conf. *Computer Design on VLSI in Computer and Processors, ICCD*, Cambridge, USA, pp. 316–319, Washington: IEEE Computer Society.

Pardey J., Amroun A., Bolton M. and Adamski M. (1994): *Parallel controller synthesis for programmable logic devices.* — Microprocessors and Microsystems, Vol. 18, No. 8, pp. 451–458.

Pastor E., Cortadella J. and Roig J. (2001): *Symbolic analysis of bounded Petri nets.* — IEEE Transactions on Computers, Vol. 50, No. 5, pp. 432–448.

Patel M. (1990): *Random logic implementation of extended timed Petri nets.* — Microprocessing and Microprogramming, Vol. 30, No. 1-5, pp. 313–319.

Thelen B. (1981): *Investigations of algorithms for computer-aided logic design of digital circuits*. — Ph.D. thesis, University of Karlsruhe, (in German).

Yakovlev A., Gomes L., Lavagno L. (Eds.) (2000): *Hardware Design and Petri Nets*. — Boston: Kluwer Academic Publisher.

Węgrzyn A. (2003): *Symbolic Analysis of Logical Control Devices using Selected Methods of Petri Net Analysis*. — Ph.D. thesis, Warsaw University of Technology, Faculty of Electronics and Information Technology, (in Polish).

Węgrzyn A. (2006): *Application of Databases for Management of Distributed Control Systems*. — Proc. IFAC Workshop *Discrete-Event System Design, DESDes*, (Adamski M., L. Gomes, M. Węgrzyn and G. Łabiak, Eds.), University of Zielona Góra Press, pp. 263–268.

Węgrzyn M. (1998): *Hierarchical implementation of logic controllers by means of Petri nets and FPGAs*. — Ph.D. thesis, Warsaw University of Technology, Faculty of Electronics and Information Technology, (in Polish).

Węgrzyn M. (2003): *Implementation of safety critical logic controller by means of FPGA*. — Annual Reviews in Control, Vol. 27, pp. 55–61.

Węgrzyn M. (2006): *Petri net decomposition approach for partial reconfiguration of logic controllers*. — Proc. IFAC Workshop *Discrete-Event System Design, DESDes*, (Adamski M., L. Gomes, M. Węgrzyn and G. Łabiak, Eds.), University of Zielona Góra Press, pp. 323–328.

Węgrzyn M. and Adamski M. (1999): *Hierarchical approach for design of application specific logic controller*. — Proc. IEEE Int. Symp. *Industrial Electronics, ISIE*, Bled, Slovenia, Vol. 3, pp. 1389–1394.

Węgrzyn M., Wolański P., Adamski M. and Monteiro J.L. (1996): *Field programmable device as a logic controller*. — Proc. 2nd Conf. *on Automatic Control*, Oporto, Portugal, Vol. 2, pp. 715–720.

Węgrzyn M., Adamski M. and Monteiro J.L. (1998): *The application of reconfigurable logic to controller design*. — Control Engineering Practice, Special Section on Custom Processes, Vol. 6, No. 7, pp. 879–887.

Wolański P., Węgrzyn M. and Adamski M. (1997): *VHDL modelling of industrial control systems*. — Proc. Int. *Scientific Colloquium, IWK*, Ilmenau, Germany, Band 1, pp. 528–533.

Zakrevskij A. (1999): *Parallel Algorithms for Logical Control*. — Press of Institute of Engineering Cybernetics of NAS of Bielarus, Minsk, (in Russian).

Chapter 15

DESIGN OF CONTROL UNITS WITH PROGRAMMABLE LOGIC DEVICES

Alexander BARKALOV*, Larysa TITARENKO*

15.1. Introduction

One of the basic features of modern society is wide usage of digital systems in all areas of its activity. According to the principle of microprogram control (Wilkes, 1951), any digital system includes two main parts (Fig. 15.1):

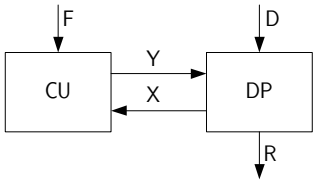


Fig. 15.1. Structure of a digital system

Here CU is the Control Unit, which can be treated as the brain of the system, and DP is the Data Path, which is an executive part of the system. The DP receives and keeps the words of information D, which should be processed, executes the microoperations Y concerning these words, estimates the values of logical conditions X and calculates the results R of these operations. The CU provides the required distribution order of microoperations in time. This order depends on the control algorithm F to be executed and the values of logical conditions X.

The methods of DP design are well known (Adamski and Barkalov, 2006; Barkalov, 2003; Clements, 2000), and they are within the scope of this chapter. The methods of control unit design depend strongly on logic elements used to implement the logic circuit of the CU. The rapid evolution in the area of semiconductor microelectronics has resulted in the appearance of Very Large Scale Integration (VLSI) microchips with

* Institute of Computer Engineering and Electronics
e-mails: {A.Barkalov, L.Titarenko}@iie.uz.zgora.pl

more than 1 billion of transistors. The power of modern VLSI is enough to implement a complex digital system in a single chip. When such VLSI includes programmable logic blocks, they are named Systems-on-a-Programmable-Chip (SoPCs) (Grushnitski *et al.*, 2002; Maxfield, 2004). The SoPC includes some tools for the implementation of arbitrary logic of a digital system and Embedded Memory Blocks (EMB) for the implementation of table functions.

An arbitrary logic is implemented here using such elements as Complex Programmable Logic Devices (CPLDs) (Kania, 2004; Solovjev, 2001) or Field-Programmable Gate Arrays (FPGAs) (Jenkins, 1994; Łuba *et al.*, 1997; Maxfield, 2004). Different approaches should be used for the optimization of the hardware amount in the cases of CPLDs and FPGAs. Design with CPLD needs minimizing each Boolean function due to a limited number of internal words (terms) in each macrocell of the CPLD (Solovjev, 2001). The limited amount of the inputs of the FPGA Look-Up Table (LUT) element leads to decreasing the number of input variables for a function to be implemented (Maxfield, 2004).

The classical model of the CU is a Finite-State-Machine (FSM) (Baranov, 1994), which can be represented as a composition of the Combinational Circuit (CC) and the ReGister (RG) (Fig. 15.2).

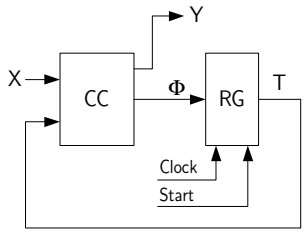


Fig. 15.2. Model of a finite-state machine

An FSM is characterized by a set of internal states $A = \{a_1, \ldots, a_M\}$, a set of logical conditions $X = \{x_1, \ldots, x_L\}$ and a set of output signals $Y = \{y_1, \ldots, y_N\}$, which are named microoperations M_0. The state $a_m \in A$ is represented by the code $K(a_m)$ with $R =]\log_2 M[$ bits, which is kept in the register. Each bit of this code is represented by a single internal variable $T_r \in T$. The excitation functions $\Phi = \{\varphi_1, \ldots, \varphi_R\}$ are used to change the content of the RG. As a rule, the register is implemented using D flip-flops (Barkalov, 2002). The pulse *Start* is used to load the code of the initial state $a_1 \in A$ into the RG, the pulse *Clock* is used to point out the instants to change the content of the RG.

The combinational circuit, can be described by systems of functions,

$$\Phi = \Phi(T, X), \tag{15.1}$$

$$Y = Y(T, X), \tag{15.2}$$

$$Y = Y(T). \tag{15.3}$$

The systems (15.1)–(15.2) correspond to the Mealy FSM and the systems (15.1), (15.3) correspond to the Moore FSM. As a rule, the model of the Moore FSM is more often used in the practice of control units design (Solovjev, 2001).

In this chapter we are to discuss the methods of the design and optimization of the following control units based on the Moore model:

- classical Moore FSM;
- Microprogram Control Unit (MCU);
- Compositional Microprogram Control Unit (CMCU).

The discussed methods of design will be oriented towards CPLDs and FPGAs, an initial control algorithm will be represented by the Flow-Chart of Algorithm (FCA) (Baranov, 1994). Such a form of specification is chosen because it permits to understand all methods under discussion better than, for example, VHDL representation, which is used now for the design of complex digital devices (Łuba, 2003; Zwoliński, 2002).

15.2. Design and optimization of the Moore FSM

This model is used when maximal performance of the control unit is the main goal of the project under design (Barkalov and Węgrzyn, 2006). The method of Moore FSM design includes the following steps (Baranov, 1994):

1. Construction of the marked flow-charts Γ.
2. Encoding of the states $a_m \in A$.
3. Construction of a direct structural table of the FSM.
4. Construction of the system of excitation functions.
5. Construction of the system of microoperations.
6. Implementation of the logic circuit of the FSM.

Let us discuss an example of Moore FSM S_1 design using the flow-chart Γ_1 (Fig. 15.3), which is marked by the states $a_1, \ldots, a_7$.

Because $M = 7$, we have $R = 3$ and $T = \{T_1, T_2, T_3\}$. Let us encode the states in the following order: $K(a_1) = 000, K(a_2) = 001, \ldots, K(a_7) = 110$. The direct structural table of the Moore FSM includes the following columns (Baranov, 1994): a_m is a current state of the FSM, $a_m \in A$; $K(a_m)$ is a code of the current state; a_s is a state of transition, $a_s \in A$; $K(a_s)$ is a code of this state; X_h is an input signal, which determines the transition $\langle a_m, a_s \rangle$, and it is equal to the conjunction of some elements of the set of logical conditions; Φ_h is a set of excitation functions, which are equal to 1 to switch the register from $K(a_m)$ to $K(a_s)$; h is a number of transition ($h = 1, \ldots, H$). The column a_m of the Direct Structural Table (DST) contains the set of microoperations $Y(a_m) \subseteq Y$, which are equal to 1 in the state $a_m \in A$. In the case under discussion, the DST includes $H = 12$ lines (Table 15.1).

The system (15.1) is formed from the DST as the following one:

$$D_r = \bigvee_{h=1}^{H} C_{rh} A_m^h X_h (r = 1, \ldots, R), \tag{15.4}$$

where C_{rh} is a Boolean variable that is equal to 1 iff the function D_r is written in the h-th line of the DST; A_m^h is a conjunction of internal variables corresponding to the

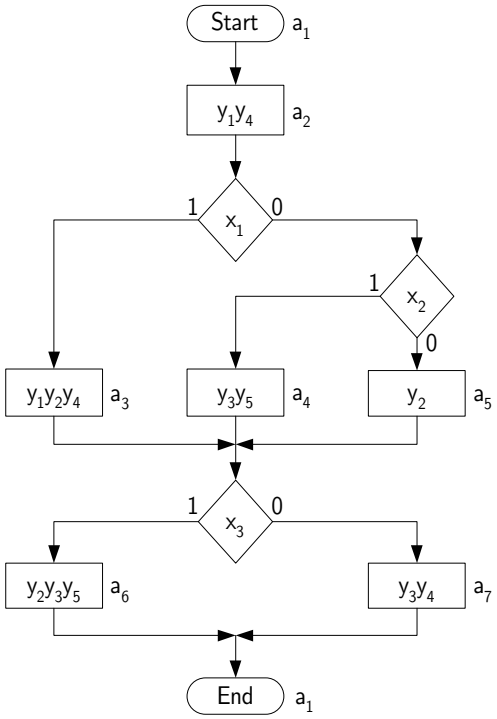


Fig. 15.3. Marked flow-chart Γ_1

code $K(a_m)$ of the current state from the h-th line of the DST. For example, we can form from Table 15.1

$$D_3 = \bar{T}_1\bar{T}_2\bar{T}_3 \vee \bar{T}_1\bar{T}_2T_3\bar{x}_1x_2 \vee \bar{T}_1T_2\bar{T}_3x_3 \vee \bar{T}_1T_2T_3x_3 \vee T_1\bar{T}_2\bar{T}_3x_3.$$

The system (15.3) is formed from the DST as

$$y_n = \bigvee_{m=1}^{M} C_{nm}A_m (n = 1, \ldots, N), \tag{15.5}$$

where C_{nm} is a Boolean variable that is equal to 1 iff $y_n \in Y(a_m)$. For example, we can obtain from Table 15.1

$$y_1 = A_2 \vee A_3 = \bar{T}_1\bar{T}_2T_3 \vee \bar{T}_1T_2\bar{T}_3.$$

The trivial structure of the Moore FSM logic circuit includes two combinational circuits and a register (Fig. 15.4).

Let us denote this structure as U_1 and let $U_i(\Gamma_j)$ mean that the circuit of the FSM U_i is implemented using a control algorithm represented by FCA Γ_j. In the case of the FSM U_1, the CC is implemented on an FPGA or a CPLD on the basis of the system (15.4). The Circuit of Formation of MicroOperations (CFMO) can be implemented using an FPGA, a CPLD or an EMB.

Table 15.1. Direct structural table of the Moore FSM S_1

a_m	$K(a_m)$	a_s	$K(a_s)$	X_h	Φ_h	h
$a_1(-)$	000	a_2	001	1	D_3	1
$a_2(y_1y_4)$	010	a_3	010	x_1	D_2	2
		a_4	011	$\bar{x}_1x_2$	D_2D_3	3
		a_5	100	$\bar{x}_1\bar{x}_2$	D_1	4
$a_3(y_1y_2y_3)$	010	a_6	101	x_3	D_1D_3	5
		a_7	110	$\bar{x}_3$	D_1D_2	6
$a_4(y_3y_5)$	011	a_6	101	x_3	D_1D_3	7
		a_7	110	$\bar{x}_3$	D_1D_2	8
$a_5(y_2)$	100	a_6	101	x_3	D_1D_3	9
		a_7	110	$\bar{x}_3$	D_1D_2	10
$a_6(y_2y_3y_5)$	101	a_1	000	1	—	11
$a_7(y_3y_4)$	110	a_1	000	1	—	12

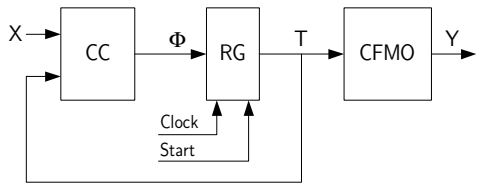


Fig. 15.4. Trivial structure of the Moore FSM circuit

There are many methods of optimizing the hardware amount in the case of the FSM U_1 (Barkalov, 2002; Barkalov and Węgrzyn, 2006). We will discuss three main methods of optimization:

1. optimal encoding of the states,
2. special encoding of the states,
3. transformation of the codes of the states into the codes of the classes of pseudoequivalent states.

An analysis of Table 15.1 shows that the columns a_s – Φ_h have identical information for the lines 5, 7, 9, and for the lines 6, 8, 10, and for the lines 11, 12. This is connected with the fact that, for example, the states a_2, a_3, a_4 of the FSM $U_1(\Gamma_1)$ correspond to the same state of the equivalent Mealy FSM (Barkalov, 1998). Let us name the states $a_m, a_s \in A$ as the pseudoequivalent states of the Moore FSM. Let us find the partition of the set A on the classes of the Pseudoequivalent States (PS): $\Pi_A = \{B_1, \ldots, B_I\}$. In the case of the FSM $U_1(\Gamma_1)$, the partition Π_A includes $I = 4$ the classes $\Pi_A = \{B_1, \ldots, B_4\}$, where $B_1 = \{a_1\}$, $B_2 = \{a_2\}$, $B_3 = \{a_3, a_4, a_5\}$, $B_4 = \{a_6, a_7\}$.

Optimal encoding of the states (Barkalov, 1998) is executed in a such way that all states $a_m \in B_i$ belong to the same generalized interval of the the R-dimensional Boolean space. These intervals are considered as the codes of particular classes. Such encoding can be worked out using, for example, special algorithms from the work (Achasova, 1987). In the case of the FSM $U_1(\Gamma_1)$, the result of optimal encoding of the states is shown by the Karnaugh map (Fig. 15.5).

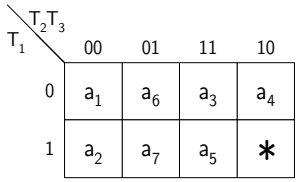


Fig. 15.5. Optimal encoding of the states of the FSM $U_1(\Gamma_1)$

The states a_3, a_4, a_5, for example, belong to the interval $\langle *, 1, * \rangle$ and, thus, the code $K(B_3)$ of the class $B_3 \in \Pi_A$ is determined as $K(B_3) = *1*$. In the same way we can find the following codes: $K(B_1) = 000$, $K(B_2) = 1 * 0$, $K(B_4) = *01$.

Now we can replace the states $a_m \in A$ by their classes $B_i \in \Pi_A$ (in the column of the current state of the DST), the codes $K(a_m)$ should be replaced by $K(B_i)$, where $a_m \in B_i$, and only one of the identical lines of the DST should stay in the final transformed DST. The transformed DST includes the columns B_i, $K(B_i)$, a_s, $K(a_s)$ X_h, Φ_h, h. Let $H_i(\Gamma_j)$ be the number of lines of the DST from the FSM $U_i(\Gamma_j)$, then $H_2(\Gamma_1) = 6$ (Table 15.2), where U_2 stays for the Moore FSM with optimal encoding of the states.

This table is the base for constructing the system (15.1), which is now represented as

$$D_r = \bigvee_{h=1}^{H_2(\Gamma)} C_{rh} B_i^h X_h (r = 1, \ldots, R), \tag{15.6}$$

where

$$B_i^h = \bigwedge_{r=1}^{R} T_r^{e_{ir}} (i = 1, \ldots, I), \tag{15.7}$$

Table 15.2. Transformed DST of the Moore FSM $U_2(\Gamma_1)$

B_i	$K(B_i)$	a_s	$K(a_s)$	X_h	Φ_h	h
B_1	000	a_2	100	1	D_1	1
B_2	1*0	a_3	011	x_1	$D_2 D_3$	2
		a_4	010	$\bar{x}_1 x_2$	D_2	3
		a_5	111	$\bar{x}_1 \bar{x}_2$	$D_1 D_2 D_3$	4
B_3	*1*	a_6	001	x_3	D_3	5
		a_7	101	$\bar{x}_3$	$D_1 D_3$	6

and $e_{ir} \in \{0, 1, *\}$, $T_r^0 = \bar{T}_r$, $T_r^1 = T_r$, $T_r^* = 1 (r = 1, \ldots, R)$. For example, we can get $D_3 = T_1\bar{T}_3x_1 \vee T_1\bar{T}_3\bar{x}_1\bar{x}_2 \vee T_2$. That formula includes only 8 literals, but in the case of the FSM $U_1(r_1)$ it has 20 literals. This method is oriented towards CPLD implementation of the CC.

Special encoding of the states (Barkalov, 2002) is executed in a such way that all functions $y_n \in Y$ have a minimal possible amount of terms. Let us encode the states of the FSM S_1 in a manner shown in Fig. 15.6.

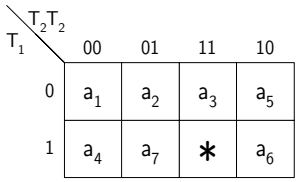


Fig. 15.6. Special encoding of the states of the FSM S_1

Let us find the Boolean formula for the functions $y_n \in Y$: $y_1 = A_2 \vee A_3 = \bar{T}_1T_3$; $y_2 = A_3 \vee A_5 \vee A_6 = T_2$; $y_3 = A_4 \vee A_6 \vee A_7 = T_1$, $y_4 = A_2 \vee A_3 \vee A_7 = T_3$; $y_5 = A_4 \vee A_6 = T_1\bar{T}_3$.

Now each MO $y_n \in Y$ is expressed by one term. This method leads to the FSM $U_3(\Gamma)$, where the CFMO is implemented using an FPGA or a CPLD.

It is clear that the methods of states encoding have different goals in the cases of U_2 and U_3. It is possible to minimize both the CC and the CFMO using the following approach:

The Transformation of the Codes (TC) leads to the Moore FSM $U_4(\Gamma)$, where the circuit TC forms the functions $T_r \in T$ (Fig. 15.7).

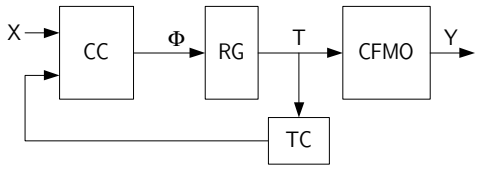


Fig. 15.7. Structure of the Moore FSM U_4

In this case each class $B_i \in \Pi_A$ is identified by the code $K(B_i)$, which has R_1 bits, where $R_1 =]\log_2 I_0[$. Here $I_0 = |\Pi'_A|$, $\Pi'_A \subseteq \Pi_A$ is a part of the partition Π_A with the states $a_m \in B_i$, where there is no edge $\langle a_m, a_1 \rangle$.

The method of Moore FSM U_4 design includes such specific steps as

- encoding of the classes $B_i \in \Pi'_A$;
- formation of the table of the code transformer TC.

In the case of the FSM $U_4(\Gamma_1)$ we have $\Pi'_A = \{B_1, B_2, B_3\}$, $I_0 = 3$, $R_1 = 2$, $T = \{T_1, T_2\}$. Let the classes $B_i \in \Pi'_A$ be encoded in the following manner: $K(B_1) = 00$, $K(B_2) = 01$, $K(B_3) = 11$. If the states $a_m \in A$ are encoded in the way shown

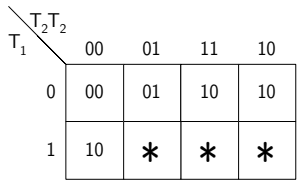


Fig. 15.8. Karnaugh map for the code transformer of the FSM $U_4(\Gamma_1)$

in Fig. 15.6, then the code transformer can be represented by the Karnaugh map (Fig. 15.8).

Using this map, we can form a system of Boolean functions,

$$\tau = \tau(T), \tag{15.8}$$

where $\tau_1 = T_1 \vee \bar{T}_1 T_2 = T_1 \vee T_2$, $\tau_2 = \bar{T}_2 T_3$. The transformed DST of the Moore FSM $U_4(\Gamma_1)$ is formed in the same way as the table of the FSM $U_2(\Gamma_1)$, and it includes $H_4(\Gamma_1) = 6$ lines.

Let us point out that none of the discussed methods decreases the performance of the control unit in comparison with the performance of the FSM U_1.

15.3. Design of microprogram control units

The structure of the MCU U_5 is based on the classical model of Wilkes (1951) and can be represented as a composition of the SeQuencer (SQ) and the Control Memory (CM) (Fig. 15.9).

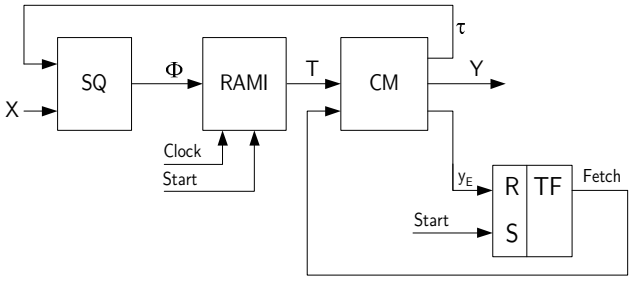


Fig. 15.9. Structural diagram of the MCU U_5

The MCU U_5 operates in the following manner (Barkalov and Węgrzyn, 2006): if the pulse $Start = 1$, then the address of the first microinstruction of the microprogram is loaded into the Register of the Address of MicroInstruction (RAMI). At the same time, $Fetch = 1$, where the signal Fetch permits reading the out control memory and is the output of flip-flop TF. The current microinstruction (MI) is fetched from the CM and the corresponding microoperations $y_n \in Y$ initialize some actions in the data-path of a digital system. New values of the logical conditions $x_e \in X$ and the control part of the MI that is represented by the variables τ are used by the SQ to form excitation functions,

$$\Phi = \Phi(\tau, X), \tag{15.9}$$

which are used to load the address of the next MI into the RAMI. The operation is terminated when the signal $y_E = 1$. This signal indicates the end of the microprogram.

Such devices were very popular in the 1960s and 1970s. Now they can be in use because of their regular structure and a very simple circuit SQ. They can be used if the resulting performance of a digital system is sufficient from the point of initial constraints of a particular project. The methods of designing such units are discussed in (Barkalov, 2003; Barkalov and Palagin, 1997; Barkalov and Węgrzyn, 2006; Salisbury, 1976), and they depend strongly on the method of microinstructions addressing. In this chapter we are to discuss the method of designing the MCU U_6 with natural addressing of microinstructions (Fig. 15.10).

Fig. 15.10. Formats of the microinstructions of the MCU U_6

There are two types of microinstructions, which differ by the Field of Attribute (FA). Let the value FA=0 correspond to Operational MicroInstruction (OMI) and the value FA=1 correspond to Control MicroInstruction (CMI). The microoperations to be executed are determined by the field FY, the field FX determines the logic condition to be checked, and the field FA_0 contains the address of transition. Let $[F]^t$ mean the content of some field F in the instant t, A^t stand for the address of the current MI and A^{t+1} determine the address of transition ($t = 1, 2, \dots$). The address of transition for the MCU U_6 is determined in the following order (Barkalov and Węgrzyn, 2006):

$$A^{t+1} = \begin{cases} A^t + 1, & \text{if} \quad [FA]^t = 0, \\ [FA_0]^t, & \text{if} \quad [FX]^t = \varnothing, \\ A^t + 1, & \text{if} \quad x_e^t = 1, \\ [FA_0]^t, & \text{if} \quad x_e^t = 0, \end{cases} \tag{15.10}$$

where x_e^t is the value of the logical condition to be checked into the instant $t (t = 1, 2, \dots)$. Let us discuss an example of MCU $U_6(\Gamma_1)$ design, using the procedure from (Barkalov and Węgrzyn, 2006).

Transformation of the initial flow-chart. This step is executed to eliminate the conflicts of addressing and to organize the termination mode of the MCU. The transformed FCA $\Gamma_1(U_6)$ is shown in Fig. 15.11.

Here the numbers from 1 to 11 correspond to the indexes of microinstructions. There are three sets of microinstructions in the case of the MCU $U_6(\Gamma_1)$: the set $1, 3, 5, 6, 10, 11$, including OMIs, the set $2, 4, 9$ of CMIs of conditional jumps, the sets $7, 8$ of CMIs that have been included to eliminate the conflicts of addressing among the OMIs 3, 5 and 6. The signal y_E is inserted into the operational nodes connected with the final node b_E.

Addressing of microinstructions. Let us form the following sequences of microinstructions: $\beta_1 = \langle 1, 2, 3, 9, 10 \rangle$, $\beta_2 = \langle 4, 5, 7 \rangle$, $\beta_3 = \langle 6, 8 \rangle$, $\beta_4 = \langle 11 \rangle$. Let Q be the number of the nodes in the transformed FCA $\Gamma(U_6)$; then the final microprogram includes Q microinstructions and they can be addressed using $R_2 =]\log_2 Q[$ bits,

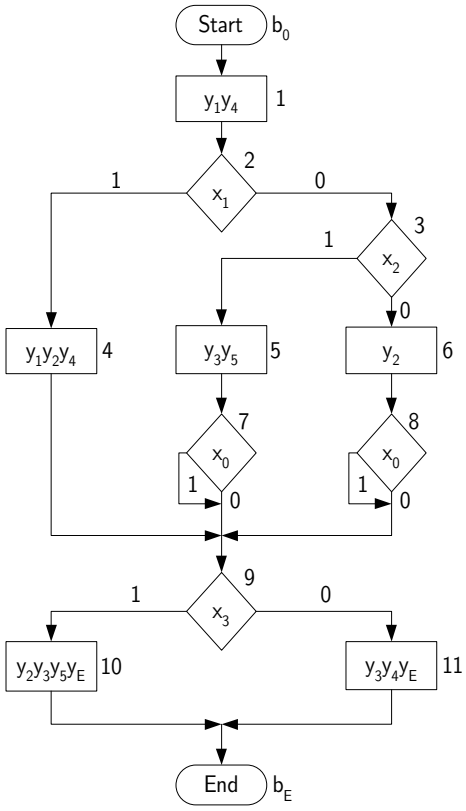


Fig. 15.11. Transformed flow-chart $\Gamma_1(U_6)$

$|T| = |Q| = R_2$. The addresses of the microinstructions of the MCU $U_6(\Gamma_1)$ are shown in Fig. 15.12.

T_1T_2 \ T_3T_4	00	01	10	11
00	1	2	3	9
01	10	4	5	7
10	6	8	11	*
11	*	*	*	*

Fig. 15.12. Addressing of the microinstructions of the MCU $U_6(\Gamma_1)$

Encoding of operational and control parts of MIs can be executed with a variety of approaches, but here we will use one-hot encoding of MOs and maximal encoding of logical conditions (Barkalov and Palagin, 1997). Let m_I, m_O, m_A mean respectively the length of the MI, its operational and address parts. These parameters can be

calculated in the following manner:

$$\begin{gathered} m_I = \max(1 + m + O, 1 + m_A), \\ m_O = N + 1, \\ m_A =]\log_2(L + 1)[+ R_2 = m_L + R_2, \end{gathered} \tag{15.11}$$

where m_L is the length of the field FX.

In our example we have $m_O = 6$, $m_L = 2$, $m_A = 6$, $m_I = 7$. Let $K(x_0) = 00$, $K(x_1) = 01$, $K(x_2) = 10$, $K(x_3) = 11$, and let the bits of the MI be represented by the set $Z = \{z_1, \ldots, z_7\}$.

The construction of the content of the control memory is reduced to the formation of a table with Q lines and m_I columns. In the case of the MCU $U_6(\Gamma_1)$ this table includes $q = 11$ lines (Table 15.3). If $z_1 = 0$, then the bit $z_7 = y_E$, otherwise it is treated as D_4.

Table 15.3. Content of the control memory of the MCU $U_6(\Gamma_1)$

Address of MI	FA	FY		Formula of transition
$T_1T_2T_3T_4$		FX	FA_0	
	z_1	z_2z_3	$z_4z_5z_6z_7$	
0000	0	10	0100	$1 \longrightarrow 2$
0001	1	01	0101	$2 \longrightarrow x_1 3 \vee \bar{x}_1 4$
0010	0	11	0100	$3 \longrightarrow 9$
0011	1	11	1010	$9 \longrightarrow x_3 10 \vee \bar{x}_3 11$
0100	0	01	1011	$10 \longrightarrow b_E$
0101	1	10	1000	$4 \longrightarrow x_2 5 \vee \bar{x}_2 6$
0110	0	00	1010	$5 \longrightarrow 7$
0111	1	00	0011	$7 \longrightarrow 9$
1000	0	01	0000	$6 \longrightarrow 8$
1001	1	00	0011	$8 \longrightarrow 9$
1010	0	00	1101	$11 \longrightarrow b_E$

The design of the sequencer is executed using standard multiplexers. In the case of the MCU U_6, the RAMI is replaced by the CounTer (CT) that executes two operations:

$$V_1 \# CT := CT + 1, \quad V_2 \# CT := \langle \Phi \rangle. \tag{15.12}$$

According to (15.12), the SQ should form the signals V_1 and V_2 to initialize the corresponding actions.

In the case of the MCU $U_6(\Gamma_1)$, The MultipleXer (MX) has $L+1 = 4$ informational inputs, $m_L = 2$ control inputs and an input of Chip Selection (CS) (Fig. 15.13). The CT has the control inputs $C_1 \# CT := CT+1$, $C_2 \# CT := \langle z_4 z_5 z_6 z_7 \rangle$ and $R \# CT := 0$.

Two gates AND are used to distribute the pulse *Clock* between the control inputs C_1 and C_2. The organization of the SQ and the CT of the MCU $U_6(\Gamma_1)$ is clear from Fig. 15.13.

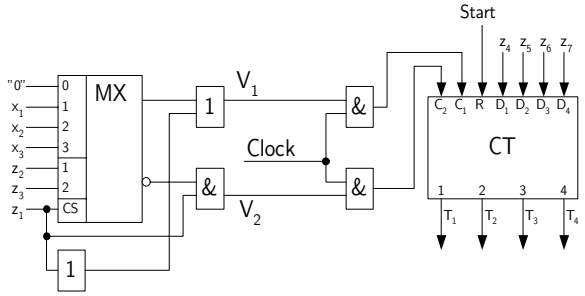


Fig. 15.13. Organization of the SQ and the CT of the MCU $U_6(\Gamma_1)$

The implementation of the MCU circuit is reduced to the design of the MX using LUT-elements (or the cells of the CPLD) and the design of the CM using EMBs.

The main advantage of the MCU is a very simple and transparent implementation of the control algorithm. But there are some drawbacks that can restrict the application of the MCU to modern digital systems (Barkalov and Węgrzyn, 2006):

i) Only one logic condition is checked during one cycle of MCU operation. The microprogram can include additional microinstructions of "if-else" and "go to" types. It leads to increasing the time of control algorithm interpretation in comparison with the Moore FSM.

ii) The format of the MI can include both operational and control parts. It means that the control memory can need more EMBs in comparison with the CFMO of the equivalent Moore FSM.

In the case of linear flow-charts these drawbacks can be eliminated due to the usage of the model of the Compositional Microprogram Control Unit (CMCU) (Barkalov and Palagin, 1997).

15.4. Design and optimization of compositional microprogram control units

Let the control algorithm be represented by the FCA $\Gamma = \Gamma(B, E)$, where $B = O_\Gamma \cup C_\Gamma \cup \{b_O, b_E\}$. Here O_Γ is a set of operational nodes, C_Γ is a set of conditional nodes, E is a set of edges. Let us use some definitions from (Barkalov and Palagin, 1997); we will need to explain the methods of design.

Definition 15.1. An Operational Linear Chain (OLC) of the FCA Γ is a finite sequence of operational nodes $\alpha_g = \langle b_{g1}, \ldots, b_{gF_g} \rangle$ such that there is an edge $\langle b_{gi}, b_{gi+1} \rangle \in E$ for each pair of adjacent components of the vector $\alpha_g (i = 1, \ldots, F_{g-1})$.

Definition 15.2. The node $b_g \in O^g$, where $O^g \subseteq O_\Gamma$ is a set of components of the OLC α_g, is named an input of the OLC α_g if there is an edge $\langle b_t, b_g \rangle \in E$, where $b_t \ni O^g$.

Definition 15.3. The node $b_g \in O^g$ is named an output of the OLC α_g if there is an edge $\langle b_g, b_t \rangle \in E$, where $b_t \ni O^g$.

In the common case, OLC α_g can have inputs $I_g^1, I_g^2, \dots$ and only one output O_g. Let a set of the OLC $C = \{\alpha_1, \dots, \alpha_G\}$ be formed for some FCA Γ and let it satisfy the condition

$$\begin{gathered} O^1 \cup O^2 \cup \dots \cup O^G = O_\Gamma, \\ O^i \cap O^j = \varnothing (i \neq j, \\ i, j \in 1, \dots, G), \quad G \longrightarrow \min . \end{gathered} \tag{15.13}$$

Let each node $b_g \in O_\Gamma$ correspond to the microinstruction with the address $A(b_g)$. Let us address these microinstructions to hold the condition

$$A(b_{gi+1}) = A(b_{gi}) + 1, \tag{15.14}$$

where $g = 1, \dots, G$, $i = 1, \dots, F_g - 1$. Equation (15.14) determines the mode of natural addressing of microinstructions (Barkalov and Węgrzyn, 2006).

Now this particular FCA Γ can be interpreted by the CMCU U_7 (Fig. 15.14), which is named a CMCU with the base structure (Barkalov and Palagin, 1997). The compositional MCU U_7 can be viewed as composition of the Mealy automaton of addressing (it includes the combinational circuit CC, and the register RG) and the MCU with natural addressing of microinstructions (it includes the control memory CM, and the counter CT).

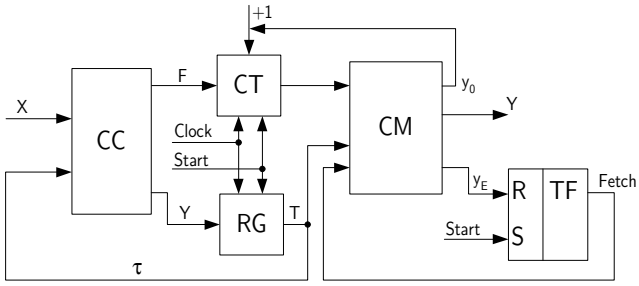


Fig. 15.14. Structural diagram of the CMCU U_7

The CMCU U_7 operates in the following manner: if the pulse *Start*=1, then the address of the first microinstruction of the microprogram is loaded into the CT and the code of the initial state is loaded into the RG. In the same instant, *Fetch*=1 and microinstructions can be read out of the CM. Let in the instant $t (t = 1, 2, \dots)$ the counter contain the address $A(b_q)$, where $b_q \in O^g$, and let the register keep the code $K(a_m)$ of the current state the of automaton of addressing. If $b_q \neq O_g$, then a special signal y_0 is formed together with the microoperations $Y(b_q) \subseteq Y$. If $y_0 = 1$, then the content of the CT is incremented and it corresponds to the mode (15.14). The content of the RG is kept without changes and the next component of the OLC $\alpha_g \in C$ will be interpreted in the instant $t+1$. If $b_q = O_g$, then $y_0 = 0$. In this case, the CC forms

excitation functions,

$$\Phi = \Phi(T, X), \tag{15.15}$$

$$\Psi = \Psi(T, X). \tag{15.16}$$

The functions Φ form the address of transition and load it into the CT, the functions Ψ form the code of the next state of the addressing automaton and load it into the RG. It means that the interpretation of some next OLC is started. If the CT contains the address $A(b_q)$ and there is an edge $\langle b_q, b_E\rangle \in E$, then $y_E = 1$. The terminates the operation of the CMCU.

The method of CMCU U_7 design includes the following steps (Barkalov and Węgrzyn, 2006):

i) construction of the set C according to (15.13);
ii) natural addressing of microinstructions;
iii) formation of the content of the control memory;
iv) transformation of the initial FCA;
v) construction of the DST of the FSM of addressing;
vi) design of the logic circuit of the CMCU.

Let us discuss an example of the design of the CMCU $U_7(\Gamma_2)$, where the FCA Γ_2 is shown in Fig. 15.15. Using the methods from (Barkalov and Palagin, 1997), we can form the following set of the OLC: $C = \{\alpha_1, \ldots, \alpha_4\}$, where $\alpha_1 = \langle b_1, b_2\rangle$, $I_1^1 = b_1$, $O_1 = b_2$; $\alpha_2 = \langle b_3, b_4, b_5\rangle$, $I_2^1 = b_3$, $I_2^2 = O_2 = b_5$; $\alpha_3 = \langle b_6, b_7\rangle$, $I_3^1 = b_6$, $O_3 = b_7$; $\alpha_4 = \langle b_8\rangle$, $I_4^1 = O_4 = b_8$. Therefore, $G = 4$, $Q = 8$, $R_2 = 3$, $T = \{T_1, T_2, T_3\}$, $\Phi = \{D_1, D_2, D_3\}$.

Let us form the vector $\alpha = \alpha_1 * \alpha_2 * \alpha_3 * \alpha_4$, where $*$ is a sign of concatenation: $\alpha = \langle b_1, b_2, \ldots, b_8\rangle$. Let the first component of this vector have the number $N_1 = 0$, the second one the number $N_2 = 1$ and so on. Let us replace each number N_g by the binary code $A(b_g)$ with $R_2 = 3$ bits. These codes are equal to the addresses of the corresponding microinstructions: $A(b_1) = 000, A(b_2) = 001, \ldots, A(b_8) = 111$. Now the microinstructions of the CMCU $U_7(\Gamma_2)$ are addressed according to (15.14).

The table of the CM includes the following columns: Address, FA, FY, "Formula of transition", where the fields FA, FY have the same sense as in the case of the MCU. The field FA contains the signal y_0, and y_1 corresponds to natural addressing of microinstructions. The table of the CM of the CMCU $U_7(\Gamma_2)$ includes $H_7(\Gamma_2)$ lines (Table 15.4).

In this table, the record "2" means that the address of transition is formed by the CC; the record "$5 \rightarrow$ *End*" means that operation of the CMCU will be terminated.

To design the CC we should form a system of formulae of transitions for the initial flow-chart. In the case of the FCA Γ_2, this system is the following one:

$$b_0 \longrightarrow I_1^1,$$

$$I_1^1 \longrightarrow x_1 I_2^1 \vee \bar{x}_1 x_2 I_2^2 \vee \bar{x}_1 \bar{x}_2 x_3 I_3^1 \vee \bar{x}_1 \bar{x}_2 \bar{x}_3 I_4^1, \tag{15.17}$$

$$I_2^1, I_2^2, I_3^1, I_4^1 \longrightarrow b_E.$$

The system (15.17) is the base to form the transformed FCA $\Gamma_2(U_7)$ that is used to construct the direct structural table of the Mealy FSM. Each operational node of

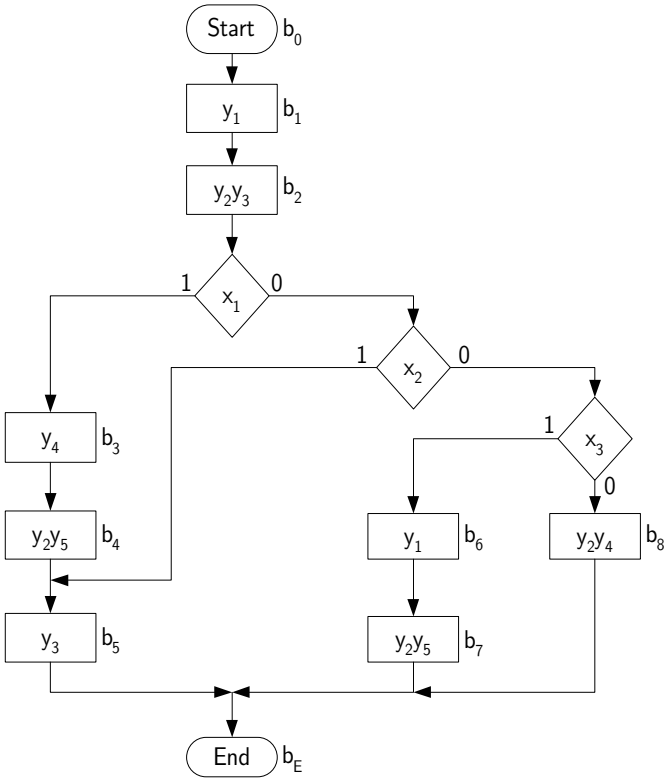


Fig. 15.15. Initial flow-chart Γ_2

Table 15.4. Table of the content of the CM of the CMCU $U_7(\Gamma_2)$

Address			FA	FY						Formula of transition
T_1	T_2	T_3	y_0	y_1	y_2	y_3	y_4	y_5	y_E	
0	0	0	1	1	0	0	0	0	0	$1 \longrightarrow 2$
0	0	1	0	0	1	1	0	0	0	2
0	1	0	1	0	0	0	1	0	0	$3 \longrightarrow 4$
0	1	1	1	0	1	0	0	1	0	$4 \longrightarrow 5$
1	0	0	0	0	0	1	0	0	1	$5 \longrightarrow End$
1	0	1	1	1	0	0	0	0	0	$6 \longrightarrow 7$
1	1	0	0	0	1	0	0	1	1	$7 \longrightarrow End$
1	1	1	0	0	1	0	1	0	1	$8 \longrightarrow End$

the transformed FCA corresponds to some input of the OLC $\alpha_g \in C$. The content of a particular operational node is determined by the address of the corresponding microinstruction. For example, the input I_2^2 corresponds to the node b_5 and the microinstruction MI_5 with the address $A(b_5) = 100$ corresponds to this node. It means

that the node I_2^2 of the transformed FCA contains the function D_1 to form the address $A(b_5)$ into the CT. The transformed FCA $\Gamma_2(U_7)$ is shown in Fig. 15.16.

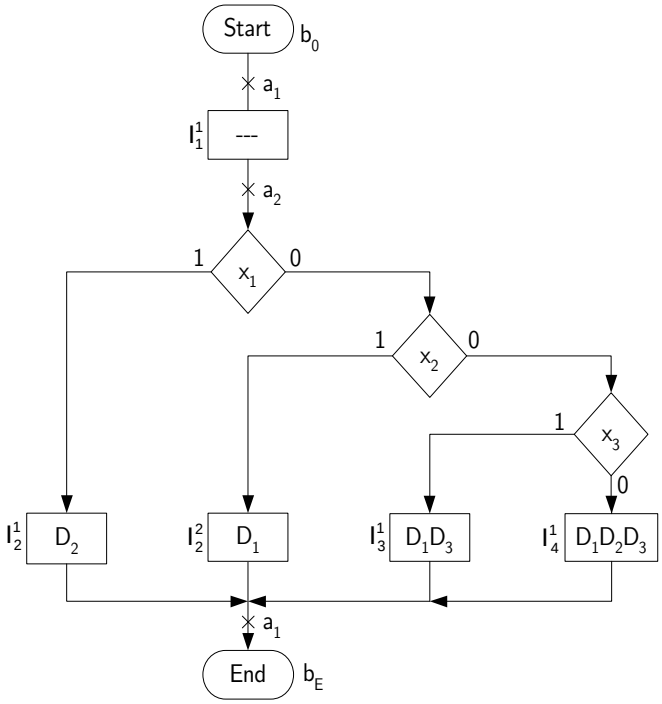


Fig. 15.16. Transformed flow-chart $\Gamma_2(U_7)$

As it follows from Fig. 15.16, the set of internal states of the Mealy FSM $A = \{a_1, a_2\}$, $M_1 = 2$. It means that

$$R_3 =]\log_2 M_1[, \tag{15.18}$$

where R_3 is the amount of bits to encode the states $a_m \in A$. So, $\tau = \{\tau_1\}$, $K(a_1) = 0$, $K(a_2) = 1$ $\Phi = \{D_4\}$. The table of transitions of the Mealy FSM of the CMCU $U_7(\Gamma_2)$ has 5 lines (Table 15.5).

This table is a base to form the systems (15.15)–(15.16):

$$D_1 = \tau_1\bar{x}_1; D_2 = \tau_1 x_1 \vee \tau_1\bar{x}_1\bar{x}_2\bar{x}_3, D_3 = \tau_1\bar{x}_1\bar{x}_2; D_4 = \bar{\tau}_1. \tag{15.19}$$

The logic circuit of the CMCU $U_7(\Gamma_2)$ is shown in Fig. 15.17.

Here the LUT elements $LUT_1 - LUT_4$ implement the functions $D_1 - D_4$ represented as (15.18). The next two LUT elements implement the timing functions of the CT and the RG:

$$C_1 = y_0\, Clock;\ C_2 = \bar{y}_0\, Clock,$$

where $C_1 \# CT := CT + 1$; $C_2 \# CT := \langle D_1, D_2, D_3\rangle$; $C_2 \# RG := \langle D_4\rangle$. The control memory is implemented using two EMBs, because in our example each EMB has 4 outputs.

Table 15.5. Table of the transitions of the CMCU $U_7(\Gamma_2)$

a_m	$K(a_m)$	a_s	$K(a_s)$	X_h	Φ_h	Ψ_h	h
a_1	0	a_2	1	1	—	D_4	1
a_2	1	a_1	0	x_1	D_2	—	2
		a_1	0	$\bar{x}_1 x_2$	D_1	—	3
		a_1	0	$\bar{x}_1 \bar{x}_2 x_3$	$D_1 D_3$	—	4
		a_1	0	$\bar{x}_1 \bar{x}_2 \bar{x}_3$	$D_1 D_2 D_3$	—	5

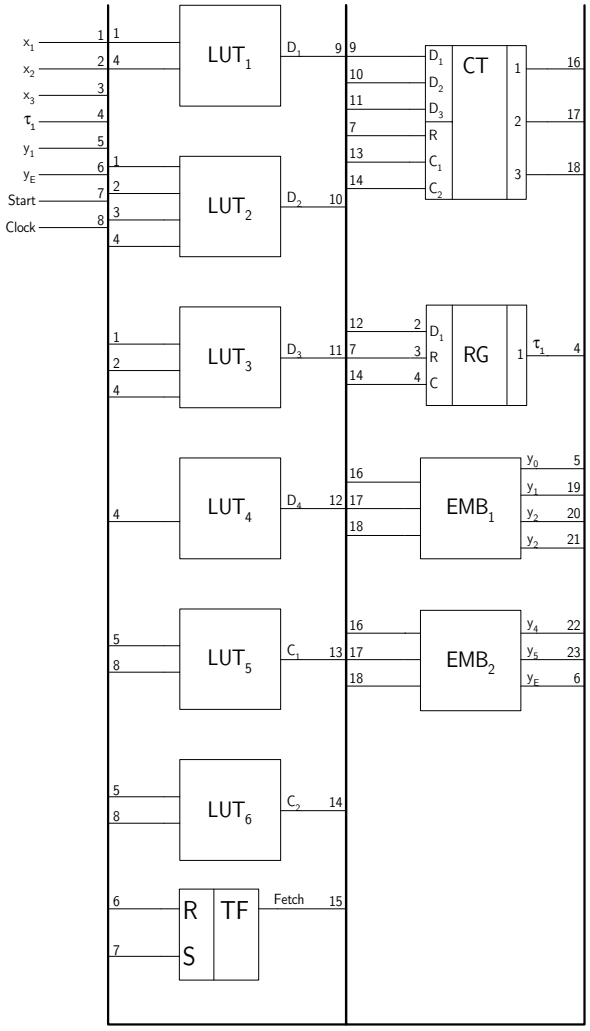


Fig. 15.17. Logic circuit of the CMCU $U_7(\Gamma_2)$

The comparison of the MCU with natural addressing of microinstructions and the CMCU with the base structure shows the following advantages of the latter approach:

1. The format of the microinstructions of the CMCU includes only the operational part. It means that the MIs of the CMCU U_7 are no longer than the MIs of the MCU U_6.
2. The number of MIs in the control memory of the CMCU is equal to the number of operational nodes of the interpreted FCA. It means that the length of the microprogram for the CMCU is the smallest among all possible microprograms for a given FCA.
3. The existence of the FSM of addressing leads to the execution of any multiways transition for one cycle of CMCU operation.

The main disadvantage of the CMCU U_7 is a bigger number of the outputs of the CC in comparison with the equivalent Moore FSM. This disadvantage can be eliminated by the application of the following approaches:

1. A CMCU with a common memory (Barkalov and Węgrzyn, 2006), where the CT is used as a source of both the address of the MI and the code of the FSM of the addressing state.
2. A CMCU with the sharing of the codes (Barkalov and Węgrzyn, 2006), where the address of the MI is represented as a concatenation of the the code of the OLC and the code of its component.

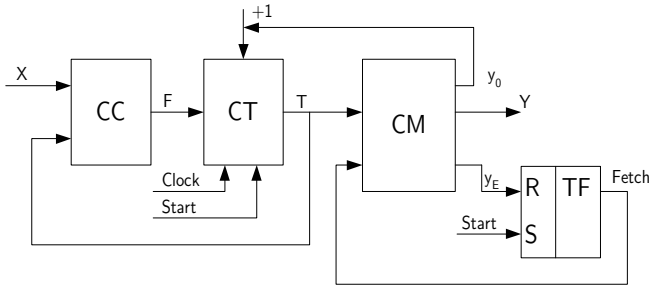


Fig. 15.18. Structural diagram of the CMCU U_8

The first approach leads to the CMCU U_8 (Fig. 15.18). It is clear that the CMCU U_8 is a Moore FSM with the CT instead of the RG. It means that the CMCU U_8 has all advantages and drawbacks of the usual Moore FSM.

Let $F_{\max} = \max(F_1, \ldots, F_G)$, $R_4 =]\log_2 G[$, $R_5 =]\log_2 F_{\max}[$, $|\tau| = R_4$, $T = |R_5|$, where R_4 is a bit capacity of the code of the OLC $\alpha_g \in C$, R_5 is a bit capacity of the code of its component. Let $K(\alpha_g)$ be the code of the OLC $\alpha_g \in C$ and $K(b_q)$ the code of its component $b_q \in \alpha_g$. In this case, the address $A(b_q)$ can be represented as

$$A(b_q) = K(\alpha_g) * K(b_q), \tag{15.20}$$

where $*$ is a sign of concatenation.

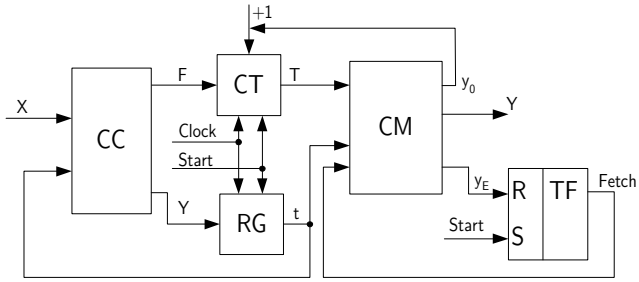


Fig. 15.19. Structural diagram of the CMCU U_9

This approach can be used if the condition

$$R_2 = R_4 + R_5 \tag{15.21}$$

holds and it leads to the CMCU U_9 (Fig. 15.9).

The choice between U_8 and U_9 can be executed on the basis of the analysis of the condition (15.21). Research in this area is continued and there are many methods of optimizing the hardware amount in the circuits of the CMCU U_7–U_9 (Barkalov and Węgrzyn, 2006; Barkalov and Wiśniewski, 2004a; 2004b; 2004c; 2004d; 2005; Barkalov *et al.*, 2005a; 2005b). The main problem in this area is a priori choice of the best structure of the CMCU that can minimize the amount of LUT-elements or cells of the CPLD in the final implementation of the project. Here "a priori" means that the choice should be made without designing all possible variants of a particular FCA implementation.

15.5. Conclusions

A control unit is one of the most important blocks of any digital system. The characteristics of the CU have a strong influence on the final effectiveness of the digital system. Modern control units are implemented using elements of programmable logic, such as the FPGA and the CPLD. One of the very important problems concerning digital units design is the optimization of the characteristics of the control unit. The methods of optimization can have three goals:

- minimization of the hardware amount;
- minimization of the cycle time;
- design of a circuit with minimal power consumption.

A control algorithm can be interpreted using different models of the control unit, such as the Mealy FSM, the Moore FSM, the MCU or the CMCU. Some of these models and their application have been discussed in this chapter. This material can be summarized in the following form:

1. The hardware amount and the performance of the control unit depend strongly on the characteristics of both the control algorithm and the elements of programmable logic that are used for its implementation.

2. There is the best model of the CU for a particular control algorithm and programmable logic devices in use.
3. The most important goal of research in the area of control units design oriented towards the Programmable Logic Device (PLD) is the search for the way for a priori estimation of the final project characteristics. This goal can be achieved if the characteristics of the CU can be estimated using some formulae or expert systems.

Therefore, research in this area should be continued.

References

Achasova S.N. (1987): *Algorithms of Synthesis of Automata on Programmable Arrays.* — Moscow: Radio i Swiaz, (in Russian).

Adamski M. and Barkalov A. (2006): *Architectural and Sequential Synthesis of Digital Devices.* — University of Zielona Góra Press, (in Polish).

Baranov S.I. (1994): *Logic Synthesis of Control Automata.* — Boston: Kluwer Academic Publishers.

Barkalov A.A. (1998): *Principles of optimization of logic circuit of Moore FSM.* — Cybernetics and System Analysis, No. 1, pp. 65–72, (in Russian).

Barkalov A.A. (2002): *Synthesis of Control Units on Programmable Logic Devices.* — Donetsk: Donetsk National Technical University Press, (in Russian).

Barkalov A.A. (2003): *Synthesis of Operational Units.* — Donetsk: Donetsk National Technical University Press, (in Russian).

Barkalov A.A. and Palagin A.V. (1997): *Synthesis of Microprogram Control Units.* — Kiev: Institute of Cybernetics of NAS of Ukraine Press, (in Russian).

Barkalov A., Titarenko L. and Wiśniewski R. (2005a): *Optimization of the amount of LUT–elements in compositional microprogram control unit with mutual memory.* — Proc. IEEE *East-West Design & Test Workshop, EWDTW'05*, Odessa, Ukraine, pp. 75–79.

Barkalov A., Wiśniewski R. and Titarenko L. (2005b): *Synthesis of comositional microprogram control unit on FPGA.* — Proc. 12-th Int. Conf. *Mixed Design of Integrated Circuits and Systems, MIXDES 2005*, Cracow, Poland, Vol. 1, pp. 205–208.

Barkalov A. and Węgrzyn M. (2006): *Design of Control Units with Programmable Logic.* — University of Zielona Góra Press, (in Polish).

Barkalov A. and Wiśniewski R. (2004a): *Design of compositional microprogram control units with transformation of the number of transactions.* — Proc. 11-th Int. Conf. *Mixed Design of Integrated Circuits and Systems, MIXDES 2004*, Szczecin, Poland, pp. 172–175.

Barkalov A. and Wiśniewski R. (2004b): *Optimization of compositional microprogram control unit with elementary operational linear chains.* — Upravljuscije Sistemy i Masiny, No. 5, pp. 25–29, (in Russian).

Barkalov A. and Wiśniewski R. (2004c): *Optimization of compositional microprogram control units with sharing of codes.* — Proc. Int. Conf. *Avtomatizacija Proektirovanija Diskretnych Sistem*, Minsk, Belarus, Vol. 1, pp. 16–22.

Barkalov A. and Wiśniewski R. (2004d): *Synthesis of compositional microprogram control units with transformation of the numbers of inputs.* — Proc. Int. Workshop *Discrete-Event System Design, DESDes'04*, Dychów, Poland, pp. 145–148.

Barkalov A. and Wiśniewski R. (2005): *Implementation of compositional microprogram control unit on FPGAs.* — Proc. IEEE *East-West Design & Test Workshop, EWDTW'05*, Odessa, Ukraine, pp. 80–83.

Clements A. 2000: *The Principles of Computer Hardware.* — New York: Oxford University Press.

Grushnitski R.I., Mursaev A.H. and Ugrjumov E.P. (2002): *Design of the Systems Using the Microchips of Programmable Logic.* — Petersburg: BHV, (in Russian).

Jenkins J. (1994): *Designing with FPGAs and CPLDs.* — New York: Prentice Hall.

Kania D. (2004): *Logic synthesis with programmable array logic.* — Zeszyty Naukowe Politechniki Śląskiej, Gliwice, pp. 240, (in Polish).

Łuba T. (Ed.) (2003): *Synthesis of Digital Circuits.* — Warsaw: Wydawnictwo Komunikacji i Łączności, (in Polish).

Łuba T., Jasiński K. and Zbierzchowski B. (1997): *Specialized Digital Devices Using PLD and FPGA.* — Warsaw: Wydawnictwo Komunikacji i Łączności, (in Polish).

Maxfield C. 2004: *The Design Warrior's Guide to FPGAs.* — Orlando: Academic Press.

Salisbury A. (1976): *Microprogrammable Computer Architectures.* — New York: Am Elstein.

Solovjev V.V. 2001: *Design of Digital Systems Using the Programmable Logic Integrated Circuits.* — Moscow: Hot Line-Telecom, (in Russian).

Wilkes M.V. (1951): *The best way to design an automatic calculating machine.* — Rep. Manchester University Computer Inaugural Conf., UK, pp. 16–18.

Zwoliński M. 2002: *Digital Circuit Design Using the VHDL Language.* — Warsaw: Wydawnictwo Komunikacji i Łączności, (in Polish).

Chapter 16

DIRECT PWM AC CHOPPERS AND FREQUENCY CONVERTERS

Zbigniew FEDYCZAK*, Paweł SZCZEŚNIAK*, Jacek KANIEWSKI*

16.1. Introduction

Direct Pulse Width Modulation (PWM) AC converters can be divided into two basic groups. The first one comprises AC Choppers (ACCs), also called controllers or conditioners, and fulfills the function of the semiconductor AC/AC voltage and current transforming circuit similarly to the conventional transformer with electromagnetic coupling. In a circuit with an ACC the fundamental harmonic frequency of the output voltage is the same as the supplying voltage frequency. The second group comprises AC Frequency Converters (ACFCs) that have the possibility to control both the amplitude and frequency of the output voltage fundamental harmonic.

Among ACCs, low-cost Thyristor Choppers (TCs) are commonly used in industrial practice (among other things for temperature control, lighting control, power control and voltage stabilization in "soft" supply networks). Their major disadvantages are the generation of higher harmonics in the source current, the generation of displacement power in phase angle control, and the generation of subharmonics in integral control (Clark, 1990; Fedyczak and Strzelecki, 1997). In order to eliminate these unfavourable properties it is proposed to use Matrix CHoppers (MCHs) and Matrix-Reactance Choppers (MRCs) with transistor switches working with the switching frequency $f_S \gg f$, where f is the supply voltage frequency. Since 1994 the Institute of Electrical Engineering has carried out extensive research on MCH and MRC (Fedyczak, 2001; 2003a; 2003b; 2003c; Fedyczak and Korotyeyev, 2003; Fedyczak and Strzelecki, 1994; 1997; 1998; Fedyczak *et al.*, 1999; 2000; 2001a; 2001b; 2002a; 2002b; 2006; Korotyeyev *et al.*, 2001; Korotyeyev and Fedyczak, 2005 Strzelecki and Fedyczak, 1995a; 1995b; 1996a; 1996b; 1996c; Strzelecki *et al.*, 1996a; 1996b; 1997a; 1997b; Tunia *et al.*, 1998).

* Institute of Electrical Engineering
e-mails: {Z.Fedyczak, P.Szczesniak, J.Kaniewski}@iee.uz.zgora.pl

One of the most desirable features of AC Frequency Converters (ACFCs) is the generation of a load voltage with an arbitrary amplitude and frequency (Apap *et al.*, 2003; Wheeler *et al.*, 2002). Among ACFCs Indirect Frequency Converters (IFCs) are commonly used in industrial practice especially for drive control and in Flexible Alternating Transmission Systems (FACTSs). In recent years, Matrix Converters (MCs), belonging to Indirect Frequency Converters (IFC), have received considerable attention as a competitor to the normally used PulseWidth-Modulated-Voltage Source Inverters (PWM-VSI). The real development of MCs starts with the work of (Venturini and Alesina, 1980). As is well known, the MC, compared to the PWM-VSI with the diode rectification stage at the input, provides sinusoidal input and output waveforms, bidirectional power flow, controllable input power factor, and more compact design (Apap *et al.*, 2003; Wheeler *et al.*, 2002). Unfortunately, the MC load output voltage is limited to 0.5 of the input voltage (in linear voltage transformation) and to 0.866 or 1.053 (in low-frequency load voltage deformations for space-vector or fictitious DC link control strategy concepts, respectively) (Apap *et al.*, 2003; Casadei *et al.*, 2002; Helle *et al.*, 2004; Venturini and Alesina, 1980; Wheeler *et al.*, 2002). In the last few years at the Institute of Electrical Engineering a new concept of Matrix-Reactance Frequency Converters (MRFCs) has been taken up (Fedyczak *et al.*, 2006; Fedyczak and Szcześniak, 2005; 2006a; 2006b), as development of the conception presented in (Fedyczak, 2003a; Zinoviev *et al.*, 2000). The topologies of these MRFCs are based on a matrix-reactance chopper with source or load switches arranged as in an MC. Such an approach gives the possibility to obtain a load output voltage greater than the input one.

The main aim of this chapter is the presentation of selected long-term research results dealing with direct PWM AC choppers and frequency converters that have been achieved at the Institute of Electrical Engineering. For PWM AC line MCHs and MRCs, as well as for MRFC general description, modeling methods and exemplary simulation and experimental test results are presented. Furthermore, conclusions and a program of further investigations follow in the final section.

16.2. PWM AC line choppers

16.2.1. General description

The basic topologies of a unipolar PWM AC line MCH are shown in Fig. 16.1. The single-phase topology is based on a singular converter (Fig. 16.1(a)), whereas the three-phase one is based on a simplified matrix converter in which only synchronous switch configurations are employed (Fig. 16.1(b)). Furthermore, in PWM AC voltage transforming circuits the bipolar PWM AC line MCH and nonsymmetrical three-phase ones are also used (Fedyczak and Strzelecki, 1997). The basic simplified topologies of such MCHs are shown in Fig. 16.2.

Exemplary time waveforms illustrating the operation of a single-phase MCH are shown in Fig. 16.3, whereas detailed operational descriptions of the whole family of single- and three-phase MCH solutions are presented in (Fedyczak, 2003a; Fedyczak and Strzelecki, 1997).

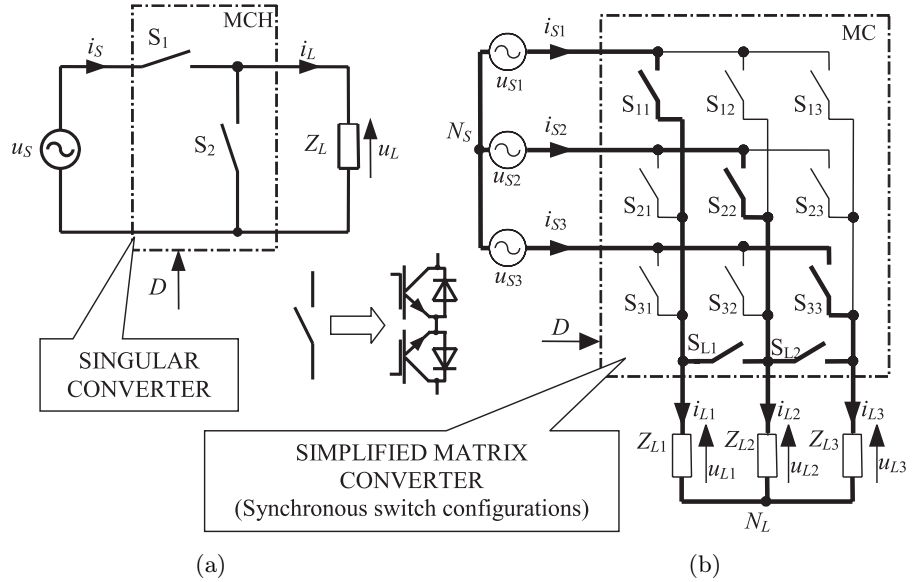


Fig. 16.1. Basic topologies of a unipolar PWM AC line MCH, (a) single-phase, (b) three-phase

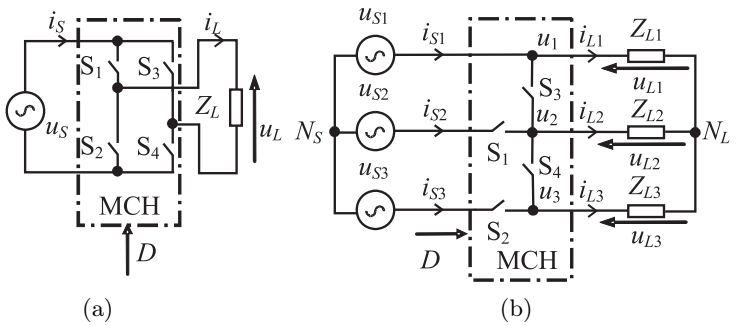


Fig. 16.2. Basic topologies of other PWM AC line MCHs, (a) single-phase bipolar one, (b) three-phase nonsymmetrical one

Single-phase topologies of PWM AC line MRCs are based on the well-known DC/DC converter structures. Their topologies are built up by means of the adaptation of respective switches as shown in Fig. 16.4 (Fedyczak, 2003a; Kim *et al.*, 1998).

Exemplary time waveforms illustrating the operation of a single-phase MRC with a buck-boost topology are shown in Fig. 16.5, whereas a detailed operational description of the whole family of single- and three-phase MRC solutions, whose simplified topologies are collected in Figs. 16.6–16.8, is presented in (Fedyczak, 2003a). It should be noted that three-phase topologies are constructed trough suitable matching of single-phase structures.

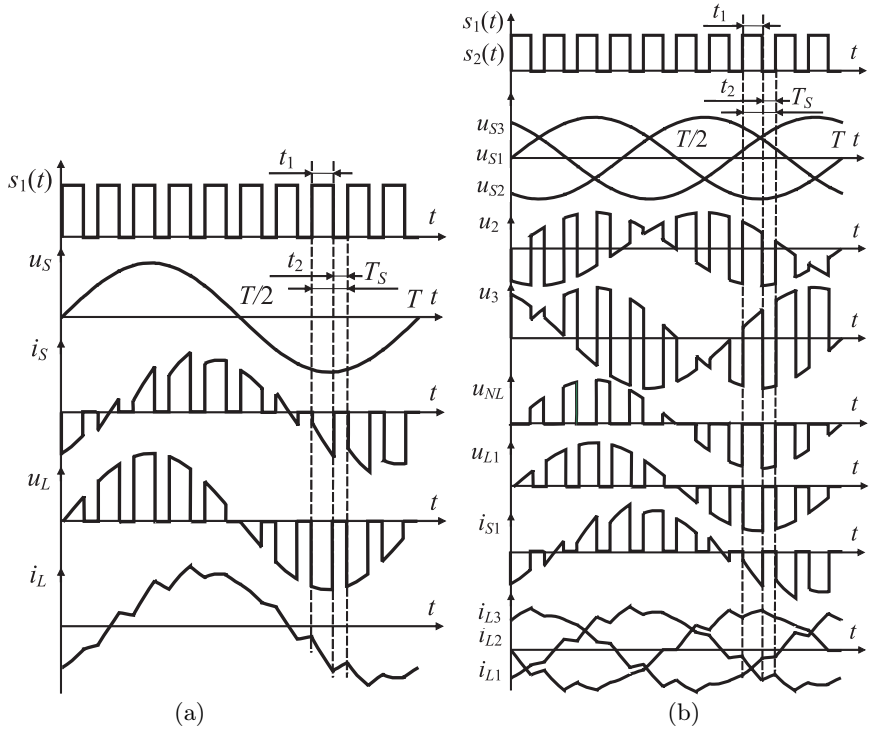


Fig. 16.3. Exemplary time waveforms in circuits with a unipolar PWM AC line MCH for an inductive load and the switching frequency $f_S = 0.5\,\text{kHz}$ at the pulse duty factor $D = 0.6$, (a) MCH in Fig. 16.1(a), (b) MCH in Fig. 16.2(b)

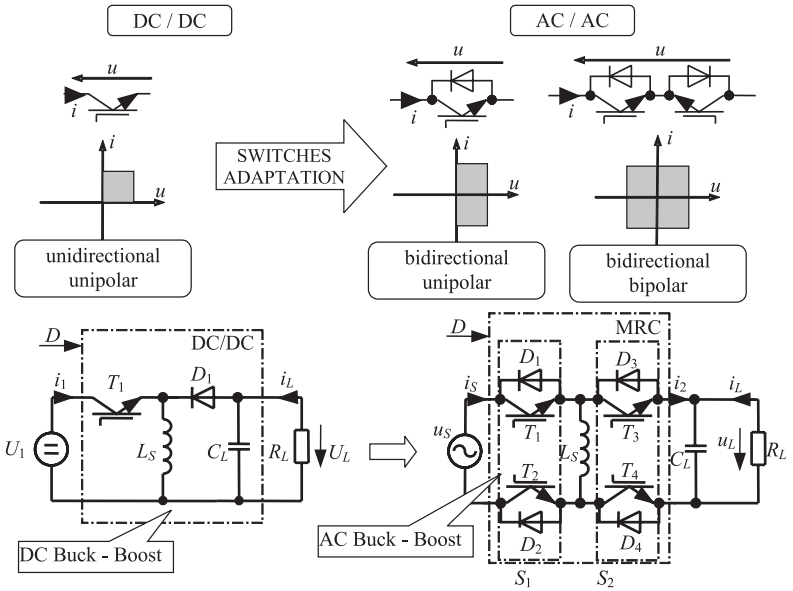


Fig. 16.4. Single-phase PWM AC line MRC with the buck-boost topology built up from a DC/DC converter structure by means of a respective switch adaptation

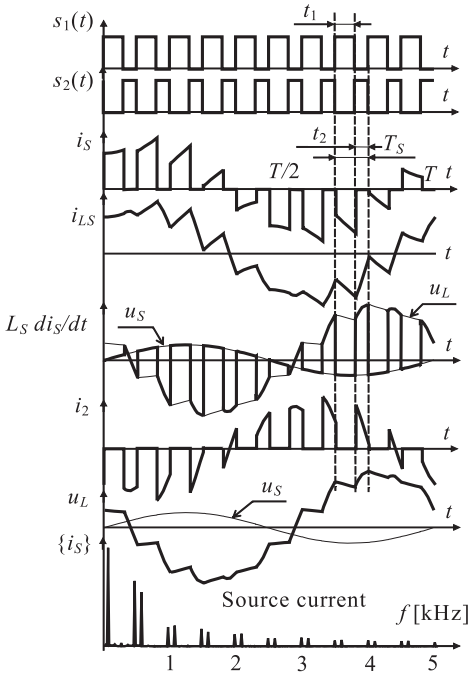


Fig. 16.5. Exemplary time waveforms and the source current spectrum in a circuit with an MRC with the buck-boost topology (Fig. 16.5) for the switching frequency $f_S = 0.5\,\mathrm{kHz}$ at the pulse duty factor $D = 0.6$

Buck	Boost	Buck-Boost	Ćuk
$H_U(D) = D$	$H_U(D) \approx \frac{1}{(1-D)}$	$H_U(D) \approx \frac{-D}{(1-D)}$	$H_U(D) \approx \frac{-D}{(1-D)}$
Zeta	Sepic	Ćuk B1	Ćuk B 2
$H_U(D) \approx \frac{D}{(1-D)}$	$H_U(D) \approx \frac{D}{(1-D)}$	$H_U(D) \approx \frac{(1-2D)}{(1-D)}$	$H_U(D) \approx \frac{(1-D)}{(1-2D)}$

Fig. 16.6. Basic and symmetrical topologies of a single-phase PWM AC line MRC, H_U – voltage transfer function, D – pulse duty factor

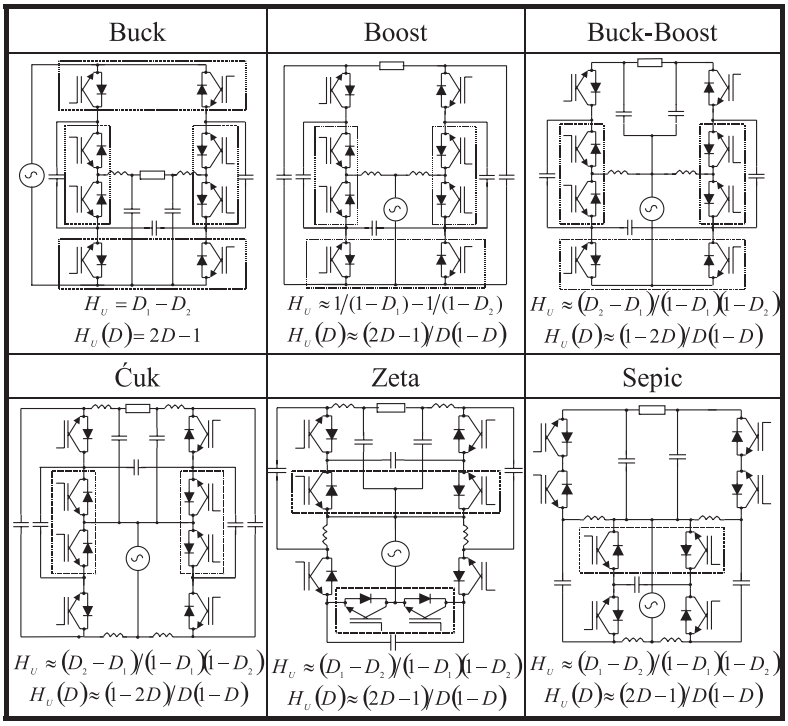


Fig. 16.7. Basic topologies of a single-phase bipolar PWM AC line MRC, H_U – voltage transfer function, D – pulse duty factor ($H_U(D)$ for the case when $D = D_1 = 1 - D_2$)

The circuit realisations of selected single- and three phase PWM AC line choppers are shown in Figs. 16.9, 16.11, 16.13, 16.15 and 16.17, whereas exemplary experimental test results are shown in Figs. 16.10, 16.12, 16.14, 16.16 and 16.18 (Fedyczak, 2003a).

16.2.2. Modelling

The discussed PWM AC line choppers are periodically nonstationary circuits with regard to impulse changing of the switch parameters used in these choppers. In steady state analysis of the presented choppers two methods have been used. The first one is based on fundamental harmonics of the switch state function. A block scheme describing this method is shown in Fig. 16.19 (Fedyczak and Strzelecki, 1997). The second method is based on the averaged state space method introduced by (Middlebrock and Ćuk, 1976) with a running averaging operator according to (Korotyeyev *et al.*, 2001). A block schema describing this method is shown in Fig. 16.20 (Fedyczak, 2003a; Korotyeyev and Fedyczak, 2005). It should be noted that for finite switching frequency T_S, the averaged solution of the state variables is distorted by errors. The obtained results of the averaged model's accuracy analysis confirm that both amplitude and phase averaging errors of the state variables decrease along with a switching frequency increase. Furthermore, some practicable quantitative results have been obtained on

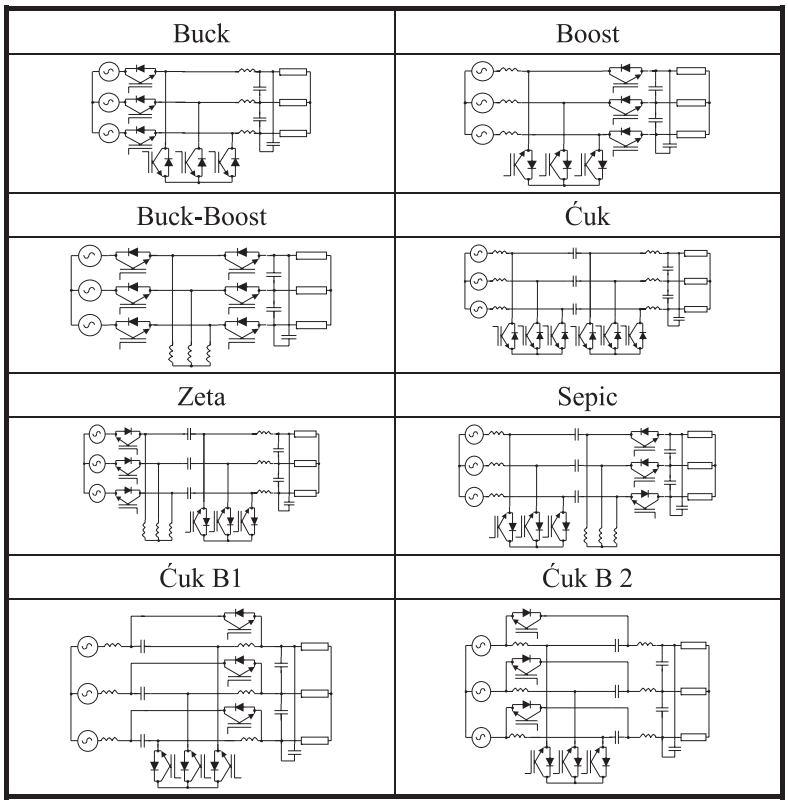


Fig. 16.8. Basic symmetrical topologies of three-phase PWM AC line MRC

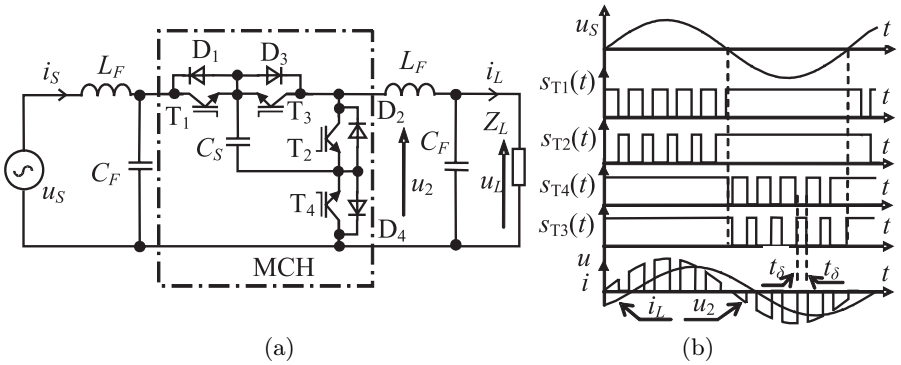


Fig. 16.9. Single-phase unipolar PWM AC line MCH, (a) schematic diagram, (b) exemplary voltage and current time waveforms; t_δ – "dead time"

the basis of this analysis. For 5 kHz switching frequency, i.e. about $3/(2\pi\sqrt{LC})$, amplitude errors are smaller than approximately 20%, whereas phase errors are smaller than 0.1 rad. The greatest amplitude and phase averaging errors of the state variables at the fixed switching frequency occur if $0.01 < R_L/\sqrt{L/C} < 1$ (Fedyczak, 2003a; Korotyeyev and Fedyczak, 2005; Korotyeyev *et al.*, 2001).

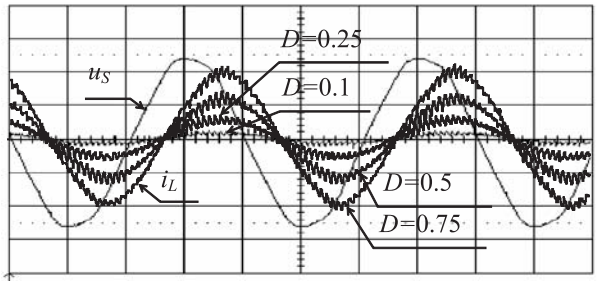
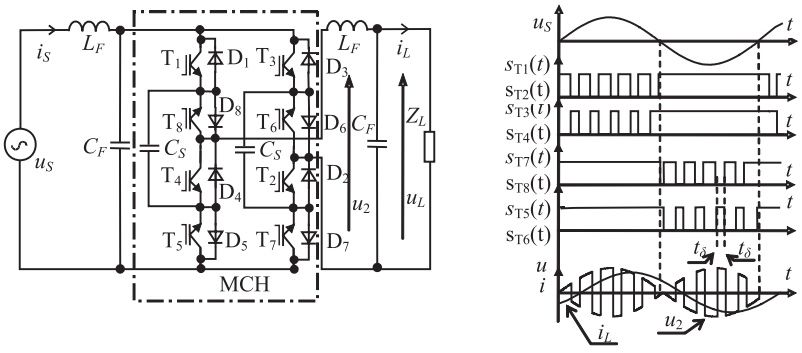
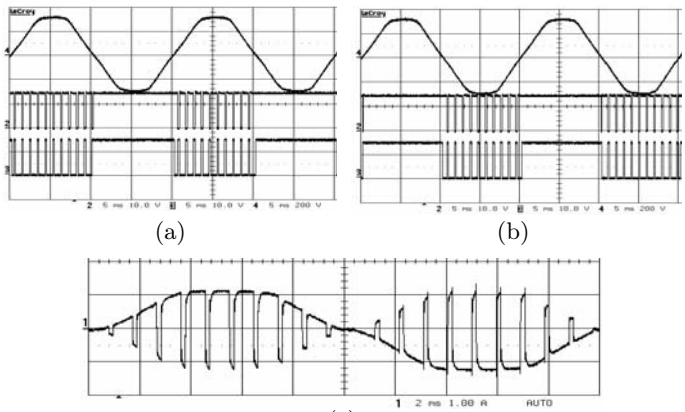


Fig. 16.10. Experimental voltage and current time waveforms in circuit shown in Fig. 16.9 for $f_S = 1\,\mathrm{kHz}$



(a) (b)

Fig. 16.11. Single-phase bipolar PWM AC line MCH, (a) schematic diagram, (b) exemplary voltage and current time waveforms; t_δ – "dead time"



(a) (b)

(c)

Fig. 16.12. Experimental voltage and current time waveforms in the circuit shown in Fig. 16.11 for $f_S = 1\,\mathrm{kHz}$, (a) supplying voltage and control signals of the transistors T1–T4, (b) supplying voltage and control signals of the transistors T5–T8, (c) output voltage u_2

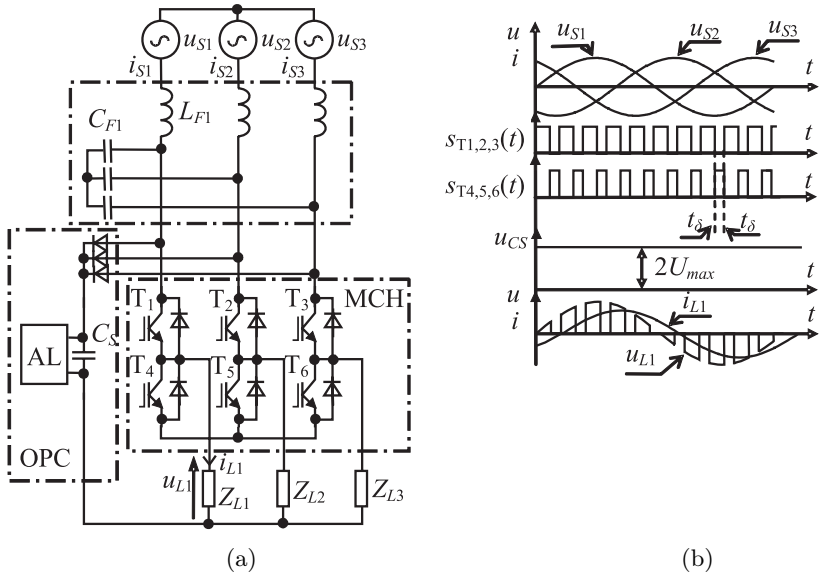


Fig. 16.13. Three-phase unipolar PWM AC line MCH, (a) schematic diagram, (b) exemplary voltage and current time waveforms; OPC – overvoltage protection circuit, AL – active load, t_δ – "dead time"

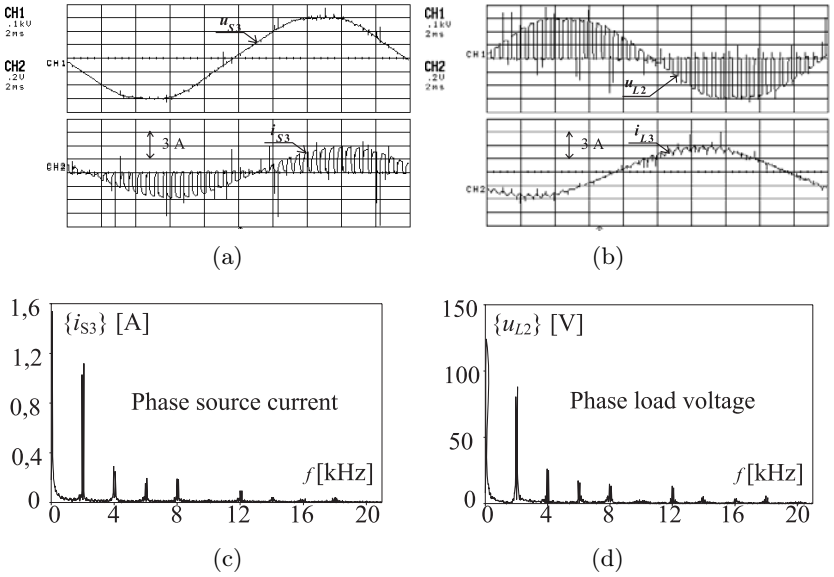
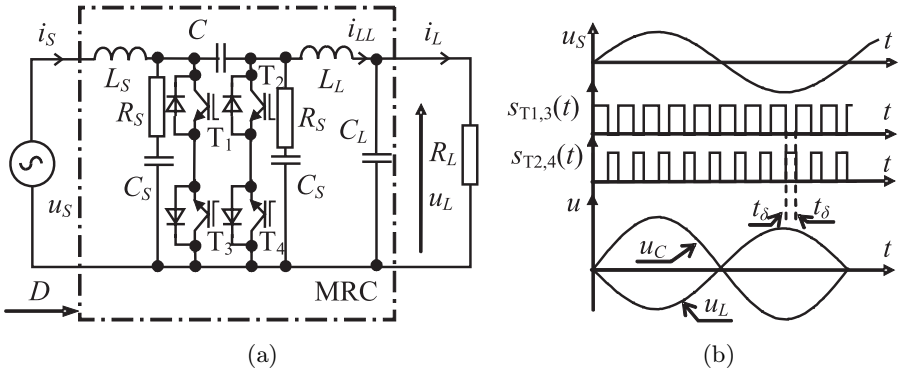
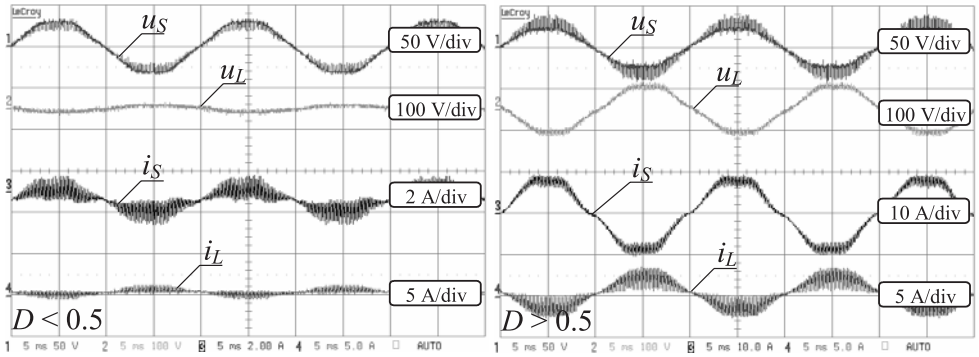


Fig. 16.14. Experimental voltage and current time waveforms and their spectra in the circuit shown in Fig. 16.13 without an input filter for $f_S = 2\,\text{kHz}$, $D = 0.6$, $I_{L\,\max} = 4.5\,\text{A}$ ($\cos\varphi_L = 0.85$), (a) source phase voltage and current, (b) load phase voltage and current (c), (d) phase source current and phase load voltage spectra



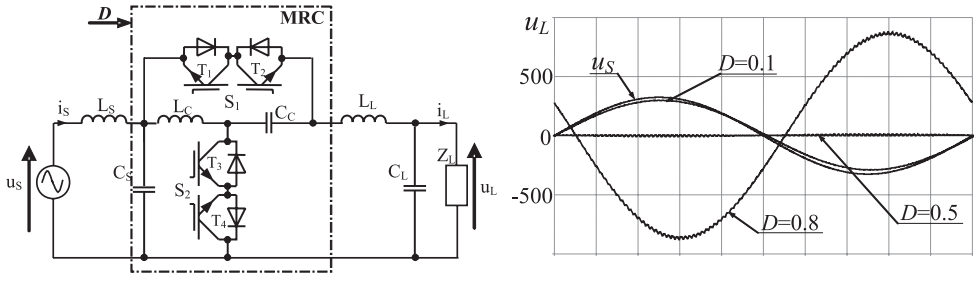
(a) (b)

Fig. 16.15. Single-phase unipolar PWM AC line MRC with the Ćuk topology, (a) schematic diagram, (b) exemplary voltage and current time waveforms; t_δ – "dead time"



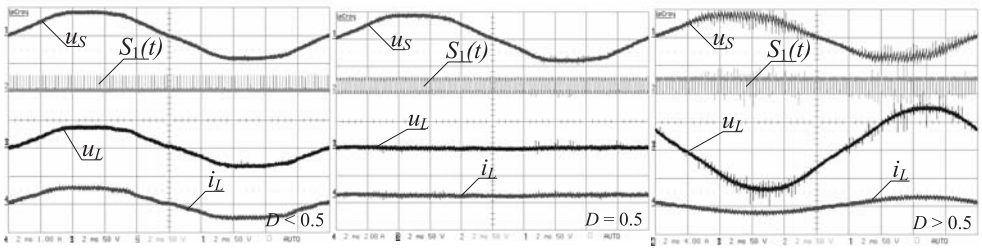
(a) (b)

Fig. 16.16. Experimental voltage and current time waveforms in circuit shown in Fig. 16.15 for $f_S = 5\,\text{kHz}$, (a) for $D < 0.5$, (b) for $D > 0.5$



(a) (b)

Fig. 16.17. Single-phase bipolar PWM AC line MRC with the Ćuk B2 topology, (a) schematic diagram, (b) exemplary voltage time waveforms



(a) (b) (c)

Fig. 16.18. Experimental voltage and current time waveforms in the circuit shown in Fig. 16.17 for $f_S = 5\,\text{kHz}$, (a) for $D < 0.5$, (b) for $D = 0.5$, (c) for $D > 0.5$

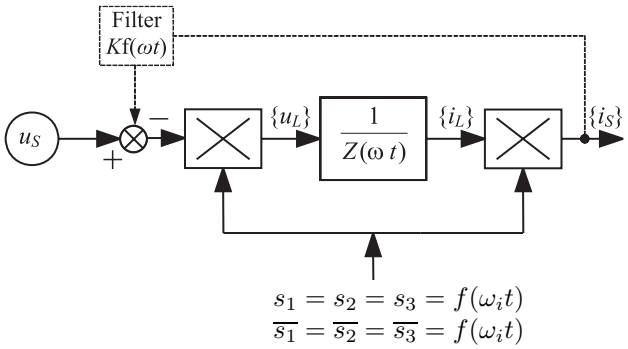


Fig. 16.19. Block schema of the method of fundamental harmonics of the switch state function

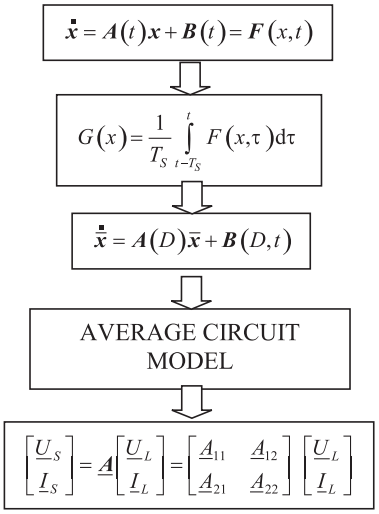


Fig. 16.20. Schema of the averaged state space method with a running averaging operator

For a single-phase PWM AC line MRC with the Ćuk topology, the circuit switched models are shown in Fig. 16.21. The state-space equations for the circuit shown in Figs. 16.21(b) and 16.21(c) are described by (16.5) and (16.6). On the basis of the averaged state space method, for a sufficiently small switching period T_S we obtain the equations (16.7) (Fedyczak, 2001). The averaged state space model (16.7) is linear and valid for DC and AC (fundamental harmonic) regimes. From (16.7) one can easily obtain the equivalent circuit realisation shown in Fig. 16.22.

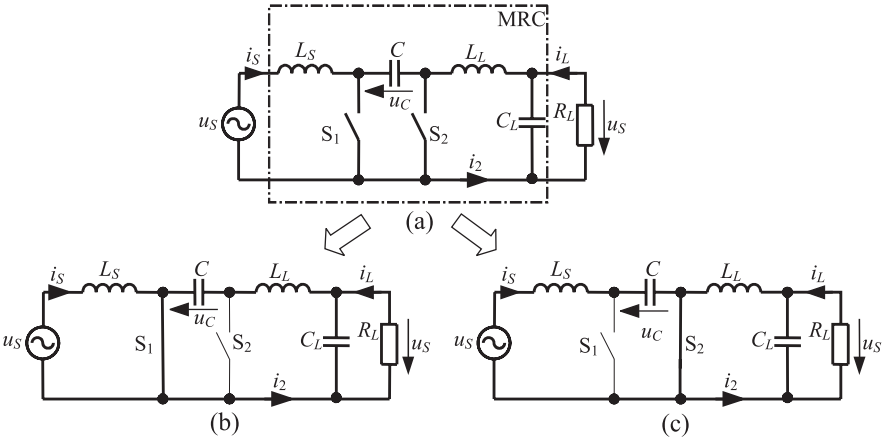


Fig. 16.21. Circuit switched models of a PWM AC line MRC with the Ćuk topology, (a) basic switched model, (b) switched model for S_1 on and S_2 off, (c) switched model for S_1 off and S_2 on

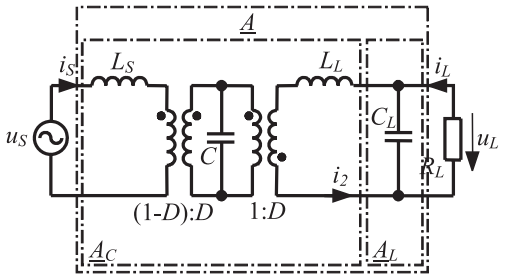


Fig. 16.22. Averaged model of a PWM AC line MRC with the Ćuk topology; $\underline{A}_C$, $\underline{A}_L$, $\underline{A}$ – four terminal chain parameters of the basic structure, a load circuit and a complete MRC

$$\begin{bmatrix} L\dfrac{\mathrm{d}i_S}{\mathrm{d}t} \\ C\dfrac{\mathrm{d}u_C}{\mathrm{d}t} \\ L_L\dfrac{\mathrm{d}i_2}{\mathrm{d}t} \\ C_C\dfrac{\mathrm{d}u_L}{\mathrm{d}t} \end{bmatrix} = \overbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & \dfrac{-1}{R_L} \end{bmatrix}}^{A_1} \begin{bmatrix} i_S \\ u_C \\ i_2 \\ u_L \end{bmatrix} + \overbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{B_2} u_S, \tag{16.5}$$

$$\begin{bmatrix} L_S\dfrac{\mathrm{d}i_S}{\mathrm{d}t} \\ C\dfrac{\mathrm{d}u_C}{\mathrm{d}t} \\ L_L\dfrac{\mathrm{d}i_2}{\mathrm{d}t} \\ C_L\dfrac{\mathrm{d}u_L}{\mathrm{d}t} \end{bmatrix} = \overbrace{\begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & 1 & \dfrac{-1}{R_L} \end{bmatrix}}^{A_2} \begin{bmatrix} i_S \\ u_C \\ i_2 \\ u_L \end{bmatrix} + \overbrace{\begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{B_2} u_S. \tag{16.6}$$

$$\begin{bmatrix} L_S\dfrac{\mathrm{d}\bar{i}_S}{\mathrm{d}t} \\ C\dfrac{\mathrm{d}\bar{u}_C}{\mathrm{d}t} \\ L_L\dfrac{\mathrm{d}\bar{i}_2}{\mathrm{d}t} \\ C_L\dfrac{\mathrm{d}\bar{u}_L}{\mathrm{d}t} \end{bmatrix} \approx \left(\overbrace{\begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & -D & 0 \\ 0 & D & 0 & -D \\ 0 & 0 & D & \dfrac{-D}{R_L} \end{bmatrix}}^{DA_1} + \overbrace{\begin{bmatrix} 0 & -(1-D) & 0 & 0 \\ (1-D) & 0 & 0 & 0 \\ 0 & 0 & 0 & -(1-D) \\ 0 & 0 & 1 & \dfrac{-(1-D)}{R_L} \end{bmatrix}}^{(1-D)A_2} \right) \begin{bmatrix} \bar{i}_S \\ \bar{u}_C \\ \bar{i}_2 \\ \bar{u}_L \end{bmatrix}$$

$$+ \left(\overbrace{\begin{bmatrix} D \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{DB_1} + \overbrace{\begin{bmatrix} (1-D) \\ 0 \\ 0 \\ 0 \end{bmatrix}}^{(1-D)B_2} \right) u_S$$

$$= \begin{bmatrix} 0 & -(1-D) & 0 & 0 \\ (1-D) & 0 & -D & 0 \\ 0 & D & 0 & -1 \\ 0 & 0 & 1 & \dfrac{-1}{R_L} \end{bmatrix} \begin{bmatrix} \bar{i}_S \\ \bar{u}_C \\ \bar{i}_2 \\ \bar{u}_L \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix} u_S, \tag{16.7}$$

where $\boldsymbol{A}(D) = \boldsymbol{A}_1 D + \boldsymbol{A}_2(1-D)$, $\boldsymbol{B}(D) = \boldsymbol{B}_1(D) + \boldsymbol{B}_2(1-D)$.

The four terminal description of the presented MRC is a natural and useful approach to one property analysis. A two-port shot is convenient during parameter synthesis and features comparison of the presented choppers. The steady-state averaged models shown in Fig. 16.22 can be readily derived in terms of the A four terminal chain parameters. Such a description has been chosen arbitrarily although a simpler form of transmittances in the case of the A parameters is preferred. Furthermore, this choice follows with a view to doing a comprehensive description of the whole group of unipolar and bipolar MRCs in spite of the translation of G and H parameters into A ones in the case of the presented choppers. The $\underline{A}_C$ and $\underline{A}$ parameters are collected in Table 16.1. The circuit averaged models and chain parameters of the whole family of the presented choppers are given in (Fedyczak, 2003a).

Table 16.1. Four terminal chain parameters of the averaged model of a PWM AC line MRC with the Ćuk topology

Parameter	Formula
$\underline{A}_{C11} = \left(\frac{\underline{U}_S}{\underline{U}_2}\right)_{\underline{I}_2=0}$	$\frac{(1-D)^2 - \omega^2 L_S C}{D(1-D)}$
$\underline{A}_{C12} = \left(\frac{\underline{U}_S}{\underline{I}_2}\right)_{\underline{U}_2=0}$	$\frac{j\omega\left[(1-D)^2 L_L - L_S\left(\omega^2 L_L C - D^2\right)\right]}{D(1-D)}$
$\underline{A}_{C21} = \left(\frac{\underline{I}_S}{\underline{U}_2}\right)_{\underline{I}_2=0}$	$\frac{j\omega C}{D(1-D)}$
$\underline{A}_{C22} = \left(\frac{\underline{I}_1}{\underline{I}_2}\right)_{\underline{U}_2=0}$	$-\frac{\omega^2 L_L C - D^2}{D(1-D)}$
$\underline{A}_{11} = \left(\frac{\underline{U}_S}{\underline{U}_L}\right)_{\underline{I}_L=0}$	$\underline{A}_{C11} + j\underline{A}_{C12}\omega C_L$
$\underline{A}_{12} = \left(\frac{\underline{U}_S}{\underline{I}_L}\right)_{\underline{U}_L=0}$	$\underline{A}_{C12}$
$\underline{A}_{21} = \left(\frac{\underline{I}_S}{\underline{U}_L}\right)_{\underline{I}_L=0}$	$\underline{A}_{C21} + j\underline{A}_{C22}\omega C_L$
$\underline{A}_{22} = \left(\frac{\underline{I}_S}{\underline{I}_L}\right)_{\underline{U}_L=0}$	$\underline{A}_{C22}$

16.2.3. Selected simulation and experimental test results

Exemplary experimental test results in the form of static characteristics of the basic dependences in a three-phase unipolar PWM AC line MCH (Fig. 16.13) are shown in Figs. 16.23 and 16.24. As is visible in Fig. 16.24, in a circuit with an MC and a

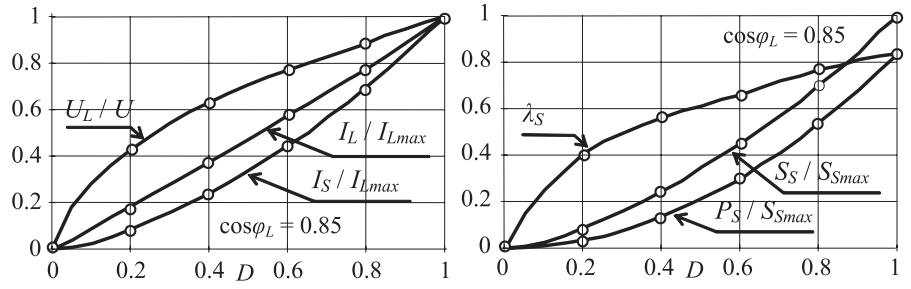


Fig. 16.23. Experimental relative dependencies in a three-phase unipolar PWM AC line MCH (Fig. 16.13), (a) RMS values of the load phase voltage and the current and also the phase source current, (b) source active and apparent power and also the input power factor

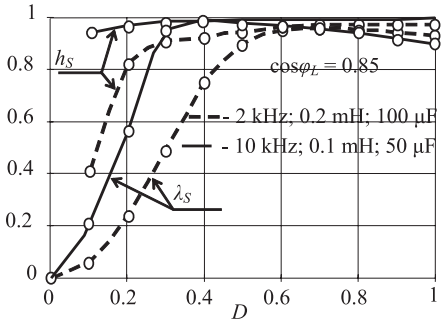


Fig. 16.24. Input power factor λ_S and the deformation factor of the phase source current in a three-phase PWM AC line MCH (Fig. 16.13) for a 25 kVA load with $\cos\varphi_L = 0.85$

necessary input LC filter the "cost" of higher harmonics filtration are favourable decreases at increasing switching frequency. Furthermore, the increasing of the switching frequency leads to a greater range of the control signal without a deterioration of the input power factor.

The analytical and simulation test results of voltage transmittance for the basic topologies of single-phase unipolar and bipolar PWM AC line MRCs, in a matching load condition, are shown in Figs. 16.25–16.28. These characteristics, for the comparative purpose, are shown along with simulation test results obtained for switching models of MRCs. As is visible in Figs. 16.25–16.28, the essential property of the presented PWM AC line MRC is the capability to increase the AC load voltage above the AC line supplying voltage without an electromagnetic transformer.

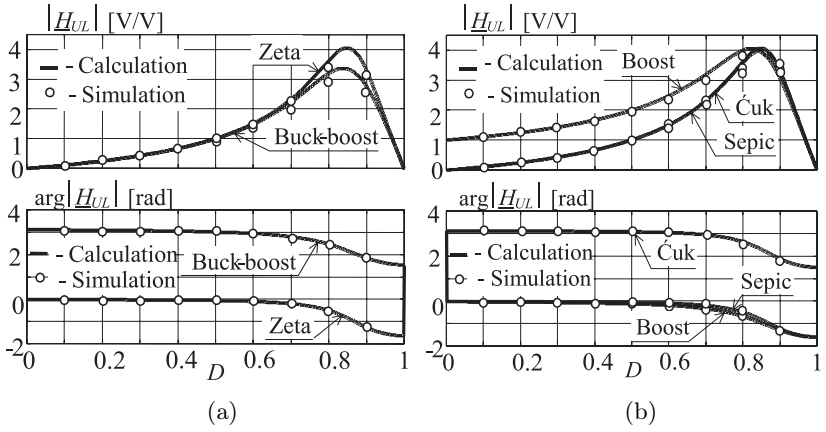


Fig. 16.25. Magnitude and phase of voltage transmittances for the basic topologies of a single-phase unipolar PWM AC line MRC in a matching load condition, (a) for the buck family, (b) for the boost family

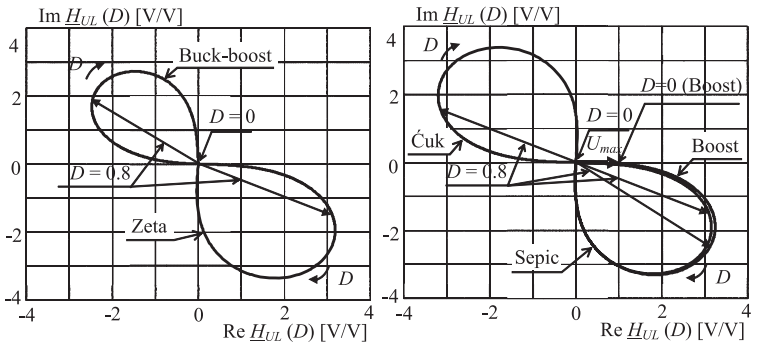


Fig. 16.26. Magnitude-phase characteristic of voltage transmittances for the basic topologies of a single-phase unipolar PWM AC line MRC in a matching load condition, (a) for the buck family, (b) for the boost family

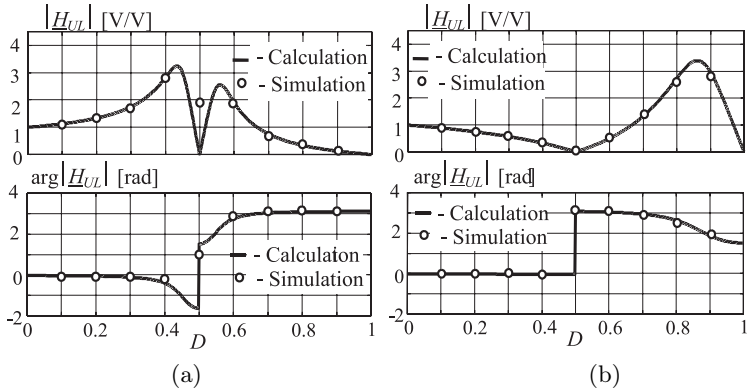


Fig. 16.27. Magnitude and phase of voltage transmittances for the basic topologies of a single-phase bipolar PWM AC line MRC in a matching load condition, (a) for the Ćuk B1 topology, (b) for the Ćuk B2 topology

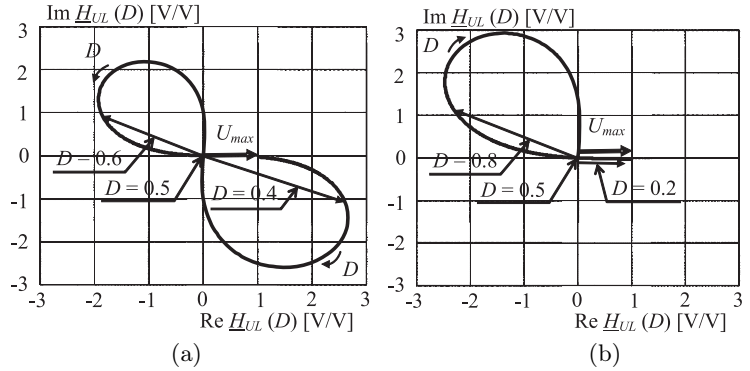


Fig. 16.28. Magnitude-phase characteristic of voltage transmittances for basic topologies of a single-phase bipolar PWM AC line MRC in a matching load condition, (a), (b) for the Ćuk B1 and the Ćuk B2 topology, respectively

16.3. Matrix-reactance frequency converters

16.3.1. General description

The topologies of the MRFC are based on a three-phase MRC structure. This structure also contains a separate switching set with a configuration as in MCs, which is illustrated in Fig. 16.29. Schematic diagrams of the new MRFCs, which have been elaborated by the authors, are shown in Figs. 16.30–16.32 (Fedyczak *et al.*, 2006;Fedyczak and Szcześniak, 2005; 2006a; 2006b). In the circuit in Fig. 16.30 there is used a three-phase buck-boost MRC with supply source switches arranged as in an MC. Furthermore, similarly as in an MC, in this circuit the input low-pass LC filter is used. Similarly, in the circuit from Fig. 16.31 there is used a three-phase Zeta MRC with supply source switches arranged as in MC. In the circuit in Fig. 16.32 there is used a three-phase Ćuk MRC with load switches arranged as in an MC. A description of the control strategy of the MRFC, in a general form, and a simplified realization are also shown in Figs. 16.30–16.32.

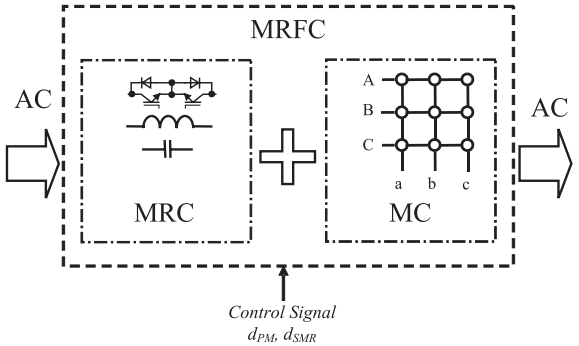


Fig. 16.29. Block scheme of an MRFC

Exemplary time waveforms illustrating the operation of an MRFC with a buck-boost topology, without an input filter and for resistance load, are shown in Figs. 16.33 and 16.34, whereas a detailed operational description of the whole family of MRFC solutions is presented in (Fedyczak and Szcześniak, 2005; 2006a; 2006b; Fedyczak *et al.*, 2006). For an MRFC with a buck-boost and Zeta topology, in each switching period T_S, in the time t_S (switching time of the source switches S_{jk}, where $j = \{1, 2, 3\}$ is the number of the output phase, $k = \{1, 2, 3\}$ is the number of the input phase), 3 of 9 source switches, S_{1k}, S_{2k}, S_{3k}, are simultaneously closed; whereas in the time t_L load the switches S_{L1}, S_{L2}, S_{L3} are simultaneously closed. For an MRFC with the Ćuk topology in each switching period T_S, in the time t_S, 3 of 9 load switches, S_{1k}, S_{2k}, S_{3k}, are simultaneously closed, and at the same time the source switches S_{S1}, S_{S2}, S_{S3} are also simultaneously closed. In the time t_L all source switches are simultaneously closed.

Defining the state function of the source switches as

$$s_{jk} = \begin{cases} 1, & \text{switch } S_{kj} \text{ closed}, \\ 0, & \text{switch } S_{kj} \text{ open}, \end{cases} \tag{16.8}$$

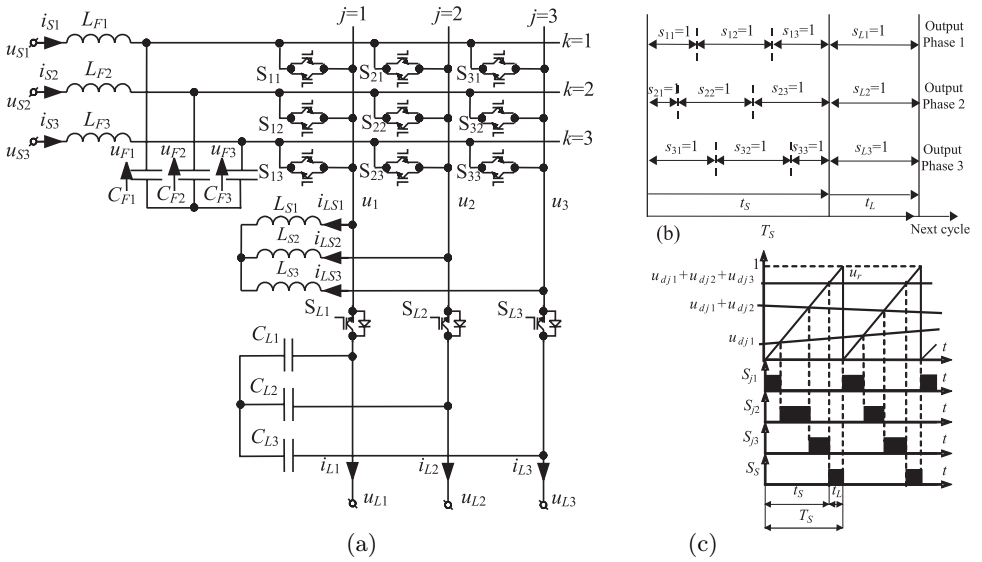


Fig. 16.30. MRFC with the buck-boost topology, (a) schematic diagram of the main circuit, (b) general form of control strategy description, (c) exemplary time waveforms of the control signals for switches in one phase

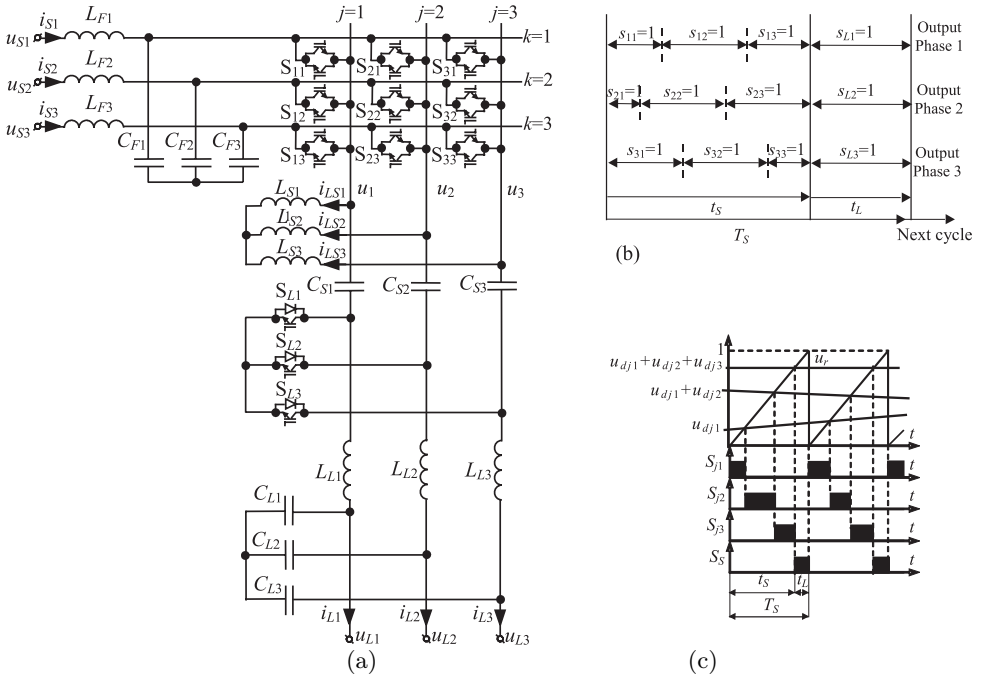


Fig. 16.31. MRFC with the Zeta topology, (a) schematic diagram of the main circuit, (b) general form of control strategy description, (c) exemplary time waveforms of the control signals for switches in one phase

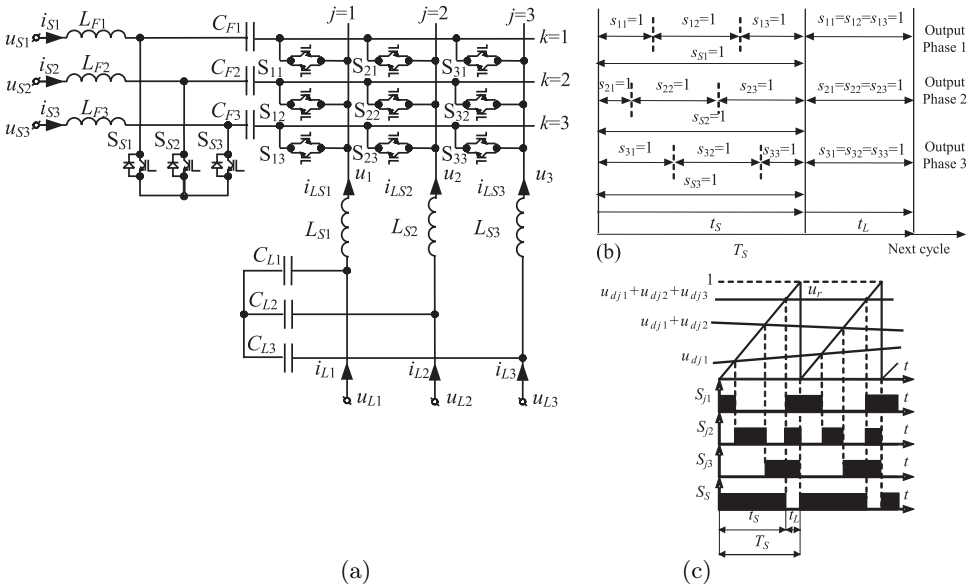


Fig. 16.32. MRFC with the Ćuk topology, (a) schematic diagram of the main circuit, (b) general form of control strategy description, (c) exemplary time waveforms of the control signals for switches in one phase

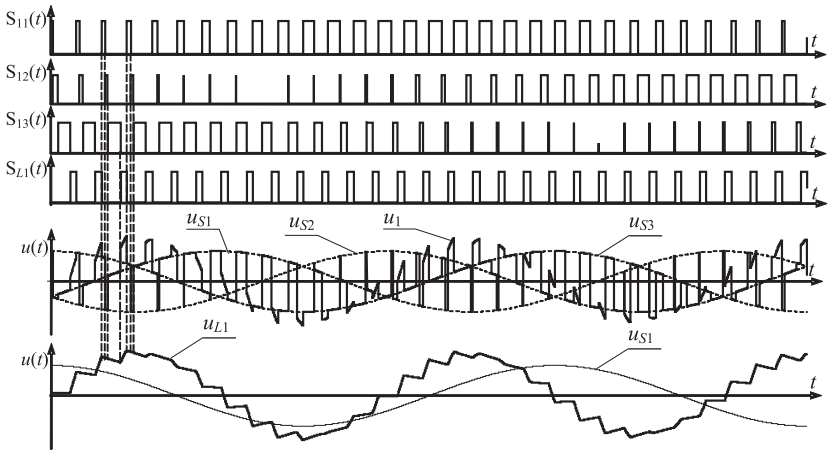


Fig. 16.33. Exemplary voltage time waveforms in a circuit with an MRFC based on the buck-boost topology (without the input filter), for the switching frequency $f_S = 1\,\text{kHz}$ at the pulse duty factor $D_j = 0.75$, and the load voltage setting frequency $f_L = 75\,\text{Hz}$

and assuming allowed constraints of the matrix switches in the time t_S (Venturini and Alesina, 1980):

$$s_{j1} + s_{j2} + s_{j3} = 1, \tag{16.9}$$

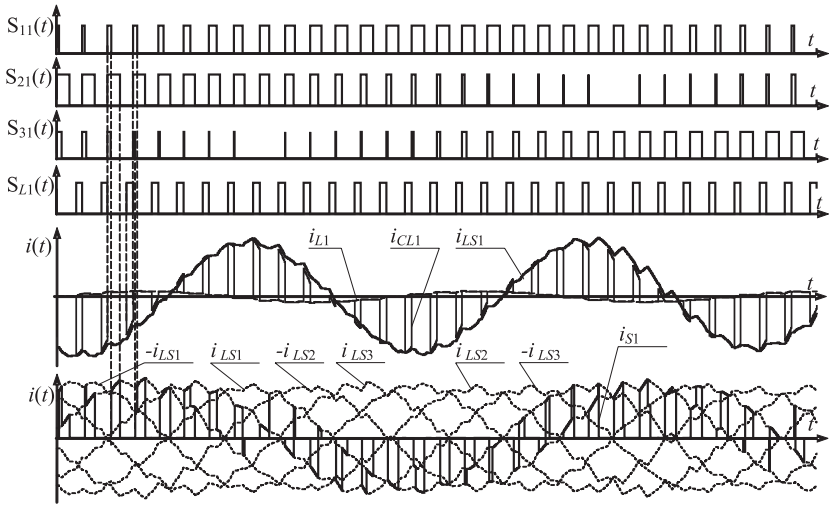


Fig. 16.34. Exemplary current time waveforms in a circuit with an MRFC with the buck-boost topology (without the input filter), for the switching frequency $f_S = 1\,\text{kHz}$ at the pulse duty factor $D_j = 0.75$, and the load voltage setting frequency $f_L = 75\,\text{Hz}$

we can use 27 switching configurations of these switches in the time (t_S), similarly as in the MC (Fig. 16.35), (Casadei *et al.* 2002). In a switching sequence, in the time T_{Seq}, $(27 + 1)$ work states can occur in the presented MRFC (Fedyczak and Szcześniak, 2005).

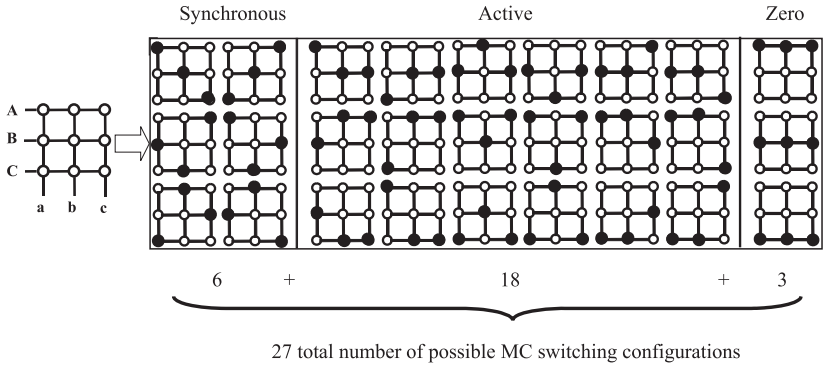


Fig. 16.35. Switching configurations of MC switches in the time t_S

16.3.2. Modelling

The steady-state averaged state-space equation of the presented circuits is described as

$$\begin{aligned} \alpha\,\dot{\bar{\boldsymbol{x}}} &\approx \boldsymbol{A}(D)\,\bar{\boldsymbol{x}} + \boldsymbol{B}(D)\,u_S, \\ \bar{\boldsymbol{y}} &= \boldsymbol{C}(D)\,\bar{\boldsymbol{x}}, \end{aligned} \tag{16.10}$$

where $\dot{\bar{\boldsymbol{x}}}$ is the vector of the averaged state variables:

$$\dot{\bar{\boldsymbol{x}}} = \left[\begin{array}{cccccc} \bar{i}_{LS1} & \bar{i}_{LS2} & \bar{i}_{LS3} & \bar{u}_{CL1} & \bar{u}_{CL2} & \bar{u}_{CL3} \end{array}\right]^T$$

– for the buck-boost topology (without input filter), (16.11)

$$\dot{\bar{\boldsymbol{x}}} = \left[\begin{array}{cccccccccccc} \bar{i}_{LS1} & \bar{i}_{LS2} & \bar{i}_{LS3} & \bar{i}_{LL1} & \bar{i}_{LL2} & \bar{i}_{LL3} & \bar{u}_{CS1} & \bar{u}_{CS2} & \bar{u}_{CS3} & \bar{u}_{CL1} & \bar{u}_{CL2} & \bar{u}_{CL3} \end{array}\right]^T$$

– for the Zeta topology, (16.12)

$$\dot{\bar{\boldsymbol{x}}} = \left[\begin{array}{cccccccccccc} \bar{i}_{LF1} & \bar{i}_{LF2} & \bar{i}_{LF3} & \bar{i}_{LS1} & \bar{i}_{LS2} & \bar{i}_{LS3} & \bar{u}_{CF1} & \bar{u}_{CF2} & \bar{u}_{CF3} & \bar{u}_{CS1} & \bar{u}_{CS2} & \bar{u}_{CS3} \end{array}\right]^T$$

– for the Ćuk topology, (16.13)

α is a diagonal matrix containing the reactance elements:

$$\alpha = \left[\begin{array}{cccccc} L_{S1} & 0 & 0 & 0 & 0 & 0 \\ 0 & L_{S2} & 0 & 0 & 0 & 0 \\ 0 & 0 & L_{S3} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{L1} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{L2} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{L3} \end{array}\right]$$

– for the buck-boost topology (without input filter), (16.14)

$$\alpha = \left[\begin{array}{cccccccccccc} L_{S1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_{S2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & L_{S3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_{L1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_{L2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & L_{L3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & C_{S1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{S2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{S3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{L1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{L2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{L3} \end{array}\right]$$

– for the Zeta topology (without input filter), (16.15)

$$\alpha = \begin{bmatrix} L_{F1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & L_{F2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & L_{F3} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & L_{S1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & L_{S2} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & L_{S3} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & C_{F1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{F2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{F3} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{S1} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{S2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & C_{S3} \end{bmatrix}$$

– for the Ćuk topology, (16.16)

$$\bar{\boldsymbol{y}} = \begin{bmatrix} \bar{i}_{S1} & \bar{i}_{S2} & \bar{i}_{S3} & \bar{u}_{L1} & \bar{u}_{L2} & \bar{u}_{L3} \end{bmatrix}^T \text{ – vector of output variables,} \tag{16.17}$$

D is the duty factor of the switch control signal, $\boldsymbol{A}(D)$ is the averaged state matrix and $\boldsymbol{B}(D)$ is the input matrix, $\boldsymbol{C}(D)$ is the output matrix (Table 16.2) (Fedyczak and Szcześniak, 2005; 2006a; 2006b).

Referring to (16.10) and assuming that

$$\alpha\,(\mathrm{d}\bar{\boldsymbol{x}}/\mathrm{d}t) = 0, \tag{16.18}$$

we obtain idealized equations, collected in Table 16.3 (Fedyczak and Szcześniak, 2005; 2006a; 2006b).

Furthermore, on the basis of the classical control strategy attributable to Venturini, assume that there will be realized a control strategy making allowances for changes of the pulse duty factor of the source switch state function s_{jk} expressed by (16.19):

$$\boldsymbol{D} = \frac{1}{3} D_j \begin{bmatrix} 1+2q\cos(\omega_m t) & 1+2q\cos(\omega_m t-2\pi/3) & 1+2q\cos(\omega_m t-4\pi/3) \\ 1+2q\cos(\omega_m t-4\pi/3) & 1+2q\cos(\omega_m t) & 1+2q\cos(\omega_m t-2\pi/3) \\ 1+2q\cos(\omega_m t-2\pi/3) & 1+2q\cos(\omega_m t-4\pi/3) & 1+2q\cos(\omega_m t) \end{bmatrix}, \tag{16.19}$$

where $q = U_{jm}/U_{Sjm}$ is the voltage gain of the source switches set S_{jk} $(0 < q \leq 0.5)$, $\omega_m = \omega_L - \omega$ is the setting value of difference between pulsations of the output and supply voltages.

Figure 16.36 shows the idealized voltage gain characteristics in an MRFC with the buck-boost topology as a function of the load voltage setting frequency and the summarized pulse duty factor D_j, for different values of the voltage gain q of the source switches set S_{jk}. The bold line in Fig. 16.36 marks the unity voltage gain, which is approximately a maximal value available in a classical frequency converter based on the MC structure.

Table 16.2. Averaged state-space matrices in the equation (16.10)

	$\boldsymbol{A}(D)$	$\boldsymbol{B}(D)$	$\boldsymbol{C}(D)$
Buck-Boost	$\begin{bmatrix} 0 & 0 & 0 & (1-D_j) & 0 & 0 \\ 0 & 0 & 0 & 0 & (1-D_j) & 0 \\ 0 & 0 & 0 & 0 & 0 & (1-D_j) \\ -(1-D_j) & 0 & 0 & \frac{-1}{R_{L1}} & 0 & 0 \\ 0 & -(1-D_j) & 0 & 0 & \frac{-1}{R_{L2}} & 0 \\ 0 & 0 & -(1-D_j) & 0 & 0 & \frac{-1}{R_{L3}} \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{21} & d_{31} & 0 & 0 & 0 \\ d_{12} & d_{22} & d_{32} & 0 & 0 & 0 \\ d_{13} & d_{23} & d_{33} & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$
Zeta	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & (1-D) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & (1-D) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & (1-D) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -D & 0 & 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -D & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -D & 0 & 0 & -1 \\ -(1-D) & 0 & 0 & D & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -(1-D) & 0 & 0 & D & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -(1-D) & 0 & 0 & D & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_{L1}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_{L2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_{L3}} \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \\ d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} d_{11} & d_{21} & d_{31} & d_{11} & d_{21} & d_{31} & 0 & 0 & 0 & 0 & 0 & 0 \\ d_{12} & d_{22} & d_{32} & d_{12} & d_{22} & d_{32} & 0 & 0 & 0 & 0 & 0 & 0 \\ d_{13} & d_{23} & d_{33} & d_{13} & d_{23} & d_{33} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$
Ćuk	$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & -(1-D_j) & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & -(1-D_j) & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -(1-D_j) & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & d_{11} & d_{12} & d_{13} & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & d_{21} & d_{22} & d_{23} & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & d_{31} & d_{32} & d_{33} & 0 & 0 & -1 \\ (1-D_j) & 0 & 0 & -d_{11} & -d_{21} & -d_{31} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & (1-D_j) & 0 & -d_{12} & -d_{22} & -d_{32} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & (1-D_j) & -d_{13} & -d_{23} & -d_{33} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_{L1}} & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_{L2}} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & \frac{-1}{R_{L3}} \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}$	$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$

where $D_j = d_{j1} + d_{j2} + d_{j3} = t_S/T_S$ is the summarized pulse duty factor, d_{jk} is the pulse duty factor of the state function of the source switch s_{jk}.

Table 16.3. Idealized voltage and current relationships

Topology	Formula	
Buck-Boost	$\bar{\boldsymbol{u}}_L \approx -\dfrac{\boldsymbol{D}}{(1-D_j)}\boldsymbol{u}_S$	$\bar{\boldsymbol{i}}_S \approx -\dfrac{\boldsymbol{D}^T}{(1-D_j)}\bar{\boldsymbol{i}}_L$
Zeta	$\bar{\boldsymbol{i}}_S \approx \dfrac{\boldsymbol{D}^T}{(1-D_j)}\bar{\boldsymbol{i}}_L$	$\bar{\boldsymbol{i}}_S \approx \dfrac{\boldsymbol{D}^T}{(1-D_j)}\bar{\boldsymbol{i}}_L$
Ćuk	$\bar{\boldsymbol{i}}_S \approx \dfrac{\boldsymbol{D}^T}{(1-D_j)}\bar{\boldsymbol{i}}_L$	$\bar{\boldsymbol{i}}_S \approx \dfrac{\boldsymbol{D}^T}{(1-D_j)}\bar{\boldsymbol{i}}_L$

where

$$\boldsymbol{D} = \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \\ d_{31} & d_{32} & d_{33} \end{bmatrix}, \tag{16.20}$$

d_{jk} – pulse duty factor of the state function of the source switch s_{jk}.

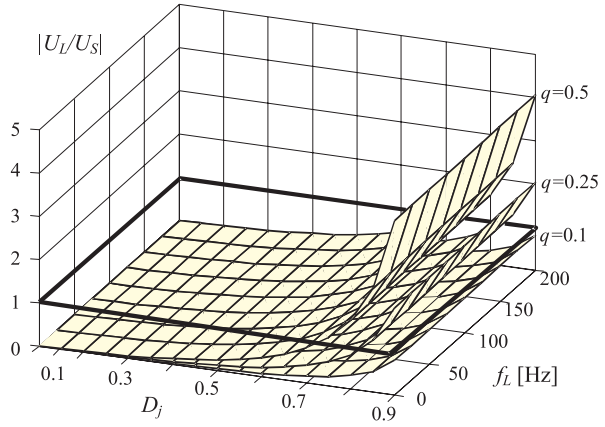
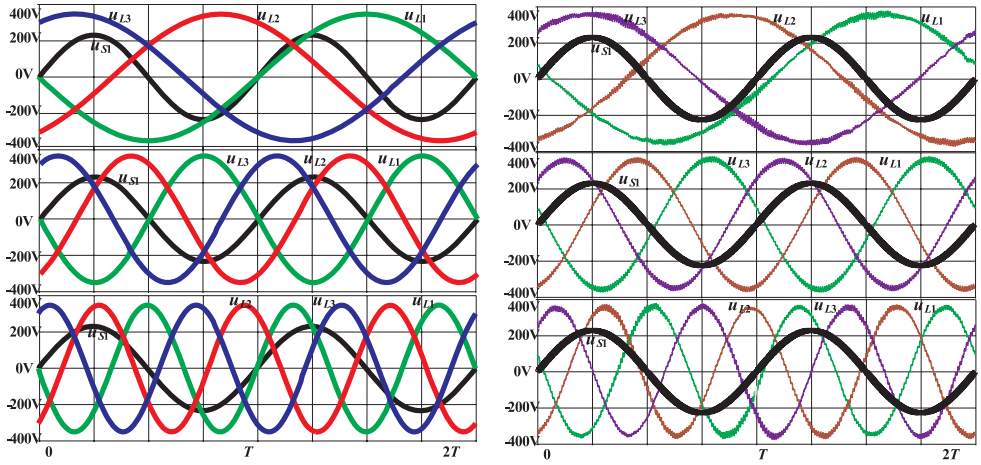


Fig. 16.36. Idealized 3D voltage gain characteristic in an MRFC with the buck-boost topology

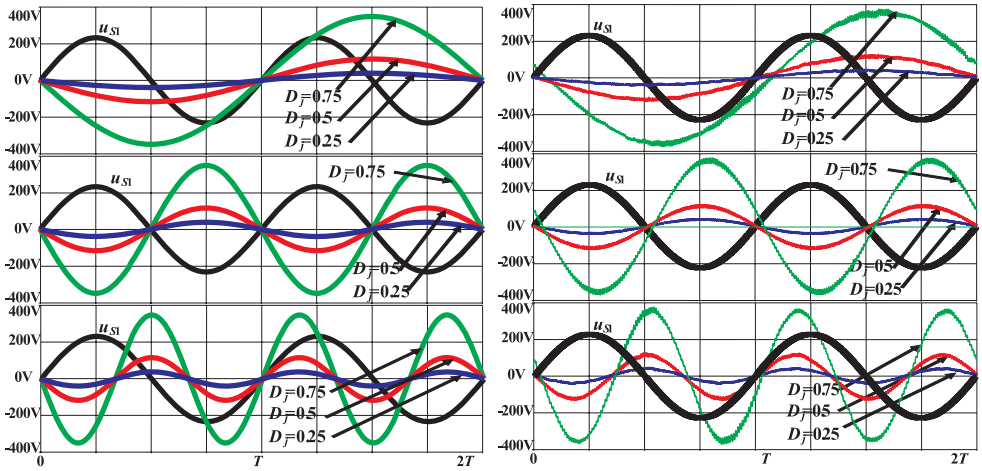
16.3.3. Selected simulation test results

In Figs. 16.37 and 16.38 there are shown exemplary time waveforms of load voltages for different settings of these voltages for $q = 0.5$ and for different values of the summarized pulse duty factor D_j. Next, exemplary time waveforms, in this case of the source current for different values of the setting pulsation of the phase load voltage, are shown in Fig. 16.39. Following, Figs. 16.40–16.42 show the voltage gain characteristics for the discussed MRFC with the buck-boost, Zeta and Ćuk topology, respectively, as a function of the load voltage setting frequency and the summarized pulse duty factor D_j at a near matching condition of the load resistance ($R_L \approx 3\sqrt{L_{Sj}/C_{Lj}} = 3\sqrt{L_{Fj}/C_{Fj}}$) (Fedyczak, 2003a). The simulation investigations have been carried out



(a) (b)

Fig. 16.37. Phase load voltage time waveforms for an MRFC with the Ćuk topology (for $f_L = 25, 50, 75\,\mathrm{Hz}$ from top to bottom, at the value $q = 0.5$) for $D_j = 0.75$, (a) obtained analytically, (b) obtained during simulation



(a) (b)

Fig. 16.38. Phase load voltage time waveforms for an MRFC with the Ćuk topology (for $f_L = 25, 50, 75\,\mathrm{Hz}$ from top to bottom, at the value $q = 0.5$) for different D_j, (a) obtained analytically, (b) obtained during simulation

with the help of the program PSpice, and the relevant circuit parameters are collected in Table 16.4. Both the time waveforms shown in Figs. 16.37 and 16.38 and the voltage gain characteristics shown in Figs. 16.40–16.42 confirm that by means of the discussed MRFC frequency conversion and buck-boost, load voltage changes are possible. Using a simple control strategy according to (16.19) for the summarized pulse duty factor $D_j >$ c.a. 0.7, a load voltage greater than the supply voltage can be obtained.

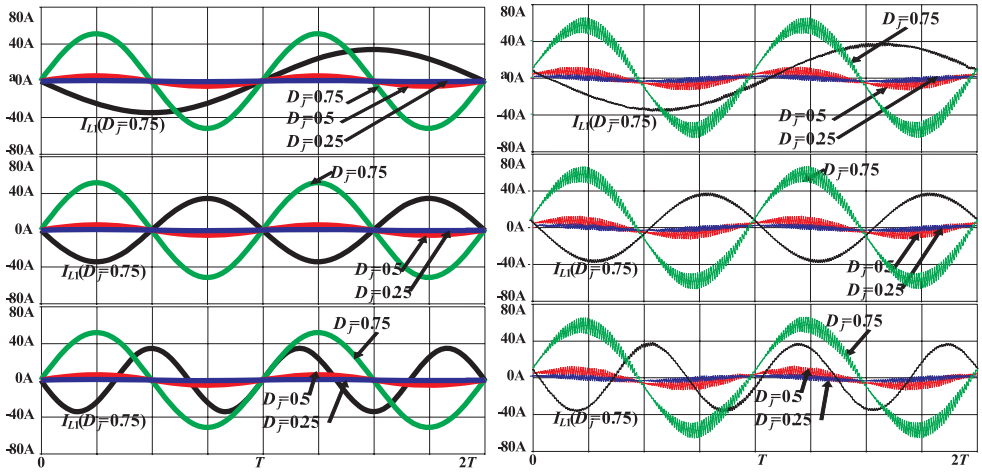


Fig. 16.39. Phase source current time waveforms for an MRFC with the Ćuk topology (for $f_L = 25, 50, 75$ Hz from up to down, at the value $q = 0.5$) for different D_j, (a) obtained analytically, (b) obtained during simulation

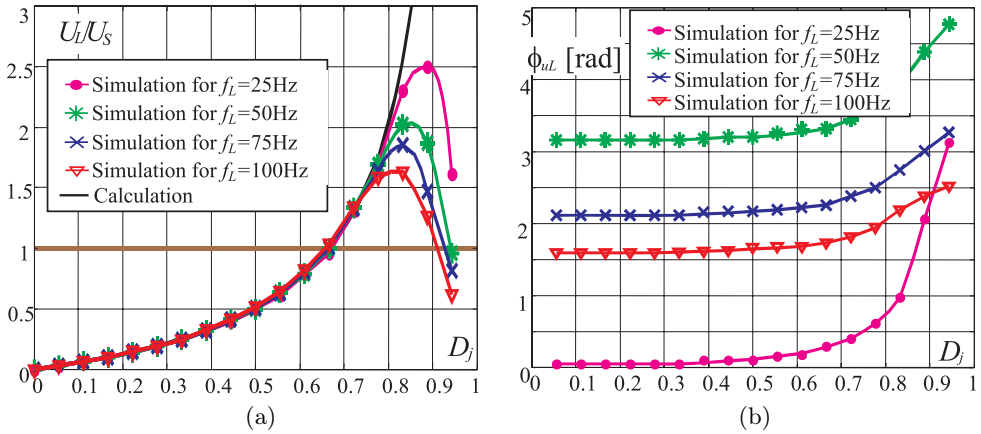


Fig. 16.40. 2D simulation characteristics of the phase load voltage gain (a), and phase shifting (b), as a function of D_j for an different f_L, for an MRFC with the buck-boost topology

As is visible in Figs. 16.37–16.42, the load voltages and source currents time waveforms and their characteristics are intrinsically different from the ones obtained by means of idealized dependences (Table 16.3). This difference is caused by the influence of the passive element parameters, which are used in the MRFC circuit. One is also visible both in Figs. 16.40–16.43, where changes of the load voltage phase and input power factor characteristics are shown, respectively. A detailed description of the influence of the passive element parameters on the properties of the MRFC is

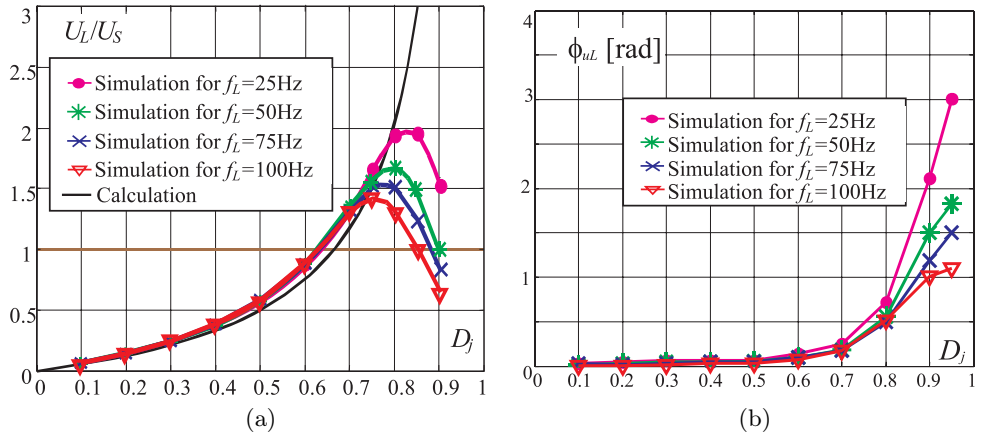


Fig. 16.41. 2D simulation characteristics of the phase load voltage gain (a), and phase shifting (b), as a function of D_j for different f_L, for an MRFC with the Zeta topology

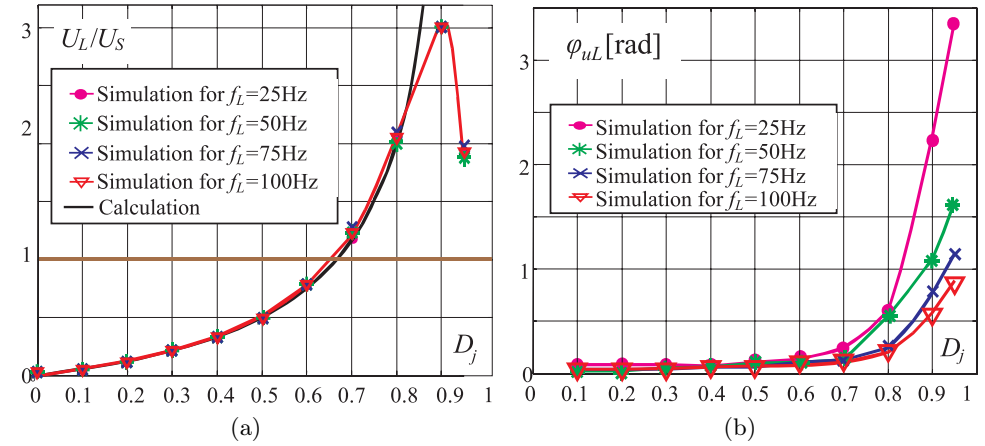


Fig. 16.42. 2D simulation characteristics of the phase load voltage gain (a), and phase shifting (b), as a function of D_j for different f_L, for an MRFC with the Ćuk topology

in this chapter treated as a distinct issue, which will be analysed in further research. In general, the influence of these parameters is similar to the influences in the MRC with a respective topology (Fedyczak, 2003a).

Schematic diagrams of the proposed main circuit experimental setup of the MRFC are shown in Figs. 16.30–16.32, whereas a block diagram of the control circuit is shown in Fig. 16.44 (Fedyczak *et al.*, 2006). We use 9 pairs of RB IGBTs IXRH 40N120 (Reverse Blocking Insulated Gate Bipolar Transistors) in the structure of the source switches S_{jk}, as the bipolar and bidirectional switches, and 3 IGBTs with a freewheeling diode IRG4PH50KD in the structure of the load switches S_L, as the unipolar and bidirectional ones.

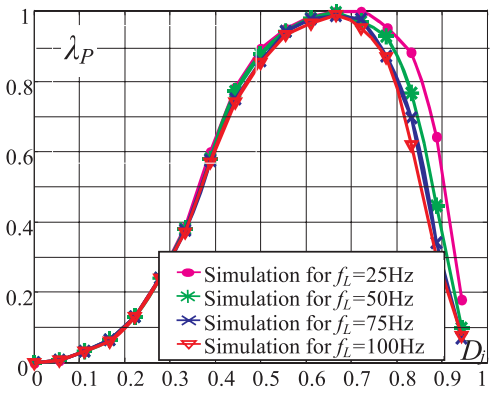


Fig. 16.43. 2D simulation characteristics of the input power factor as a function of the pulse duty factor D_j for different load voltage setting frequency, for an MRFC with the buck-boost topology

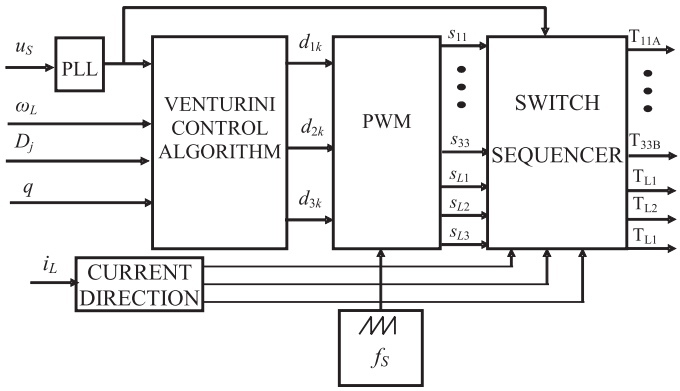


Fig. 16.44. Block diagram of the control circuit of the presented MRFC

Table 16.4. Theoretical and simulation tests circuit parameters

Parameter	Symbol	Value
Supply voltage	U_S	230 V
Supply frequency	f	50 Hz
Switching frequency	f_S	5 kHz
Inductances	L_{F1}–L_{F3}, L_{S1}–L_{S3}	0.5 mH
Capacitances	C_{F1}–C_{F3}, C_{L1}–C_{L3}	50 μF
Load resistance	R_L	10 Ω

16.4. Conclusions and further research

In the framework of PWM AC line choppers, further research will be focused on the implementation of both MCHs and MRCs within the following application proposals: in temperature control devices as electrical power controllers; in drive systems for electrical motor "soft starters"; in flexible AC transmission systems in series AC voltage stabilizers (Fedyczak *et al.*, 2005; 2006); in the quadrature-booster phase shifter (Fedyczak and Jankowski, 2005) in VAr compensators (Fedyczak *et al.*, 2000). Furthermore, new research will be focused on developing single- and three-phase Hybrid Transformer (HT) circuits with PWM AC line choppers (Fedyczak and Kaniewski, 2006; Kaniewski, 2006).

In the frame of MRFCs, further research will be focused on a detailed theoretical analysis of the presented MRFC properties in steady and transient states. The implementation of other control strategies used in MCs will also be continued for different load conditions. Furthermore, the formulation of an improved control strategy for an input power factor with a wider range of load parameters and the continuation of the research producing results obtained by experimental verification are also planned.

References

Apap M., Clare J.C., Wheeler P.W. and Bradley K.J. (2003): *Analysis and comparison of AC-AC matrix converter control strategies.* — Proc. IEEE 34th *Annual Power Electronics Specialist Conf.*, PESC, Acapulco, Mexico, Vol. 3, pp. 1287–1292.

Casadei D., Serra G., Tanti A. and Zaroi L. (2002): *Matrix converter modulation strategies: A new general approach based on space-vector representation of switch state.* — IEEE Trans. Industrial Electronics, Vol. 49, No. 2, pp. 370–381.

Clark J.W. (1990): *AC Power Conditioners: Design and Applications.* — New York: Academic Press, Inc.

Fedyczak Z. (2001): *Four-terminal chain parameters of averaged AC models out of basic non-isolated matrix-reactance PWM AC line conditioners.* — Archive of Electrical Engineering, Vol. 50, No. 4, pp. 395–409.

Fedyczak Z. (2003a): *PWM AC Voltage Transforming Circuits.* — Oficyna Wydawnicza Uniwersytetu Zielonogórskiego, (in Polish).

Fedyczak Z. (2003b): *Steady state modeling and circuit functions of the bipolar PWM AC line matrix-reactance choppers.* — Proc. 3rd Int. Workshop *Compatibility in Power Electronics, CPE*, Gdańsk–Zielona Góra, Poland, pp. 206–213.

Fedyczak Z. (2003c): *Steady state modelling of the bipolar PWM AC line matrix-reactance choppers based on Ćuk topologies.* — Archive of Electrical Engineering, Vol. 52, No. 3, pp. 303–316.

Fedyczak Z. and Jankowski M. (2005): *Modeling and analysis of the quadrature-booster phase shifter with PWM AC bipolar MC and passive load.* — Proc. 4th Int. Workshop *Compatibility in Power Electronics*, CPE, Gdynia, Poland, CD-ROM.

Fedyczak Z. and Kaniewski J. (2006): *Single phase hybrid transformer using bipolar matrix – reactance chopper.* — Przegląd Elektrotechniczny, No. 7–8, pp. 80–85, (in Polish).

Fedyczak Z. and Korotyeyev I. (2003): *Bipolar PWM AC line matrix-reactance choppers – the steady state basic energetic properties.* — Proc. 10th European Conf. *Power Electronics and Applications, EPE*, Toulouse, France, CD-ROM.

Fedyczak Z. and Strzelecki R. (1994): *Three-phase impulse power controller with (2m-2) transistorized keys.* — Proc. Int. Conf. *Power Electronics and Motion Control, PEMC*, Warsaw, Poland, Vol. 1, pp. 225–230.

Fedyczak Z. and Strzelecki R. (1997): *Power Electronic Circuits Used for AC Power Control.* — Toruń: Wydawnictwo Adam Marszałek, (in Polish).

Fedyczak Z. and Strzelecki R. (1998): *Three-phase PWM AC line boost and buck-boost conditioners under small quality factor circumstance.* — Proc. 15th Symp. *Electromagnetic Phenomena in Nonlinear Circuits, EPNC*, Liege, Belgium, pp. 184–187.

Fedyczak Z. and Szcześniak P. (2005): *Study of matrix-reactance frequency converter with buck-boost topology.* — Proc. Int. Conf. *Power Electronics and Intelligent Control for Energy Conservation, PELINCEC*, Warsaw, Poland, CD-ROM.

Fedyczak Z. and Szcześniak P. (2006a): *Study of matrix-reactance frequency converter with Zeta topology.* — Wiadomości Elektrotechniczne, No. 3, pp. 26–29, (in Polish).

Fedyczak Z. and Szcześniak P. (2006b): *Study of matrix-reactance frequency converter with Ćuk topology.* — Przegląd Elektrotechniczny, No. 7/8, pp. 42–47, (in Polish).

Fedyczak Z. Strzelecki R. and Benysek G. (2002a): *Single-phase PWM AC/AC semiconductor transformer topologies and applications.* — Proc. IEEE 33rd *Annual Power Electronics Specialists Conf., PESC*, Cairns, Australia, Vol. 2, pp. 1048–1053.

Fedyczak Z., Frąckowiak L. and Jankowski M. (2006): *A serial AC voltage controller using bipolar matrix-reactance chopper.* — Int. J. Computation and Mathematics in Electrical and Electronic Engineering, COMPEL, Vol. 25, No. 1. pp. 244–258.

Fedyczak Z., Jankowski M. and Szcześniak P., (2005): *A comparison of basic properties of single-phase serial AC voltage controllers using bipolar PWM chopper.* — Proc. 11th European Conf. *Power Electronics and Applications, EPE*, Dresden, Germany, CD-ROM.

Fedyczak Z., Klytta M. and Strzelecki R. (2001a): *Three-phase AC/AC semiconductor transformer topologies and applications.* — Proc. 2nd Conf. *Power Electronics Devices Compatibility, PEDC*, Zielona Góra, Poland, pp. 25–38.

Fedyczak Z., Strzelecki R. and Skórski K. (1999): *Three-phase PWM AC line conditioner based on the Ćuk converter topology: Study of the basic energetic properties.* — Proc. 8th European Conf. *Power Electronics and Applications, EPE*, Lausanne, Switzerland, pp. P1–P10.

Fedyczak Z., Strzelecki R. and Sozański K. (2002b): *Review of three-phase AC/AC semiconductor transformer topologies and applications.* — Proc. Symp. *Power Electronics Electrical Drives Automation & Motion, SPEEDAM*, Ravello, Italy, pp. B5-19–B.5-24.

Fedyczak Z., Strzelecki R., Frąckowiak L. and Kempski A. (2001b): *An AC voltage transformation circuits based on Zeta or Sepic PWM AC line conditioners.* — Proc. 9th European Conf. *Power Electronics and Applications, EPE*, Graz, Austria, CD-ROM.

Fedyczak Z., Strzelecki R., Kasperek R. and Skórski K. (2000): *Three-phase self-commutated VAr compensator based on Ćuk converter topology.* — Proc. IEEE 31st *Annual Power Electronics Specialists Conf., PESC*, Galway, Ireland, Vol. 1, pp. 494–499.

Fedyczak Z., Szcześniak P. and Klytta M. (2006): *Matrix-reactance frequency converter based on buck-boost topology.* — Proc. 12th Int. Conf. *Power Electronics and Motion Control, EPE-PEMC*, Portoroz, Słowenia, pp. 763–768.

Helle L., Larsen K.B., Jorgensen A.H., Munk-Nielsen S. and Blaabjerg F. (2004): *Evaluation of modulation schemes for three-phase to three-phase matrix converters*. — IEEE Trans. Industrial Electronics, Vol. 51, No. 1, pp. 158–171.

Kaniewski J. (2006): *Single phase hybrid transformer using matrix converter*. — Wiadomości Elektrotechniczne, No. 3, pp. 46–48, (in Polish).

Kim J.H., Min B.D., Kwon B.H. and Won S.C. (1998): *A PWM buck-boost AC chopper solving the commutation problem*. — IEEE Trans. Industry Electronics, Vol. 45, No. 5, pp. 832–835.

Korotyeyev I.Y. and Fedyczak Z. (2005): *Steady state modelling of basic unipolar PWM AC line matrix-reactance choppers*. — Int. J. Computation and Mathematics in Electrical and Electronic Engineering, COMPEL, Vol. 24, No. 1, pp. 55–68.

Korotyeyev I.Y., Fedyczak Z., Strzelecki R. and Sozański K. (2001): *An averaged AC models accuracy evaluation of non-isolated matrix-reactance PWM AC line conditioners*. — Proc. 9th European Conf. *Power Electronics and Applications, EPE*, Graz, Austria, CD-ROM.

Middlebrook R.D. and Ćuk S. (1976): *A general unified approach to modelling switching-converter power stages*. — Proc. IEEE *Power Electronics Specialists Conf., PESC*, San Diego, USA, pp. 18–34.

Strzelecki R. and Fedyczak Z. (1995a): *Three-phase PWM AC power controller with (3+3) unilateral switches*. — Proc. 6th European Conf. *Power Electronics and Applications, EPE*, Sevilla, Spain, pp. 3.292–3.297.

Strzelecki R. and Fedyczak Z. (1995b): *Three-phase PWM AC line matrix choppers*. — Przegląd Elektrotechniczny, No. 4, pp. 85–93, (in Polish).

Strzelecki R. and Fedyczak Z. (1996a): *Economical circuit of a three-phase PWM AC power controller with a new control algorithm without "dead time"*. — Proc. IEEE Int. *Power Electronics Congress*, Cuernavaca, Mexico, pp. 77–82.

Strzelecki R. and Fedyczak Z. (1996b): *Properties and structures of three-phase PWM AC line conditioners*. — Proc. Symp. *Power Electronics, Electrical Drives, Advanced Electrical Motors*, Capri, Italy, pp. B3/11–18.

Strzelecki R. and Fedyczak Z. (1996c): *Properties and structures of three-phase PWM AC power controllers*. — Proc. 27th *Power Electronics Specialists Conf., PESC*, Baveno, Italy, Vol. 1, pp. 740–746.

Strzelecki R., Fedyczak Z. and Kasperek R. (1996a): *Design and tests of a three-phase PWM AC power controller with two transistorized switches*. — Proc. IEEE Int. Symp. *Industrial Electronics*, Warsaw, Poland, pp. 1/499–504.

Strzelecki R., Fedyczak Z. and Kasperek R. (1996b): *Three-phase PWM AC power controller with active by-pass suppressor circuit*. — Proc. IEEE 7th Int. *Power Electronics & Motion Control Conference*, Budapest, Hungary, pp. 1/306–309.

Strzelecki R., Fedyczak Z., Kobyłecki G. and Kasperek R. (1997a): *Improvement methods of conversion quality in three-phase AC line power controllers – Topology and basic properties*. — Proc. 7th European Conf. *Power Electronics and Applications, EPE*, Trondheim, Norway, pp. 2.940–2.945.

Strzelecki R., Fedyczak Z., Kobyłecki G. and Kasperek R. (1997b): *AC line power control methods in three-phase circuits*. — Proc. 4th Int. Conf. *Electrical Power Quality and Utilisation, EPQU*, Cracow, Poland, pp. 225–232.

Tunia H., Strzelecki R. and Fedyczak Z. (1998): *AC power control metod in three-phase actuators and three-phase PWM AC line matrix Hopper.* — Polish patent No. 176234, 1998.12.03, (in Polish).

Venturini M. and Alesina A. (1980): *The generalized transformer: a new bi-directional sinusoidal waveform frequency converter with continuously adjustable input power factor.* — Proc. IEEE *Power Electronics Specialists Conf., PESC*, Atlanta, USA, pp. 242–252.

Wheeler P.W., Rodriguez J., Clare J.C., Empringham L. and Weinstejn A. (2002): *Matrix converters: A technology review.* — IEEE Trans. Industrial Electronics, Vol. 49, No. 2, pp. 276–288.

Zinoviev G.S., Obuchov A.Y., Otchenasch W.A. and Popov W.I. (2000): *Transformerless PWM AC boost and buck-boost converters.* — Technicznaja Elektrodinamika, Vol. 2, pp. 36–39, (in Russian).

Chapter 17

ANALYSIS OF PROCESSES IN CONVERTER SYSTEMS

Igor Ye. KOROTYEYEV*, Radosław KASPEREK*

17.1. Introduction

Electromagnetic processes in converters with closed-loop feedback are described by non-linear differential and algebraic equations. For the solution of such equations, numerical and numerical-analytical methods are used. Experimental research and computer simulations in DC/DC converters show the existence of periodic, quasi-periodic and chaotic oscillations (Banerjee and Verghese, 2001; Strzelecky *et al.*, 2001; Zhuykov and Korotyeyev, 2001). The sort of the oscillations is determined by the relation of the parameters between the elements of the converter, a control system, a power supply and a load. For the process identification there are used Poincaré sections, a calculation of Lyapunov exponents, a correlation function and a fractal dimension. The periodic processes can be distinguished by stability analyses. As chaotic processes are non-periodical, long time intervals are needed for their identification. In some cases it is difficult to distinguish quasi-periodical and chaotic processes.

AC/AC converters are usually employed as power conditioners. With advanced control strategies the conditioner also enables the suppression of sags or unbalances of the load voltage in power supplies and the compensation of reactive power (Jang and Choe, 1995; Kasperek, 2003). In such converters there are used control methods providing the possibility for a dynamic change of the transformation ratio with a time constant which is much less than the period of the supply voltage (Lefeuvre *et al.*, 2001; Veszpremi and Hunyar, 2000). Since processes in such systems are described by non-linear and non-stationary equations, the balance between the use of numerical and analytical methods for process analysis is a complex problem. The problems of process analysis in DC/DC and AC/AC converters are considered in this chapter.

* Institute of Electrical Engineering
e-mails: {I.Korotyeyev, R.Kasperek}@iee.uz.zgora.pl

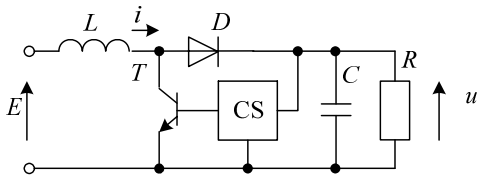


Fig. 17.1. Topology of a Boost converter

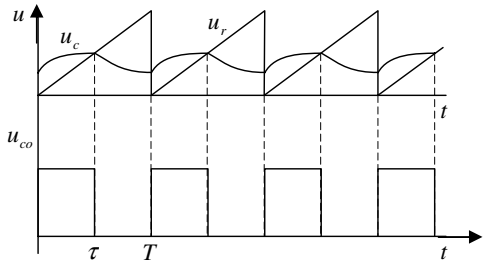


Fig. 17.2. Time diagrams of voltages in the control system with PWM

17.2. Analysis of processes in a DC/DC converter

17.2.1. Mathematical model

Let us analyse processes in the Boost DC voltage converter shown in Fig. 17.1 (Korotyeyev and Klytta, 2002). Assume that the transistor and diode are described by Series Resistance (RS) models and in the on-state have the same resistances; the inductor and capacitor are linear elements. The processes in the control system CS with Pulse Width Modulation (PWM) are presented in Fig. 17.2.

In the control system, processes are described by the following equation set:

$$u_c = k(u_{\rm ref} - k_r u), \quad u_{\rm com} = u_c - u_r, \quad \gamma = \gamma(u_{\rm com}), \tag{17.1}$$

where k_r is the output voltage ratio, k is the voltage feedback gain, u_c is the control voltage, $u_{\rm ref}$ is the reference voltage, $u_{\rm com}$ is the voltage on an input of a comparator, u_r is the independent sawtooth ramp voltage, T is the period of the voltage of a generator, τ is the impulse duration on the output of the control system, $\gamma(t)$ is the switching function (Fig. 17.3).

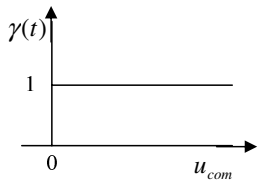


Fig. 17.3. Switching function

Let us write differential equations for the intervals of constancy of converter topology. On the interval $nT \leq t \leq nT + t_n$ (n is the number of periods), the transistor is in the on-state and the diode in the off-state. The inductor current i and the voltage across the capacitor u are described by the differential equations

$$E = r_1 i + L\frac{\mathrm{d}i}{\mathrm{d}t}, \quad 0 = C\frac{\mathrm{d}u}{\mathrm{d}t} + \frac{u}{R}, \tag{17.2}$$

where $r_1 = r_t + r_i$ is the sum of resistances of the inductor and the transistor in the on-state.

On the interval $nT + t_n \leq t \leq (n+1)T$ the transistor is off, the diode on. The differential equations describing the electromagnetic processes are as follows:

$$r_2 i + L\frac{\mathrm{d}i}{\mathrm{d}t} + u, \quad i = C\frac{\mathrm{d}u}{\mathrm{d}t} + \frac{u}{R}, \tag{17.3}$$

where r_2 is the the sum of resistances of the inductor and the diode in the on-state, $r_2 = r_1$.

Let us combine the equations (17.2) and (17.3) using the switching function. We assume that the value of the switching function equal to one corresponds to the on-state of the transistor and that the zero value corresponds to the off-state of the transistor. Taking this into account we can write the differential equations for all intervals in the form

$$L\frac{\mathrm{d}i}{\mathrm{d}t} = -r_1 i - (1-\gamma)u + E, \quad C\frac{\mathrm{d}u}{\mathrm{d}t} = (1-\gamma)i - \frac{1}{R}u. \tag{17.4}$$

This equation set we represent as follows:

$$\frac{\mathrm{d}X}{\mathrm{d}t} = A(\gamma)X + B, \tag{17.5}$$

where $X = \begin{vmatrix} i \\ u \end{vmatrix}$ is the vector of the state variables, $A(\gamma) = \begin{vmatrix} -\frac{r_1}{L} & -\frac{1-\gamma}{L} \\ \frac{1-\gamma}{C} & -\frac{1}{RC} \end{vmatrix}$, $B = \begin{vmatrix} \frac{E}{L} \\ 0 \end{vmatrix}$. The set of the equations (17.1) and (17.5) describes processes in the closed-loop system of the converter with PWM-2.

A steady-state process is determined on the basis of finding an initial condition which satisfies the condition $X(nT) = X((n+1)T)$. For the interval $nT \leq t \leq nT + t_n$, when the transistor is on, the processes are described by the expression

$$X(t) = e^{A_1(t-nT)}X(nT) + A_1^{-1}\left(e^{A_1(t-nT)} - I\right)B, \tag{17.6}$$

and for the interval when the transistor is off, the processes are described as follows:

$$X(t) = e^{A_2(t-nT-t_n)}X(nT+t_n) + A_2^{-1}(e^{A_2(t-nT-t_n)} - I)B. \tag{17.7}$$

where e^{At} is the matrix exponential. Substituting $t = nT + t_n$ in (17.6) and $t = (n+1)T$ in (17.7) and taking into account the periodicity condition $X(nT) = X((n+1)T)$, we find the steady-state process

$$X(nT) = (I - e^{A_2(T-t_n)}e^{A_1 t_n})^{-1}[e^{A_2(T-t_n)}A_1^{-1}(e^{A_1 t_n} - I) + A_2^{-1}(e^{A_2(t-nT-t_n)} - I)]B, \tag{17.8}$$

where $t_n = \mathrm{const}$.

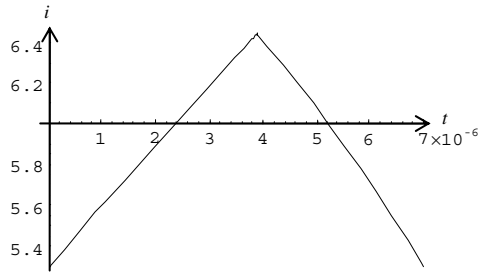


Fig. 17.4. Steady-state inductor current

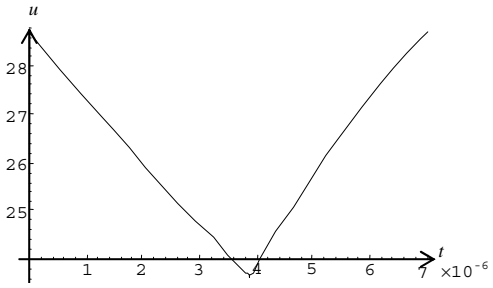


Fig. 17.5. Steady-state voltage across the capacitor

17.2.2. Calculation of processes and stability in closed-loop systems

Let us calculate steady-state and transitional processes for the following values of parameters: $E = 12$ V, $U_{ag} = 5$ V, $r_1 = 0.05\ \Omega$, $R = 10\ \Omega$, $C = 2 \times 10^{-6}$ F; $L = 4 \times 10^{-5}$ H; $T = 7 \times 10^{-6}$ s; $k_r = 0.01$, $U_{\text{ref}} = 1.5$ V, $k = 2.2$. The graphs of the steady-state processes are presented in Figs. 17.4 and 17.5.

The transitional process in the circuit of the converter is calculated for the initial values equal to zero. Time diagrams of the transient process for the current and the voltage are presented in Figs. 17.6 and 17.7, respectively.

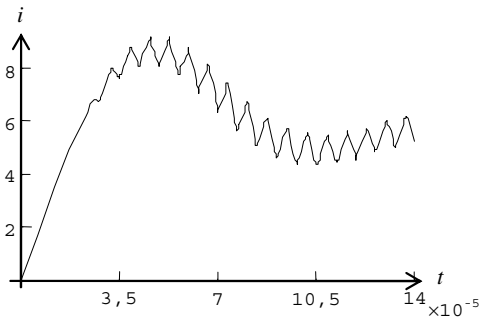


Fig. 17.6. Time diagram of the transient inductor current

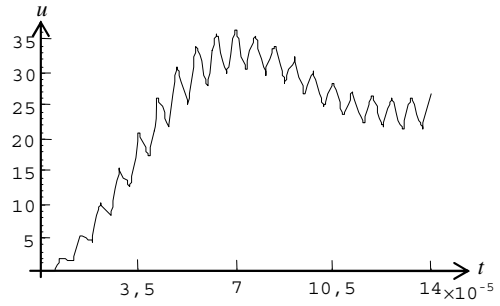


Fig. 17.7. Time diagram of the transient voltage across the capacitor

The equation sets (17.1) and (17.5) are non-linear. The stability analysis will be based upon the first Lypunov method. For the stability analysis we use the method (Rozenwasser and Yusupov, 1981), which is based on a linearization of a differential and an algebraic equation set.

Let us linearize the equations (17.1) and (17.5) describing a transient process. We vary the state space variable corresponding to the initial conditions on the infinitesimal value $X_\xi(mT)$ (Zhuykov *et al.*, 1989). In such a way we find equations describing changes of state space variables:

$$\frac{\mathrm{d}X_\xi}{\mathrm{d}t} = A_\xi(\gamma)X + A(\gamma)X_\xi + B_\xi(\gamma), \quad u_{\xi c} = -kk_r u_\xi, \quad u_{\xi\mathrm{com}} = u_{\xi c}, \tag{17.9}$$

where $A_\xi(\gamma)$, $B_\xi(\gamma)$ are variables of the matrix $A(\gamma)$ and the vector $B(\gamma)$. Calculating the expressions for $A_\xi(\gamma)$, $B_\xi(\gamma)$ in (17.9), we obtain

$$\frac{\mathrm{d}X_\xi}{\mathrm{d}t} = A(\gamma)X_\xi + \sum_{\mu=0}^{\infty} D_\mu u_\xi(t_\mu)\delta(t - t_\mu), \tag{17.10}$$

where

$$D_\mu = -kk_r \frac{B_\gamma(y) + A_\gamma(\gamma)X(t_\mu)}{|u_{t\mathrm{com}}(t_\mu)|}, \quad A_\gamma = \frac{\partial A}{\partial \gamma}, \quad B_\gamma = \frac{\partial B}{\partial \gamma},$$

$$A_\gamma(\gamma) = \begin{vmatrix} 0 & \dfrac{1}{L} \\ -\dfrac{1}{C} & 0 \end{vmatrix}, \quad B_\gamma(\gamma) = \begin{vmatrix} 0 \\ 0 \end{vmatrix}, \quad \text{and} \quad u_{t\mathrm{com}}(t_\mu) = \lim_{t \to t_\mu - 0} \frac{\mathrm{d}u_{\mathrm{com}}}{\mathrm{d}t}.$$

The equation (17.10) is a linear non-stationary differential equation. In order to determine stability conditions, it is necessary to find the solution of this equation for the period T:

$$X_\xi((n+1)T) = e^{A_2(T-\tau)} N_2 e^{A_1 \tau} N_1 X_\xi(nT), \tag{17.11}$$

where

$$N_2 = \begin{vmatrix} 1 & d_2^1 \\ 0 & 1+d_2^2 \end{vmatrix}, \quad D_2 = \begin{vmatrix} d_2^1 \\ d_2^2 \end{vmatrix}, \quad d_2^1 = -\frac{kk_r u(\tau)}{k_2 L}, \; d_2^2 = \frac{kk_r i(\tau)}{k_2 C}, \quad k_2 = |u_{\mathrm{tcom}}(\tau)|,$$

$$N_1 = \begin{vmatrix} 1 & d_1^1 \\ 0 & 1+d_1^2 \end{vmatrix}, \; D_1 = \begin{vmatrix} d_1^1 \\ d_1^2 \end{vmatrix}, \quad d_1^1 = -\frac{kk_r u(0)}{k_1 L}, \; d_1^2 = \frac{kk_r i(0)}{k_1 C}, \quad k_1 = |u_{\mathrm{tcom}}(0)|.$$

In line with Lyapunov's first method, if the solution of the equation (17.11) is stable, the initial non-linear system is stable (stability is small). The stability of the linearized system is determined by eigenvalues of the matrix

$$H = e^{A_2(T-\tau)} N_2 e^{A_1 \tau} N_1. \tag{17.12}$$

The system will be stable if all absolute values of eigenvalues of the matrix H are less than unity.

According to the form of the generator voltage, the matrix H takes a different form. The expression (17.12) corresponds to the case when both front edges of the generator voltage have finite slopes. For the case under consideration shown in Fig. 17.2, the matrix takes the form

$$H = e^{A_2(T-\tau)} N_2 e^{A_1 \tau}. \tag{17.13}$$

For the computation of the eigenvalues it is necessary to find the elements of the matrices N_1 and N_2. At first the values for vectors of the state variables $X(nT) = X(0)$, $X(nT+\tau) = X(\tau)$ for the steady-state process should be determined. Next, one computes the values of the coefficients k_1 and k_2. For the calculation of the derivative we use the initial differential equation set

$$\frac{\mathrm{d}X}{\mathrm{d}t} = A_1 X + B_1, \qquad nT \le t \le nT + \tau,$$

$$\frac{\mathrm{d}X}{\mathrm{d}t} = A_2 X + B_2, \qquad nT + \tau \le t \le (n+1)T.$$

The left-hand derivatives of the vector X at the moments of a structure change are defined by the expressions

$$\left.\frac{\mathrm{d}X}{\mathrm{d}t}\right|_{t=\tau-0} = A_1 X(\tau) + B_1, \quad \left.\frac{\mathrm{d}X}{\mathrm{d}t}\right|_{t=-0} = A_2 X(0) + B_2. \tag{17.14}$$

The elements of (17.14) are the derivatives

$$u_t(\tau) = \left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{t=\tau-0}, \quad u_t(0) = \left.\frac{\mathrm{d}u}{\mathrm{d}t}\right|_{t=-0},$$

which are used for the computation of the values $u_{tc}(\tau) = -kk_\delta u_t(\tau)$ and $u_{tc}(0) = -kk_\delta u_t(0)$, respectively.

As a result of the computation we determine the eigenvalues of the matrix H $0.82071 \pm \mathrm{j}0.311253$ and the absolute value of the eigenvalues $\{0.877749,\ 0.877749\}$. Since the absolute value is less than unity, the system is stable.

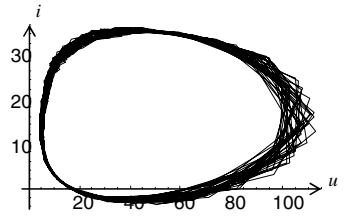


Fig. 17.8. Phase-plane portrait for $k = 3.32$

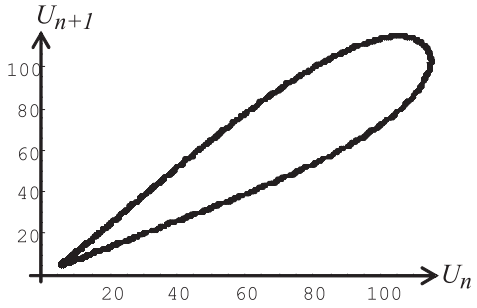


Fig. 17.9. Poincaré section for $u((n+1)T) = f(u(nT))$ for $k = 3.32$

17.2.3. Processes identification

By increasing the gain, a bifurcation takes place at $k = 3.28$ and in the system there is formed a quasi-periodic process. The phase-plane portrait for $k = 3.32$ is presented in Fig. 17.8.

For process identification there are used the Poincaré section, calculations of a correlation function, a fractal dimension and Lyapunow exponents. The Poincare section $u((n+1)T) = f(u(nT))$ for $k = 3.32$ is shown in Fig. 17.9.

During process calculation we found the values of the current and the voltage at the point of time aliquot to the period T. Then it is expedient to use these data for the calculation of a correlation function. For discrete processes the correlation function is defined as follows:

$$K(m) = \lim_{N_M \to \infty} \frac{1}{N_M} \sum_{i=0}^{N_M - 1} \hat{x}_{i+m} \hat{x}_i, \tag{17.15}$$

where $\hat{x}_i = x_i - \bar{x}$, $\bar{x} = \lim_{N_M \to \infty} \frac{1}{N_M} \sum_{i=0}^{N_M - 1} x_i$.

The correlation function for $k = 3.2$ is shown in Fig. 17.10. Both the Poincaré section and the correlation function confirm that the discussed process is quasi-periodic.

Let us analyse processes in the converter for other values of parameters: $E = 12$ V, $U_{ag} = 5$ V, $r_1 = 0.05\ \Omega$, $R = 80\ \Omega$, $C = 2 \times 10^{-6}$ F, $L = 1 \times 10^{-5}$ H, $T = 1 \times 10^{-5}$ s, $k_r = 0.01$. The bifurcation diagram for $2 \leq k \leq 6$ is presented in Fig. 17.11. This bifurcation diagram has been obtained by assuming that bi-directional switches are used in the converter.

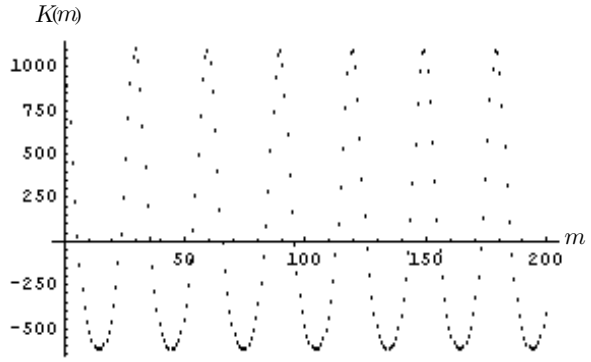


Fig. 17.10. Correlation function for $k = 3.2$

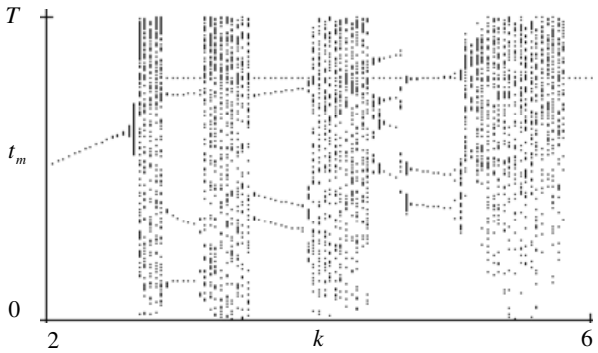


Fig. 17.11. Bifurcation diagram for $2 \leq k \leq 6$

For the gain $k < 2.4$ the period-one operation of the converter is stable. For a gain greater than that in the system, a bifurcation occurs. The phase-plane portrait for $k = 2.759$ is presented in Fig. 17.12.

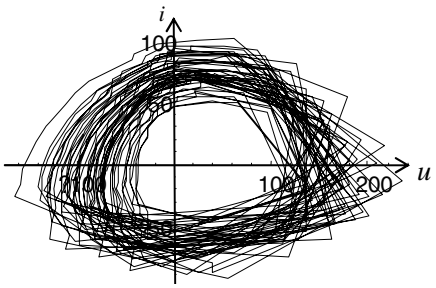


Fig. 17.12. Phase-plane portrait for $k = 2.759$

Poincaré section $u((n+1)T) = f(u(nT))$ for $k = 2.759$ is shown in Fig. 17.13. The calculation is done for an interval equal to $8000T$, at which the plotting is realized for $7500T \leq t \leq 8000T$.

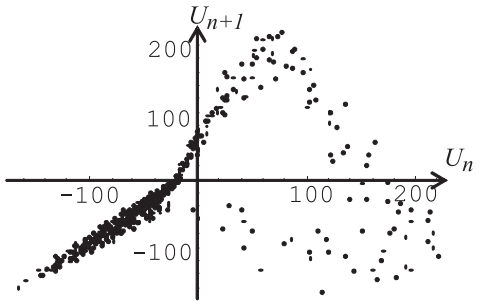


Fig. 17.13. Poincaré section for $u((n+1)T) = f(u(nT))$ for $k = 2.759$

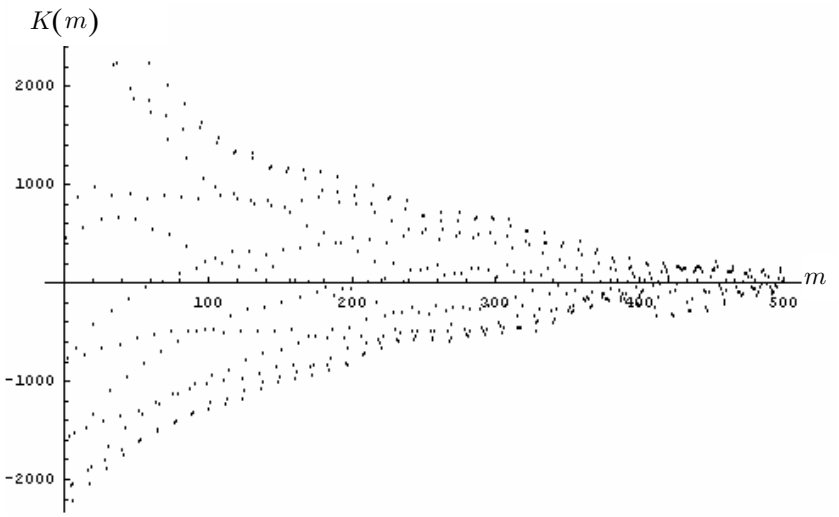


Fig. 17.14. Correlation function for $k = 2.759$

The correlation function for $k = 2.759$ is shown in Fig. 17.14. The convergence of the correlation function indicates that the process considered corresponds to a chaotic one.

A fractal dimension characterizes the number of degrees of freedom of points corresponding to this attractor. In calculating a process dimension it is convenient to use the definition of the dimension on the basis of the correlation function

$$C(r) = \frac{1}{N_M^2} \sum_i \sum_j H\left(r - \|x_i - x_j\|\right), \tag{17.16}$$

where N_M is the number of points, r is the radius of the circle in the point x_i, $\|\dots\|$ is the distance between the points x_i and x_j, H is the Heaviside step function. The correlation dimension is defined by

$$d_2 = \lim_{r \to 0} \frac{\ln C(r)}{\ln r}. \tag{17.17}$$

The graph of the $\ln C(r) = f(\ln r)$ function for the first case is presented in Fig. 17.15.

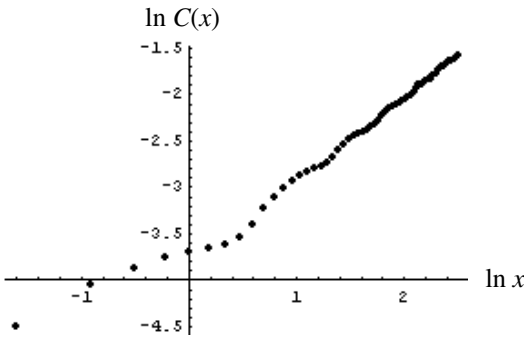


Fig. 17.15. Dependence $\ln C(r) = f(\ln r)$ for $k = 3.32$

For the second case the dependence $\ln C(r) = f(\ln r)$ for $k = 2.759$ is shown in Fig. 17.16

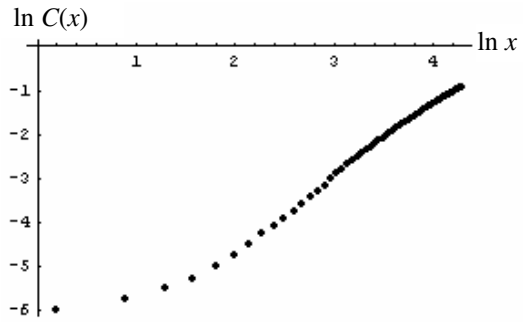


Fig. 17.16. Dependence $\ln C(r) = f(\ln r)$ for $k = 2.759$

Comparing Figs. 17.15 and 17.16 one can see the change in the value of the tangent in the second case is greater.

17.3. Analysis of processes in systems with a power conditioner

17.3.1. Mathematical model

Consider a mathematical model of the AC/AC conditioner. The topology of the circuit shown in Fig. 17.17 is based on the well-known Buck converter. In this system a load voltage depends linearly on the pulse width of signals controlling the bi-directional S switches.

This power line conditioner provides direct conversion of the AC voltage. No intermediate circuit is used for energy storage (e.g. a DC capacitor or large inductors). In the model, a voltage unbalance is generated by the connection of the resistor R_n in series with the source's internal R_{in} resistances.

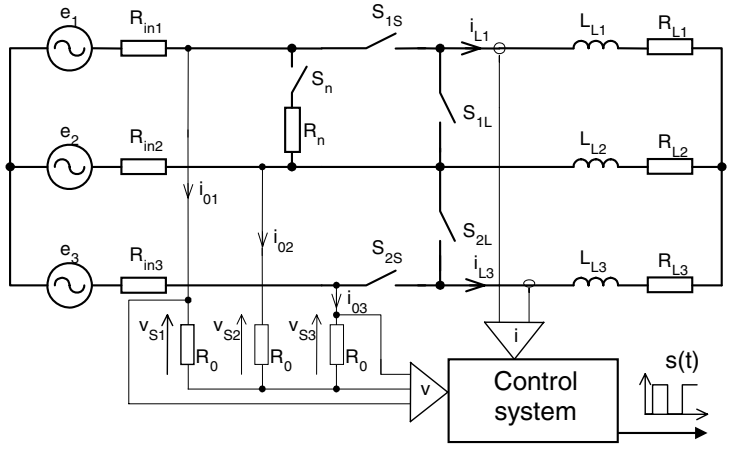


Fig. 17.17. System topology with a power conditioner

Assume that the switches are described by the RS model, and the inductors and load are linear. In this case the electromagnetic processes for the time interval, as the switches S_{1s} and S_{2s} are closed and the switches S_{1L} and S_{2L} are opened, are described by the matrix differential equation (Korotyeyev and Kasperek, 2004):

$$LL\frac{\mathrm{d}I}{\mathrm{d}t} = -A_{11}I - AI_{11}i - AU_{11}I_0 + E, \tag{17.18}$$

where

$$LL = \begin{vmatrix} L_{L1}+L_{L2} & L_{L2} \\ L_{L2} & L_{L2}+L_{L3} \end{vmatrix}, \qquad I = \begin{vmatrix} i_{L1} \\ i_{L3} \end{vmatrix}, \quad I_0 = \begin{vmatrix} i_{01} \\ i_{03} \end{vmatrix},$$

$$A_{11} = \begin{vmatrix} R_{L1}+R_{L2}+R_{in1}+R_{in2} & R_{L2}+R_{in2} \\ R_{L2}+R_{in2} & R_{L2}+R_{L3}+R_{in2}+R_{in3} \end{vmatrix}, \quad AI_{11} = \begin{vmatrix} R_{in1}+R_{in2} \\ R_{in2} \end{vmatrix},$$

$$AU_{11} = \begin{bmatrix} R_{in1}+R_{in2} & R_{in2} \\ R_{in2} & R_{in2}+R_{in3} \end{bmatrix}, \qquad E = \begin{bmatrix} e_1-e_2 \\ e_2-e_3 \end{bmatrix}.$$

Suppose that the switch S_n is in the on-state. Then the current is

$$i = i_{11} + RD_{11}I_0 + RP_{11}I, \tag{17.19}$$

where

$$i_{11} = \frac{e_1+e_2}{R_s}, \quad R_s = R_{in1}+R_{in2}+R_n, \quad RD_{11} = RP_{11} = \left[-\frac{R_{in1}+R_{in2}}{R_s} - \frac{R_{in2}}{R_s}\right].$$

The algebraic equation for the balancing circuit should be

$$E = AD_{11}I_0 + AN_{11}i + AP_{11}I, \tag{17.20}$$

where

$$AD_{11} = \begin{vmatrix} R_{in1} + R_{in2} + 2R_0 & R_{in2} + 2R_0 \\ R_{in2} + 2R_0 & R_{in2} + R_{in3} + 2R_0 \end{vmatrix}, \quad AN_{11} = AI_{11},$$

$$AP_{11} = \begin{vmatrix} R_{in1} + R_{in2} & R_{in2} \\ R_{in2} & R_{in2} + R_{in3} \end{vmatrix}.$$

The second time interval – when the switches S_{1L} and S_{2L} are closed and the switches S_{1s} and S_{2s} are opened – can be described by the matrix differential equation

$$LL\frac{\mathrm{d}I}{\mathrm{d}t} = -A_{22}I, \tag{17.21}$$

$$i = i_{11} + RD_{11}I_0, \tag{17.22}$$

$$E = AD_{11}I_0 + AN_{11}I, \tag{17.23}$$

where $A_{22} = \begin{vmatrix} R_{L1} + R_{L2} & R_{L2} \\ R_{L2} & R_{L2} + R_{L3} \end{vmatrix}$.
Combining the equations (17.18)–(17.20) and (17.21)–(17.23), we obtain

$$LL\frac{\mathrm{d}I}{\mathrm{d}t} = -A_{22}I - \gamma AP_{11}I - \gamma AU_{11}I_0 - \gamma AI_{11}i + \gamma E, \tag{17.24}$$

$$i = i_{11} + RD_{11}I_0 + \gamma RP_{11}I, \tag{17.25}$$

$$E = AD_{11}I_0 + AN_{11}i + \gamma AP_{11}I. \tag{17.26}$$

where γ is the switching function for the switches S_S and S_L.

The control system is presented in Fig. 17.18 (Korotyeyev and Kasperek, 2004). The input signals are the supply voltages u_s and the load currents i_L. For the calculation process the Clark-Park transformation is used. The fluctuation of the instantaneous power is used to generate the impulses $S(t)$ that control the switches of the main circuit.

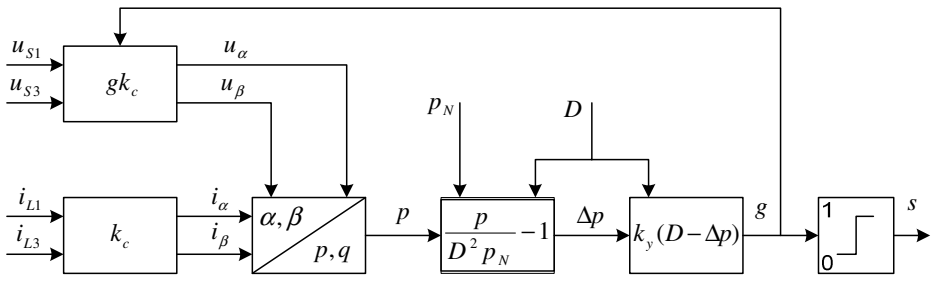


Fig. 17.18. Schematic diagram of the control system

The processes in the control system are described by the following: equations

$$\begin{gathered} I_\alpha = k_c I, \quad U_\alpha = k_c U, \quad P_\alpha = g U_\alpha^T I_\alpha, \quad \Delta p = \frac{P_\alpha}{D^2 p_n} - 1, \\ g = k_y(D - \Delta p), \quad u_{\text{com}} = g u_g(t), \quad \gamma = \gamma(u_{\text{com}}), \end{gathered} \tag{17.27}$$

where $I_\alpha = \begin{vmatrix} i_\alpha \\ i_\beta \end{vmatrix}$, $U_\alpha = \begin{vmatrix} u_\alpha \\ u_\beta \end{vmatrix}$, $U = \begin{vmatrix} u_{S1} \\ u_{S3} \end{vmatrix}$, $k_c = \begin{bmatrix} 1 & 0 \\ \frac{1}{\sqrt{3}} & \frac{2}{\sqrt{3}} \end{bmatrix}$, D and p_n are constancies, k_y is the gain, $u_g(t)$ is the voltage of the independent generator, U_α^T is the transposed vector.

Since $|AP_{11}| < |A_{22}|$, $|AU_{11}| < |A_{22}|$, $|AP_{11}I| < |AN_{11}i|$ and $|RP_{11}I| < i_{11}$, the equation (17.24) is simplified:

$$LL\frac{\mathrm{d}I}{\mathrm{d}t} = -A_{22}I + \gamma E_1, \tag{17.28}$$

where $E_1 = E - AI_{11}i_{11} - (AI_{11}RD_{11} + AU_{11})I_0$, $I_0 = RS^{-1}(E - AN_{11}i_{11})$, $RS = AD_{11} + AN_{11}RD_{11}$, and RS_{11}^{-1} is the inverse matrix.

The voltage U is defined by the expression

$$U = R_0 I_0. \tag{17.29}$$

Using the averaged state space method (Middlebrock and Cúk, 1976) we transform the equations (17.27) and (17.28) into the averaged state space equations

$$LL\frac{\mathrm{d}\bar{I}}{\mathrm{d}t} = -A_{22}\bar{I} + \mathrm{d}E_1, \tag{17.30}$$

$$\begin{gathered} U = R_0 I_0, \quad \bar{I}_\alpha = k_c \bar{I}, \quad U_\alpha = k_c U, \\ d = \frac{g}{U_{ag}}, \quad g = \frac{k_y(D+1)}{1 + k_y k_d U_\alpha^T \bar{I}_\alpha} \end{gathered} \tag{17.31}$$

where $\bar{I}$, $\bar{I}_\alpha$, are the averaged vectors, d ia the duty factor, U_{ag} is the amplitude of the generator voltage, $k_d = \frac{1}{D^2 p_n}$.

17.3.2. Determination of a steady-state process

As a result of the Clark-Park transform, the signal of the basic harmonic appears as a constant signal, and the unbalance as the second harmonic. We will find the solution of the equation set as a sum of those two constant components:

$$d = d_0 + d_2 \sin(2\omega t + \varphi), \tag{17.32}$$

where d_0 is the constant component, d_2 and φ represent the amplitude and the phase of the second harmonic.

Let us determine the solution of the set (17.30)–(17.31) using the Laplace transform:

$$\begin{aligned} I(p) = {} & \left(pI - LL^{-1}A_{22}\right)^{-1} LL^{-1} d_0 E(p) \\ & + d_2 (pI - LL^{-1}A_{22})^{-1} LL^{-1} \left[E(p) * \frac{p\sin\varphi + 2\omega\cos\varphi}{p^2 + (2\omega)^2} \right], \end{aligned} \tag{17.33}$$

where $I(p)$, $E(p)$ are transforms of the vectors $\bar{I}$ and E_1, I is the unity matrix, $(\cdot)^{-1}$ and LL^{-1} are inverse matrices, * represents convolution in the complex domain.

We can write the steady-state current as follows:

$$\bar{I} = I_0(t) + I_2(t), \tag{17.34}$$

where

$$I_0(t) = 2\mathrm{Re}\left[(j\omega I - LL^{-1}A_{22})^{-1}\right]\frac{d_0}{2j\omega}LL^{-1}E_n(j\omega)e^{j\omega t} \tag{17.35}$$

is determined with respect to the poles $p = \pm j\omega$ of the vector $E(p)$. In this expression $E_n(j\omega)$ denotes the numerator of the vector $E(j\omega)$.

Consider the second term in the solution (17.33). Calculating the convolution of the functions in (17.33), we obtain

$$E_c(p) = E(p) * \frac{p\sin\varphi + 2\omega\cos\varphi}{p^2 + (2\omega)^2} = \sum_{\substack{k=-1\\k\neq 0}}^{1} \frac{(p - kj\omega)\sin\varphi + 2\omega\cos\varphi}{(p + kj\omega)(p - kj3\omega)}\frac{E_n(kj\omega)}{k2j\omega}.$$

It follows from the expression that in the solution there are presented the first and third harmonics. Since the third harmonic does not participate in the energy conversion, from this expression we extract only the first harmonic:

$$E_{c1}(p) = \sum_{\substack{k=-1\\k\neq 0}}^{1} \frac{-2kj\omega\sin\varphi + 2\omega\cos\varphi}{(p + kj\omega)(-kj4\omega)}\frac{E_n(kj\omega)}{k2j\omega}.$$

Then the value of the vector of the current $I_2(t)$ in (17.34) is determined by the expression similar to (17.35):

$$I_2(t) \cong 2\mathrm{Re}\left[(j\omega I - LL^{-1}A_{22})^{-1}\frac{d_2}{2j\omega}LL^{-1}E_{c1n}(j\omega)e^{j\omega t}\right], \tag{17.36}$$

where $E_{c1n}(j\omega)$ is the numerator of the vector $E_{c1}(j\omega)$.

The instantaneous power

$$P_\alpha = U_\alpha^T \bar{I}_\alpha \tag{17.37}$$

can be written as follows:

$$P_\alpha = P_0 + P_2\sin(2\omega t + \psi), \tag{17.38}$$

where $P_0 = \frac{1}{T}\int_0^T P_\alpha \mathrm{d}t$ is the constant component of the instantaneous power, P_2 and ψ are the amplitude and the phase of the second harmonic.

Using the Laplace transform, we calculate the power (17.37) with the help of convolution:

$$\begin{aligned} P_\alpha(p) = U_\alpha^T(p) * I_\alpha(p) &= \sum_{\substack{k=-1\\k\neq 0}}^{1} \frac{U_{\alpha n}^T(kj\omega)I_{\alpha n}(p - jk\omega)}{2kj\omega\left[(p - jk\omega)^2 + \omega^2\right]} \\ &= \sum_{\substack{k=-1\\k\neq 0}}^{1} \frac{U_{\alpha n}^T(kj\omega)I_{\alpha n}(p - jk\omega)}{2kj\omega p(p - 2jk\omega)}, \end{aligned} \tag{17.39}$$

where $I_{\alpha n}(kj\omega)$, $U_{\alpha n}^T(kj\omega)$ are numerators of the vectors $I_\alpha(kj\omega)$ and $U_\alpha^T(kj\omega)$.

As we have expected, from the expression (17.39) it follows that in the solution there exist the constant component and the second harmonic. We determine the constant component using the theorem of the final value of the Laplace transform:

$$P_0 = \lim_{p\to 0} pP_\alpha(p) = \sum_{\substack{k=-1\\k\neq 0}}^{1} \frac{U_{\alpha n}^T(kj\omega)I_{\alpha n}(-jk\omega)}{4k^2\omega^2}.$$

The transformation of the second harmonic is determined by the calculation of residues:

$$P_2(p) = \sum_{\substack{k=-1\\k\neq 0}}^{1} \frac{\lim\limits_{p\to 2jk\omega}(p-j2k\omega)P_\alpha(p)}{p-j2k\omega} = P_2\frac{p\sin\psi + 2\omega\cos\psi}{p^2+4\omega^2}.$$

In this expression, P_2 and ψ are determined as a result of the calculation of the limit of the transform $P_\alpha(p)$.

Taking into account that the power P_2 is small in comparison to P_0, we expand (17.31) in Taylor series about P_0. Then

$$d = \frac{k_y(D+1)}{(1+k_yk_dP_0)U_{ag}} - \frac{k_y(D+1)k_yk_dP_2\sin(2\omega t+\psi)}{(1+k_yk_dP_0)^2U_{ag}}.$$

Using the expression for the duty factor (17.32), we extract the equation for the constant component:

$$d_0 = \frac{k_y(D+1)}{(1+k_yk_dP_0)U_{ag}}, \tag{17.40}$$

and the equations for the amplitude d_2 and the phase φ are the following:

$$d_2 = \frac{k_y(D+1)k_yk_dP_2}{(1+k_yk_dP_0)^2U_{ag}}, \tag{17.41}$$

$$\psi = \varphi + \pi. \tag{17.42}$$

It is convenient to transform the equation (17.40) as follows:

$$d_0 = \frac{k_y(D+1)}{(1+k_yk_dd_0\tilde{P}_0)U_{ag}},$$

where $\tilde{P}_0 = \frac{P_0}{d_0}$, $\tilde{P}_0 = \frac{1}{T}\int\limits_0^T U_\alpha^T I_{0\alpha}\mathrm{d}t$, $I_{0\alpha} = k_cI_0(t)$.

Then the constant component is determined as a result of the solution of the square equation

$$d_0U_{ag}(1+k_yk_dd_0\tilde{P}_0) = k_y(D+1). \tag{17.43}$$

To determine d_0, d_2 φ, first from the equation (17.43) we calculate d_0 and then we solve the equations (17.41) and (17.42) simultaneously.

17.3.3. Calculation of steady-state processes

Let us calculate a steady-state process for the following values of parameters: $E = 310$ V, $D = 0.5$, $k_u = 0.0326$, $k_i = 3.256$, $P_n = 97.5$ W, $U_{ag} = 1$ V, $R_{in} = 1\ \Omega$, $R_n = 10\ \Omega$, $R_L = 100\ \Omega$, $L_L = 75$ mH, $T = 20 \times 10^{-3}$ s (the switching period of the converter). The coefficients k_u and k_i are used for the calculation of the vectors $\bar{I}_\alpha = k_i k_c \bar{I}$, and $U_\alpha = k_u k_c U$.

Temporal changes in the duty factor calculated by the numerical and discussed methods are shown in Fig. 17.19.

Figure 17.20(a) presents the currents in the load calculated on the basis of the doscussed and numerical methods. The above mathematical model has been verified by PSpice simulation tests. As an example, the same load currents obtained by simulation are presented in Fig. 17.20(b). The instantaneous power P and the duty factor d obtained from the PSpice simulation are presented in Fig. 17.21.
The time waveforms of all the above signals are similar.

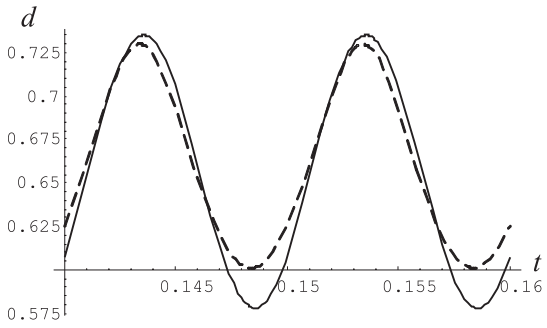


Fig. 17.19. Temporal changes in the duty factor for the numerical and discussed methods. The continuous line corresponds to the discussed method, the dashed line to the numerical one

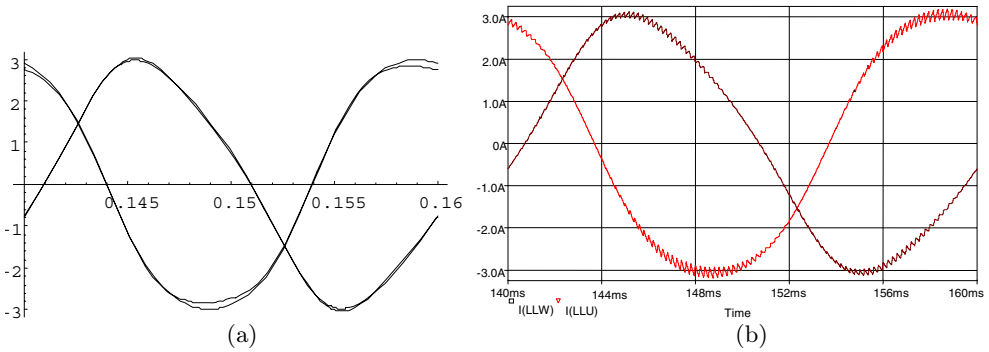


Fig. 17.20. Load currents i_{L1} and i_{L3}: (a) calculated on the basis of the discussed and the numerical method, (b) obtained from PSpice simulation

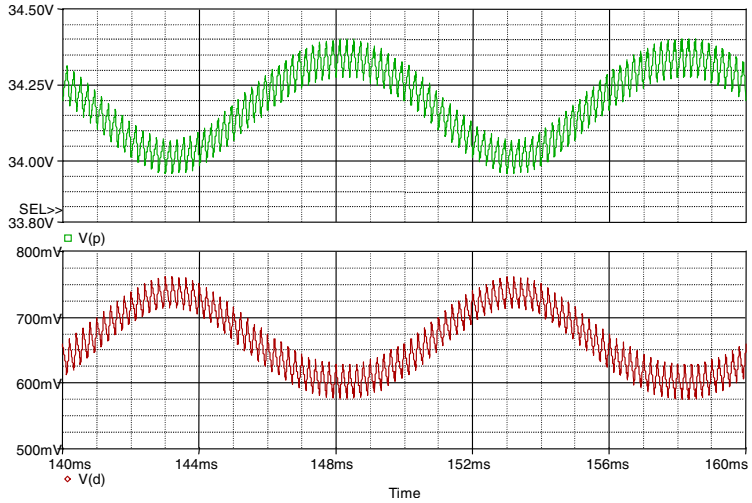


Fig. 17.21. Instantaneous power P and the duty factor d obtained from PSpice simulation

17.4. Conclusions

A model DC converter has been presented. The analysis of processes in the closed-loop system depending on the gain has been performed. It has been shown by process modelling that the comparison of phase portraits, Poincaré sections, correlation functions and fractal dimensions permits process identification in the DC/DC converter.

The concept of the control method based on instantaneous power observation is presented. The model of the AC line conditioner has been established by nonlinear, non-stationary equations. These have been solved under the assumption that the solution can be presented as a superposition of the basic component and the second harmonics. Simulation test results of the PSpice model of the converter and the results of calculation for the discussed method and the numerical method have been compared.

References

Banerjee S. and Verghese G.C. (2001): *Nonlinear Phenomena in Power Electronics: Attractors, Bifurcations, Chaos and Nonlinear Control.* — New York: IEEE Press.

Jang D-H. and Choe G-H. (1995): *Improvement of input power factor in AC choppers using asymmetrical PWM technique.* — IEEE Trans. Ind. Electronics, Vol. 42, No. 2, (CD-ROM).

Kasperek R. (2003): *Control algorithms of the PWM AC line conditioners under unbalanced input voltage.* — Proc. Int. Conf. *Actual Problems of Electrical Drives and Industry Automation*, Estonia, Tallin, (CD-ROM).

Korotyeyev I.Y. and Kasperek R. (2004): *Mathematical model of the AC chopper operating under line voltage unbalance. Concept of control method based on instantaneous power*

observation. — Proc. Int. Conf. *European Power Electronics – Power Electronics and Motion Control*, Riga, Latvia, (CD-ROM).

Korotyeyev I.Ye and Klytta M. (2002): *Stability analysis of DC/DC converters*. — Technical Electrodynamics, Power Electronics and Energy Efficiency, No. 1, pp. 51–54.

Lefeuvre T., Meynard P. and Viarogue P. (2001): *Fast line voltage conditioners using a new PWM AC chopper technology*. — Proc. European Conf. *Power Electronics and Applications*, Graz, Austria, (CD-ROM).

Middlebrock R.D. and Cúk S. (1976): *A general unified approach to modelling switching converter power stages*. — Proc. IEEE Conf. *Power Electronics Specialists*, Cleveland, Ohio, USA, pp. 18–34.

Rozenwasser Ye.N. and Yusupov R.M. (1981): *Control Systems' Sensitivity*. — Moscow: Nauka, (in Russian).

Strzelecky R., Korotyeyev I.Ye. and Zhuykov V.Ya. (2001): *Chaotic Processes in Systems of Power Electronics*. — Kiev: Avers, (in Russian).

Veszpremi K. and Hunyar M. (2000): *New application fields of the PWM IGBT AC chopper*. — IEEE Power Electronics and Variable Speed Drives, pp. 18–19.

Zhuykov V.Ya. and Korotyeyev I.Ye. (2001): *Research of process in the direct current boost stabilizer*. — Technical Electrodynamics, Up to Date Electrical Engineering Problems, No. 3, pp. 89–93, (in Russian).

Zhuykov V.Ya., Korotyeyev I.Ye., Ryabenky V.M., Pavlov G.V., Racek V., Vegg A. and Liptak N.A. (1989): *Closed-up Systems of Electrical Power Transform*. — Kiev: Technika; Bratislava: Alpha.

Chapter 18

ELECTROMAGNETIC COMPATIBILITY IN POWER ELECTRONICS

Adam KEMPSKI*, Robert SMOLEŃSKI*, Emil KOT*

18.1. Introduction

According to the Council Directive 89/336/EEC of 3 May 1989, "ElectroMagnetic Compatibility (EMC) means the ability of a device, unit of equipment or system to function satisfactorily in its electromagnetic environment without introducing intolerable electromagnetic disturbances to anything in that environment". Problems connected with electromagnetic compatibility become more important due to an increasing amount of susceptible equipment that generates disturbances of a high level.

Due to a tendency towards decreasing the dimensions of equipment and greater flexibility of energy conversion, power electronic converters with pulse modulation are becoming commonly used (Mohan *et al.*, 1995). However, these benefits have been counterbalanced by many unwanted side effects, especially a high level of ElectroMagnetic Interferences (EMI) (Jin *et al.*, 2004; Kempski, 2003; Palis *et al.*, 1997; Teulgins *et al.*, 1997; Tihanyi, 1995; Tse *et al.*, 2000; Weston, 1991).

The problem has been exacerbated with the rapid spread of a new generation of fast switching power semiconductors, such as the IGBT (Insulated Gate Bipolar Transistor). The nearly square-wave shape of voltage/current waveforms resulting from the switching states of the power converter caused a wide-band, high level spectrum of generated EMI (Williams and Armstrong, 2000).

Figure 18.1 schematically shows intentional and parasitic electromagnetic processes in systems containing power electronic converters (Kempski, 2005).

Intentional processes comprise power conversion processes in a low frequency band and control processes in a higher frequency range. Due to the wide frequency range of useful electromagnetic processes in power converters, unwanted EMI spectra spreading over the frequency range from DC to radiated emissions should be expected.

* Institute of Electrical Engineering
e-mails: {A.Kempski, R.Smolenski, E.Kot}@iee.uz.zgora.pl

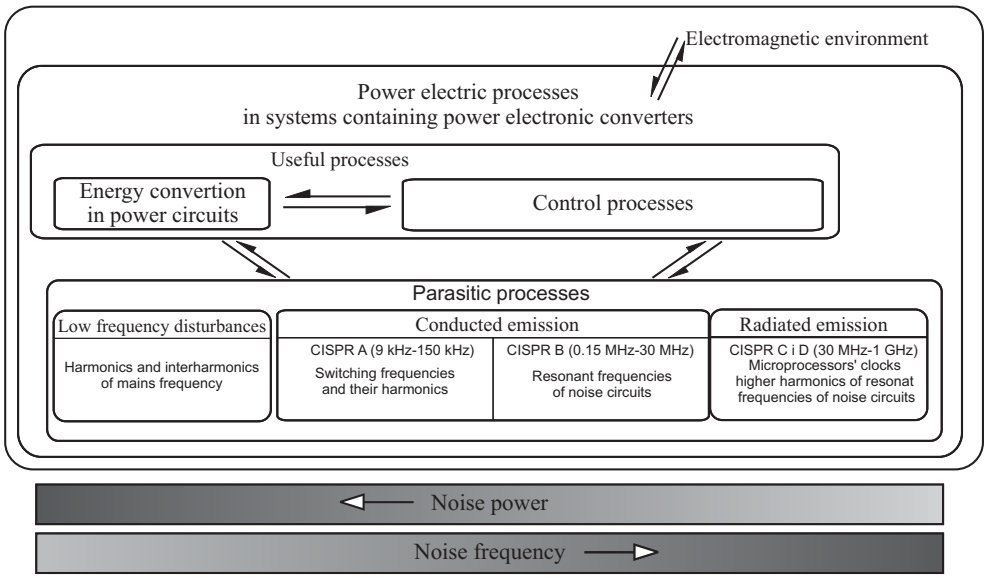


Fig. 18.1. Intentional and parasitic electromagnetic processes in systems with power electronic converters

Traditionally, EMC regulations covered from 9 kHz to 1GHz. This frequency range is divided into two major classes of disturbances: the conducted emission (9 kHz–30 MHz) and the radiated emission (30 MHz–1 GHz). Both conducted and radiated emissions are subdivided into CISPR (Comité international spécial des perturbations radioélectriques) bands. Other standards associated more with power quality than EMC concern low frequency phenomena (DC-50th harmonic of mains supply). It is evident that for verifying whether equipment complies with the standard limits, it is necessary to have at one's disposal a sophisticated and delicate EMC laboratory system. Investigations in each of the CISPR bands require specific measuring methods and test equipment stipulated in international standards.

The experimental results presented in this chapter have been obtained in the EMC Lab at the Institute of Electrical Engineering of the University of Zielona Góra. There are two fully automated systems at our disposal:

- *system for electromagnetic emission measurements* – fully compliant with CISPR-16. The EMI test receiver (9 kHz–2,75 GHz) permits measurements of conducted emissions (9 kHz–30 MHz) using the LISN (Line Impedance Stabilization Network) as well as radiated emissions (30 MHz–1 GHz) using the GTEM (Gigahertz Transverse ElectroMagnetic) cell of septum height 950 mm,
- *system for electromagnetic immunity measurements* – a set of equipment which permits immunity measurements in both conducted and radiated frequency ranges, in full agreement with international standards.

In the EMC Lab there can be performed other conducted immunity measurements (e.g. flicker, dips, short terms interruptions, ElectroStatic Discharges (ESD))

as well as measurements of electromagnetic field levels over a wide frequency range (up to 5 GHz).

The majority of the presented results concerns PWM (Pulse Width Modulation) adjustable speed drives. Power converters have experienced an unprecedented growth in industrial drives and the power electronics area over the past dozen years. However, with the introduction of modern inverter technology, several authors have reported an increased number of serious EMC problems. A lot of theoretical and experimental research has been done to resolve these problems (von Jouanne *et al.*, 1998; Kempski *et al.*, 2000; Ran *et al.*, 1998; Skibiński *et al.*, 1999), but many aspects are still under discussion (Jin and Weiming, 2004; Kempski *et al.*, 2002; Qu and Chen, 2002; Shen *et al.*, 2004).

The significance and diversity of EMC aspects in PWM power electronic converter drives are the reasons for choosing this area as an exemplification of EMC problems in power electronics.

18.2. Conducted EMI in power electronic systems

The limits for conducted electromagnetic emission generated by electric equipment which are stated in the standards concern only EMI that can affect mains. However, in the case of a system containing a power converter, conducted EMI on both sides of the converter should be taken into account. In order to describe EMI current paths and the appropriate application of emission reduction techniques, the method of splitting EMI current into a Common Mode (CM) and a Differential Mode (DM) is commonly used. The CM noise is a type of EMI induced on signals with respect to a reference ground. The remaining of the total conducted EMI is defined as the DM noise. While the CM/DM separation is well defined and understood for the single-phase or the DC system, the same cannot be said of three-phase converter systems, common for general-purpose adjustable speed drives, and there is no universal CM/DM definition (Shen *et al.*, 2004). However, splitting into the CM/DM in three-phase systems is still possible if a symmetrical, linear and time invariant three phase-system is considered.

Figure 18.2 schematically shows CM/DM current paths in three-phase systems for step excitation in the phase B. Lumped LC elements represent parasitic and residual parameters (that are in fact distributed) of EMI noise paths. Using some orthogonal transformations [e.g. γ, δ, 0] and the Laplace transform, it is possible to transform the CM and DM to one-phase circuits (Kempski, 2005).

The main EMC problem is CM noise because of its flow through large "ground loop" and its coupling by means of the common impedance of the grounding arrangement. Consequently (Williams and Armstrong, 2000),

- the effects of CM noise are difficult to predict and control,
- it can change with uncontrolled structural changes,
- it can contaminate different unrelated equipment.

This means that CM noise has a potential to cause a malfunction of nearby systems (external compatibility) and unwanted effects inside the converter-load arrangement (internal compatibility).

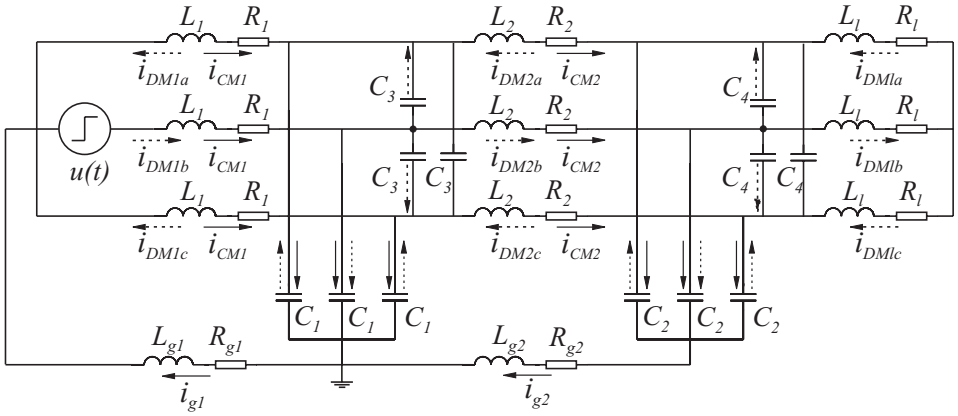


Fig. 18.2. CM/DM current paths in three-phase systems

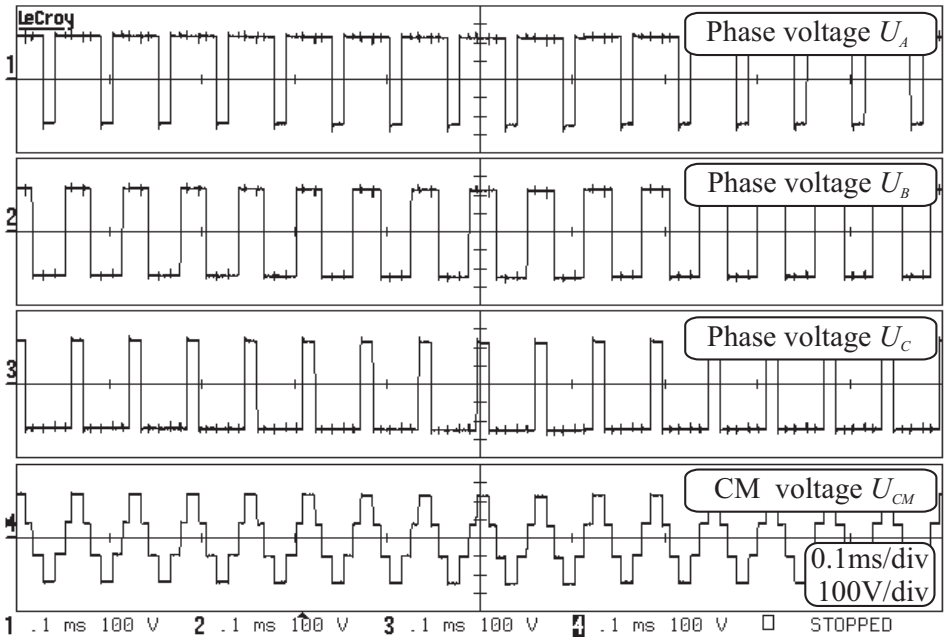


Fig. 18.3. Phase voltages and common mode voltage at the output of the inverter

Generally, the source of CM interferences can be expressed as the sum of phase voltages with respect to ground in the CM current one-phase equivalent circuit. Figure 18.3 shows phase voltages and CM voltage at the output of a three-phase inverter for PWM sinusoidal modulation. The CM voltage is a staircase function with the step $\pm 1/3 U_d$ (U_d – DC link voltage).

Steep pulses of the CM voltage excite parasitic capacitive couplings in drive system components, especially in the cable/load arrangement. The transient line-to-ground capacitive current has an impulse-like waveform with HF oscillation. The

frequency of the oscillatory mode is determined by the values of residual and parasitic parameters of the CM current path. Since the line-to-ground system impedance is predominantly capacitive, then the magnitude of CM currents in general terms is proportional to $\mathrm{d}v/\mathrm{d}t$ of voltage steps (typically a few kV/μs), Fig. 18.4.

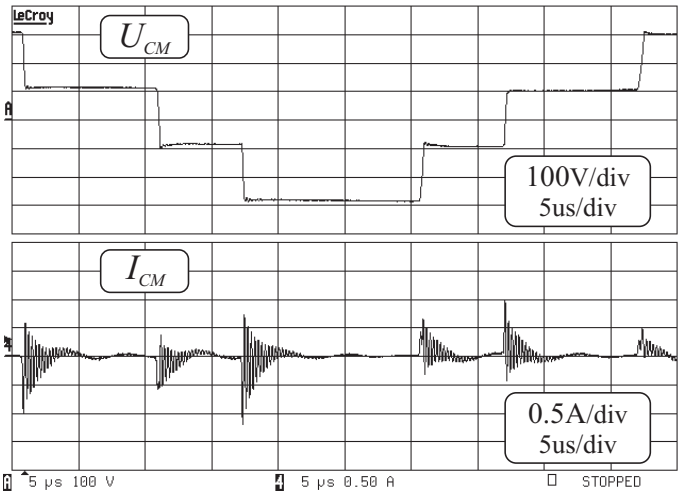


Fig. 18.4. Phase voltage at the output of the inverter and the CM current in a motor PE wire

18.3. Electromagnetic interferences in power converter drives

18.3.1. EMI currents in a PWM two-quadrant inverter drive

The tendency to increase inverter switching frequency has improved the performance of drives in such features as easy operation and control, ultimate dynamic, low switching losses and relatively low harmonic content in the load current. However, the higher carrier frequency and very short switching events have introduced several problems in systems originating from a step output voltage change:

- overvoltages,
- increase of conducted and radiated electromagnetic disturbances,
- unwanted tripping of protective equipment,
- discharging bearing currents.

These problems are related to both internal and external EMC of a system. Additionally, electromagnetic disturbances can be hazardous to electromagnetic environment and people's health, both directly and indirectly because electronic technology is increasingly used in safety-related applications. Consequently, errors and misoperations of electronic devices due to inadequate EMC can result in hazardous situations with an increased risk of harm to people's health and safety (so-called functional safety).

In order to trace the real CM current paths, the measurements have been taken in a typical PWM drive system, which had been supplied and grounded in a manner

commonly used in practice. The experimental arrangement and measuring points are depicted in the Fig. 18.5.

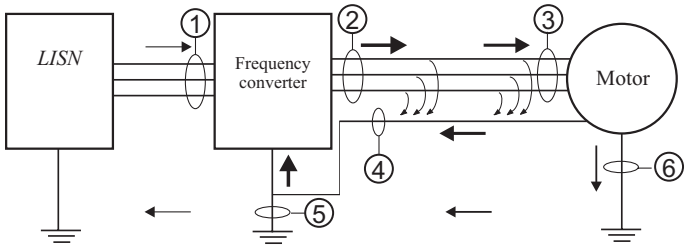


Fig. 18.5. CM current flow in the experimental arrangement

We have tested a 2-pole 1.5 kW induction motor fed by a commercially available industrial inverter. All measurements have been done using the wideband current probes with linear frequency range up to 50 MHz.

Figure 18.6 shows the experimental results of a CM current passage through the system, and the CM current paths are marked in Fig. 18.5 by means of arrows of different thickness.

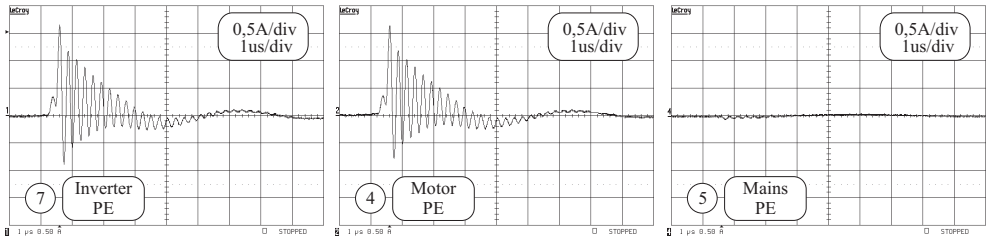


Fig. 18.6. Passage of common mode currents in the system (experimental results)

The common mode currents, which are generated by switching states of the inverter, close partly inside the cable. Next, the CM currents split according to the proportion of HF impedance of a PE cable wire (or shield) and HF impedance of the grounding arrangement between the grounding points of the inverter and the motor. The main return path for the CM currents passes via the heatsink to DC link capacitance. The CM current causes a CM voltage drop on this capacitance. In a blocking state of diodes of the rectifier only a small part of this current flows through the inverter supply arrangement, Fig. 18.7 (note different scales in the figure).

In the conduction state, this voltage drop causes oscillation of small amplitude and relatively low frequency in a closed loop consisting of a DC link to heatsink capacitance and the resultant inductance of the mains (or the LISN), the cable and the input filter. These conclusions, especially concerning the role of the heatsink to DC link capacitance, have key importance for the solving of EMC problems in two-quadrant inverter drives. The presented experimental results have been confirmed using the simulation method (Kempski *et al.*, 2002).

Figure 18.8 shows the spectrum of conducted EMI measured according to the EN 61800-3 standard (EMC product standard for PDS).

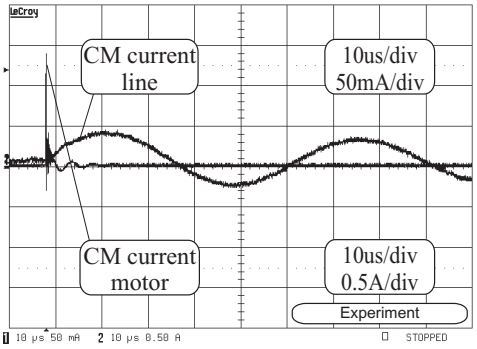


Fig. 18.7. CM currents on the line and motor side of the inverter

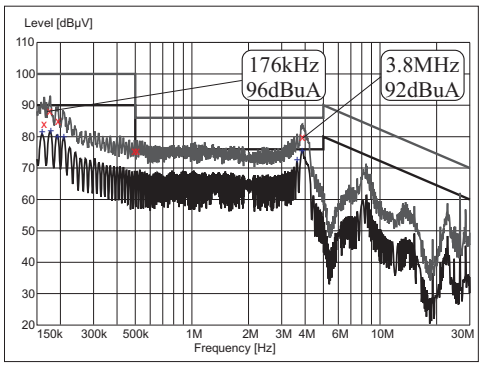


Fig. 18.8. Conducted emission spectrum according to EN 61800-3

EMI noise is located below limits (for both quasi-peak and average detectors) in the whole frequency band. This means that the drive is externally compatible in the conducted emission frequency band. However, it should not mean that internal compatibility is assured at the same time.

Figure 18.9 shows the spectra of CM currents on both sides of the converter (additional measurement not required by the standards).

The main oscillatory modes are created on the motor side of the converter. The same modes appear on the line side, but their magnitudes are significantly lower. It means that EMC problems such as overvoltages in the converter-load arrangement, bearing currents and possible misoperations of the converter are still unsolved.

18.3.2. EMI currents in a PWM four-quadrant inverter drive

Four-quadrant PWM frequency converters are increasingly being used in adjustable speed drives of high mechanical performance. There are many advantages of these converters, such as active modulation to reduce the harmonic content and DTC (direct torque control) that ensures quick responses to the load.

The main circuit of this converter comprises two IGBT three-phase bridges and an intermediate circuit allowing two-way energy flow and four-quadrant operation,

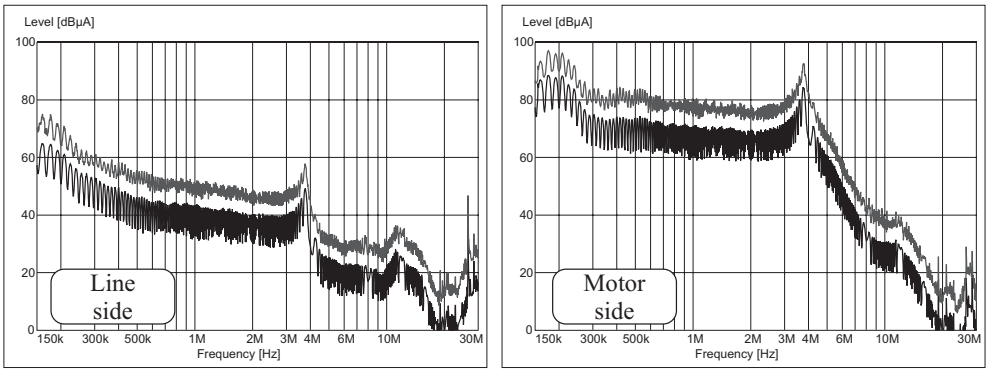


Fig. 18.9. CM currents spectra on both sides of the converter

Fig. 18.10. A line side converter is an active IGBT rectifier, and an inverter provides AC power for the motor in motoring quadrants.

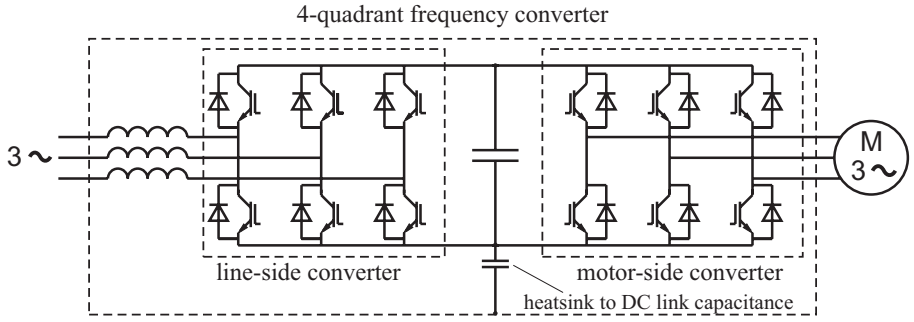


Fig. 18.10. Main circuit diagram of the 4-quadrant inverter drive system

The suppression of EMI noise is very important in the case of a 4-quadrant AC drive because of fast switching of the active rectifier on the line side and stringent limits included in the international standard EN 61800-3.

The measurements have been done in the system presented in Fig. 18.10. We have tested a 2-pole, 10 kW induction motor fed by a typical industrial four-quadrant frequency converter supplied via the LISN. All measurements have been done using current probes. The measuring points are depicted in Fig. 18.11.

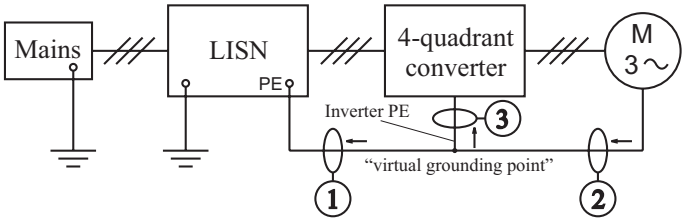


Fig. 18.11. Experimental arrangement

Figure 18.12 shows the results of measurements which have been done in the system consisting of a LISN and EMI test receiver, in the frequency range specified in the EN 61800-3 standard for a PDS (and additionally in a CISPR A frequency band), Fig. 18.13.

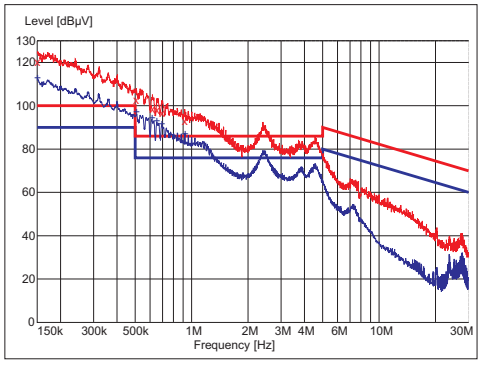


Fig. 18.12. Conducted EMI spectrum (drive without filters)

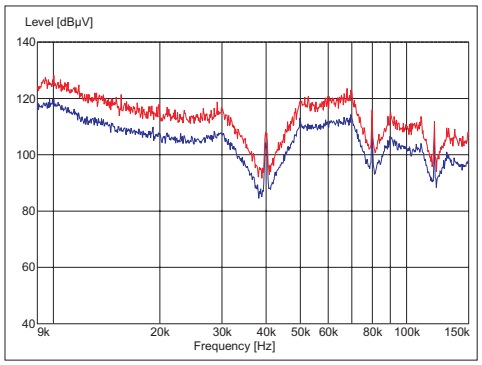


Fig. 18.13. Conducted EMI spectrum (CISPR A)

In the low frequency part of the spectrum (CISPR A) there have been observed repeatable changes at frequencies 40 kHz, 80 kHz, ... We have identified them with the time of the synchronized impulse of transistor switching (25 μs) and its harmonics. The envelope of this spectrum is characteristic for a damped sinewave pulse at the frequency of about 70 kHz. In the high frequency part of the spectrum (CISPR B), the level of noise is very high and significantly exceeds the limits specified in the standard for frequencies up to 5 MHz. There have been observed oscillation modes of EMI current waveforms at frequencies 2.5 MHz, 3.8 MHz and 4.7 MHz.

In order to establish the significance of the CM component in the EMI spectrum, the measurements using a current probe have been done in PE wires on the line and motor side, Fig. 18.14.

In CM current spectra, it is possible to observe the main oscillation modes of the EMI spectrum. The low frequency component prevails on the line side, whereas on the motor side the higher frequency components are more significant.

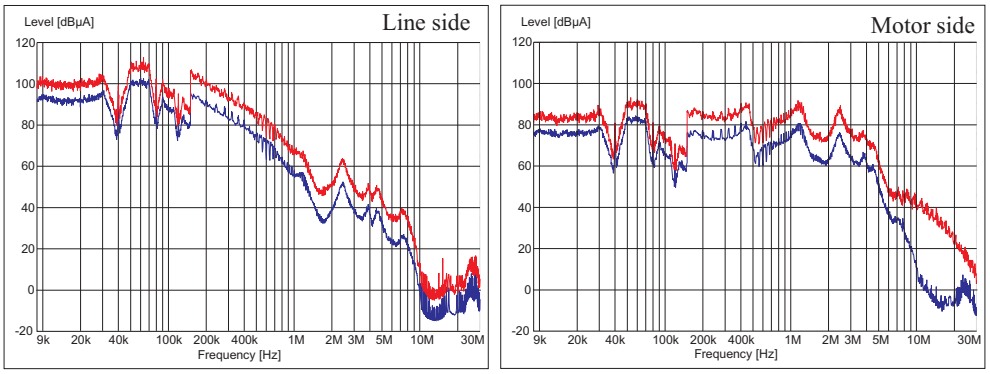


Fig. 18.14. Spectra of CM currents on the line and motor side

The measurements in the frequency domain seems to indicate that there are two CM voltage sources in a four-quadrant AC drive system: one on the line side of the converter and the other on the motor side.

The measurements in the time domain have been done in order to confirm this presumption and the location of the sources and paths of CM currents. The additional PE wire marked as "inverter PE" in Fig. 18.11 makes this possible.

Figure 18.15 shows the passage of the CM current through the "virtual grounding point". The arrows in Fig. 18.11 depict real CM current paths in the system.

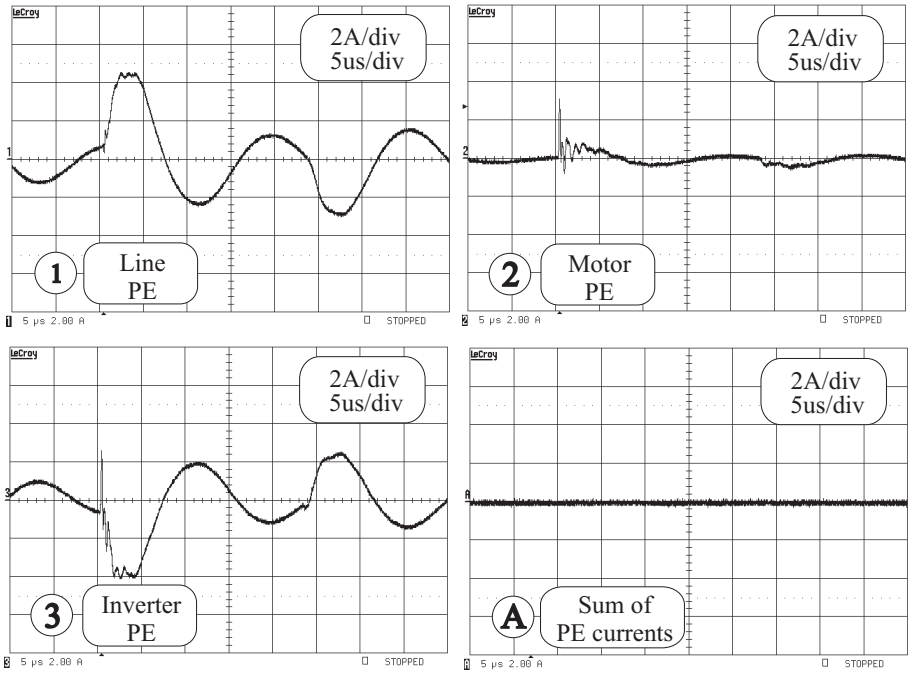


Fig. 18.15. Passage of common mode currents through the "virtual grounding point"

The CM current on the line side of the converter has a damped oscillation form with an amplitude up to 10 A, a frequency of about 70 kHz and an RMS value which exceeds 1 A. This current emerges at each switching instant of the controlled rectifier due to the presence of the heat-sink to DC link capacitance. On its way back to the source, the CM current flows through the mains and input filter impedances, causes voltage drops on them, and creates a CM voltage at the input terminals (Kempski *et al.*, 2003).

It is possible to identify high frequency oscillation modes from the spectrum of the CM current in the CM current waveform on the motor side. These components are superimposed by the low frequency oscillation due to the voltage drop across the heatsink to DC link capacitance caused by the CM current on the line side. The amplitude of the CM current can reach a value of about 4 A.

The CM currents on both sides of the converter form two almost independent loops. The heatsink to DC link capacitance is a common part of these two loops. The shape of the CM current in the inverter PE wire confirms the essential role of the heatsink to DC link capacitance for both creating and conducting CM noise.

18.4. Special EMC problems in inverter-fed drives

18.4.1. Bearing currents

In recent years, Electric Discharge Machining (EDM) bearing currents have been found in the industry the main cause of premature bearing damage in PWM (Pulse Width Modulation) inverter fed drives. The energy transferred at the moment of a breakdown can induce an accelerated bearing damage by pitting bearing races.

The basic reason for bearing currents in a PWM inverter drive is the common mode (CM) voltage. The steps of voltage pulses can stimulate capacitive couplings between the stator windings, the stator frame and the rotor. The thin oil film in a bearing acts as a capacitor and allows the voltage to build up on the motor shaft. The shaft voltage can be expressed by the voltage divider of the internal parasitic capacitance of a motor (Macdonald and Grey, 1999):

$$\frac{U_S}{U_N} = \frac{C_{SR}}{C_{SR} + C_B + C_{AG}}, \tag{18.1}$$

where U_S is the shaft voltage, U_N is the voltage of the stator windings neutral point, C_{SR} is the capacitance between the stator windings and the rotor, C_B is the bearing capacitance (between the balls and races of bearing), C_{AG} is the air gap capacitance.

The experimental results have shown that even in small machines the shaft voltage can reach a value of a dozen volts. The waveform of the shaft voltage almost perfectly maps the waveform of the CM voltage in the stator winding neutral point in accordance with the divider proportion, Fig. 18.16.

If the shaft voltage exceeds the critical value of a bearing threshold voltage that is required for a spark breakdown of the oil insulating film in the bearings, the shaft will be unloaded in the form of electric discharge. It is the so-called Electric Discharge Machining (EDM) current. The destructive EDM current is an impulse with the

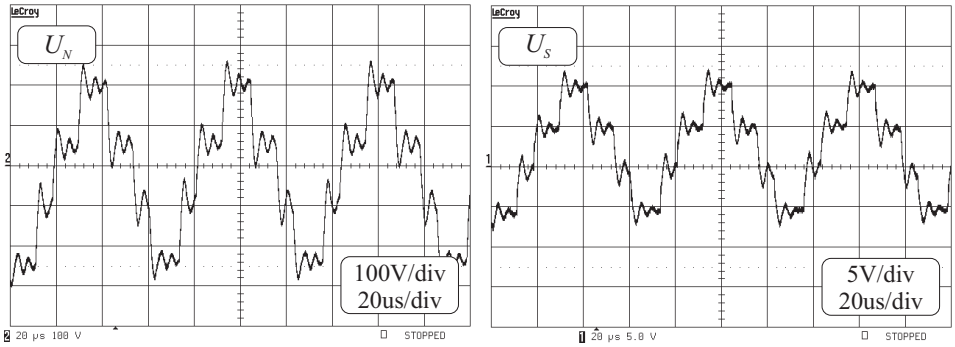


Fig. 18.16. Common mode voltage and the shaft voltage

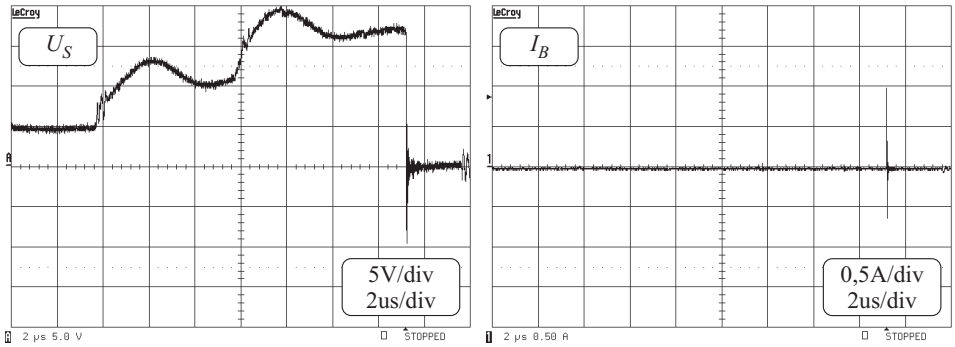


Fig. 18.17. Shaft voltage and the discharging bearing current

risetime of nanoseconds (Kempski, 2001). A peak of the EDM current can reach the value of a few amperes, Fig. 18.17.

It has been decided to treat the amplitudes of EDM currents and the moments of a breakdown as random variables because of the delay time occurring in well-known mechanisms of an electrical breakdown. The measuring instrument allows triggering and storing a large amount of data (above 6000) for very fast events with a high-resolution spread out over long periods of time. The procedure for data collecting was presented in our previous work (Smoleński *et al.*, 2002).

It is possible to obtain and analyze a very large set of discharging events. This phenomenon depends on many various factors caused by the modulation strategy of an inverter, such as carrier frequency, inverter output frequency etc. It also depends on the physical proprieties of the insulating oil film and the hydromechanical bearing behavior. A large measuring data base allows observing the influenced factors and takes them into consideration to develop the most appropriate statistical model.

Figure 18.18 shows an example of the dependence of awaiting time for puncture on electrical factors introduced by an inverter (in this instance, inverter output frequency). The selection of inverter output frequency, which equals mains frequency, causes the superimposing of rectifier and inverter harmonics, which in turn results in the concentration of EDM samples in this region. In this special case deviations

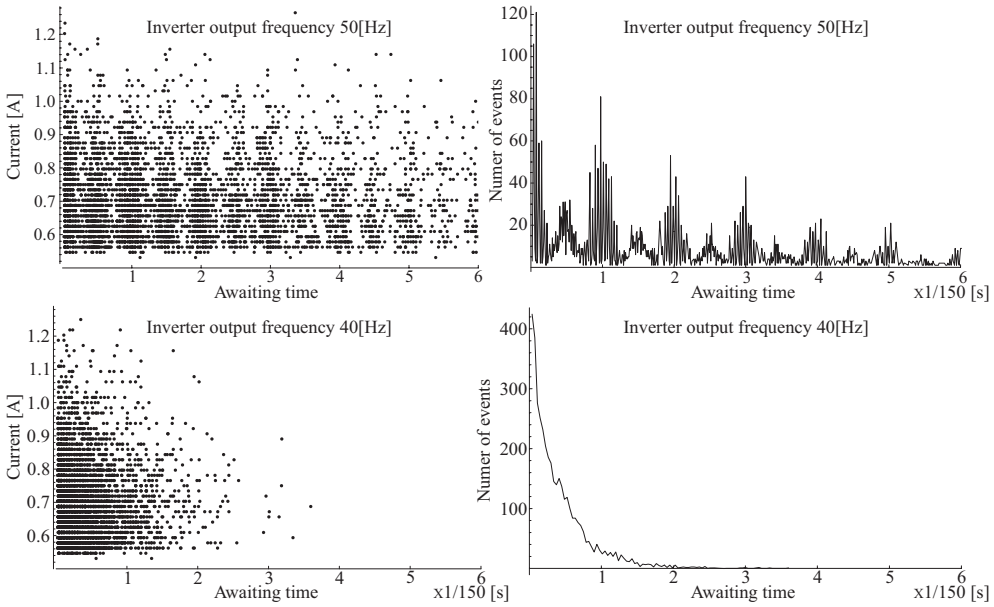


Fig. 18.18. Distribution of EDM currents and histograms of awaiting times to puncture

from exponential distribution selected to describe awaiting time for puncture are most noticeable.

We have assumed a hypothesis that the histogram of awaiting times for puncture (τ) can be approximated by exponential distributions (depicted by the continuous line in Fig. 18.19) (Kempski *et al.*, 2005; Smoleński *et al.*, 2002; Smoleński, 2003). The parameter of the distribution has been estimated using the least square method. The measuring data have passed the test of the goodness of fit with exponential distribution, for all output frequencies, even in the case when inverter output frequency equaled mains frequency,

$$f(\tau) = \lambda e^{-\lambda\tau}. \tag{18.2}$$

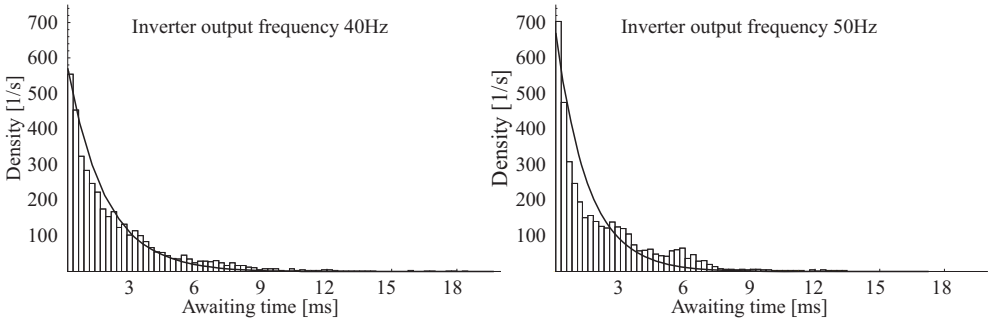


Fig. 18.19. Histograms of awaiting times to puncture

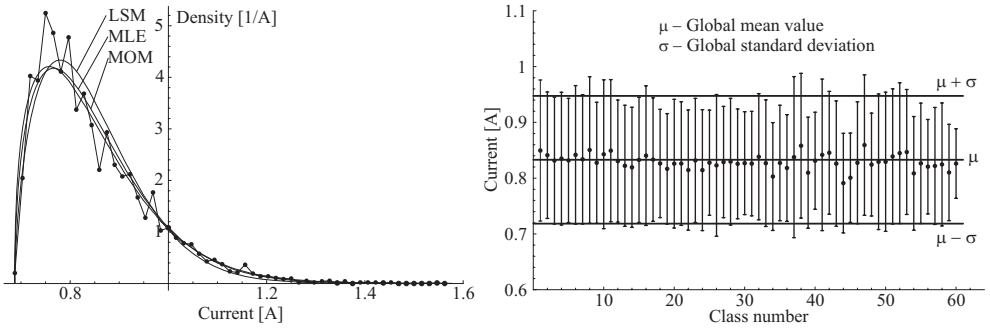


Fig. 18.20. Distribution of the amplitudes of EDM currents and the mean values of classes with standard deviations

In our previous works we have selected truncated normal distribution to describe the amplitudes of EDM currents. The measuring data have passed the test of the goodness of fit with this distribution. However, we notice now that the location of points in the horizontal lines in Fig. 18.18 is determined by the resolution of an eight bit digital oscilloscope. This additional piece of information allows developing another method of the division of data into classes. The constructed histogram shows consistence with the Weibull distribution, which is commonly used to describe discharging phenomena,

$$f\left(a\right) = e^{-\left(\frac{a-A_0}{\eta}\right)^{\beta}} \left(a - A_0\right)^{\beta-1} \beta\eta^{-\beta}, \quad a > A_0, \tag{18.3}$$

where a represents the amplitudes of EDM currents, A_0, β, η, are the parameters of the Weibull distribution.

Figure 18.20 shows a histogram of the amplitudes of EDM currents and theoretical Weibull distributions based on estimation using three methods (Least Means Square – LSM, Maximum Likelihood – MLE, Method of Moments – MOM), and the mean values of classes with standard deviation bars, with respect to the global mean value and the global standard deviation.

The mean values and standard deviations of classes did not differ significantly from the global values. This means that marginal distributions of awaiting times for puncture and the amplitudes of EDM currents are statistically independent. This allows adopting of a model described by joint distributions variables,

$$f(a,\tau) = e^{-\left(\frac{a-A_0}{\eta}\right)^{\beta}} \left(a - A_0\right)^{\beta-1} \beta\eta^{-\beta}\, \lambda e^{-\lambda\tau}, \quad a > A_0, \quad \tau > 0. \tag{18.4}$$

Figure 18.21 shows a three dimensional histogram of EDM current amplitudes in puncture awaiting time, analytical distribution and their superimposition.

18.4.2. Transmission line phenomena

On account of high $\mathrm{d}u/\mathrm{d}t$ at the output of the inverter, EMI currents in the motor cable are, in fact, traveling wave phenomena in feeder cables and motor windings (Kempski, 2005; Magnusson *et al.*, 2001; Moreira *et al.*, 2002). The understanding of these phenomena is very important for a proper analysis of EMI spectra. An exemplification of the effect of traveling wave phenomena on CM and DM noise spectra is

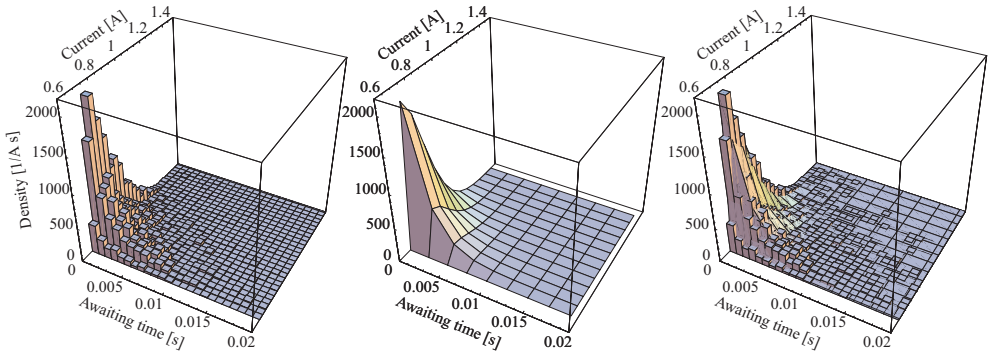


Fig. 18.21. Three dimensional histogram of EDM currents amplitudes in awaiting time to puncture, analytical distribution and their superimposing

shown in Fig. 18.22 and Fig. 18.23. Traveling waves have been excited in a shielded and unshielded open-ended cable fed by an inverter.

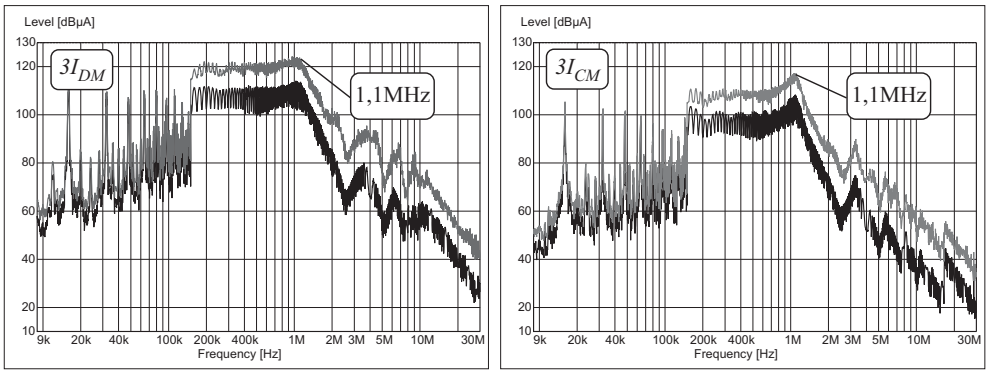


Fig. 18.22. EMI currents spectra in a shielded cable

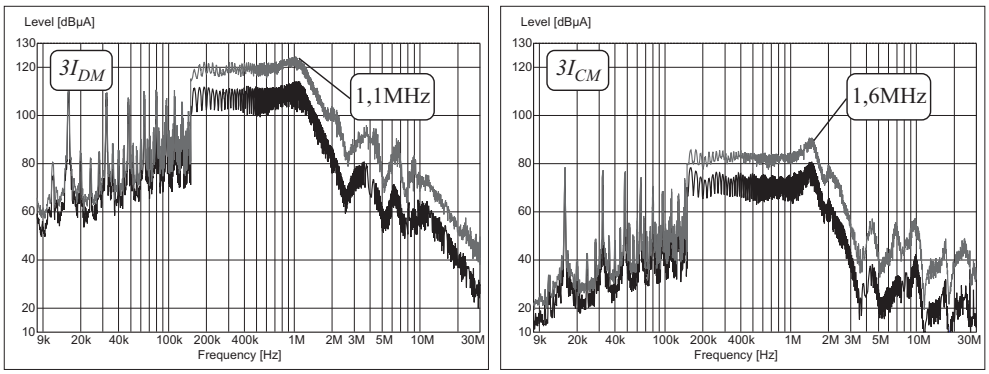


Fig. 18.23. EMI currents spectra in an unshielded cable

The frequency of the noise is formed by multiple reflections of the traveling wave and depends on the propagation time between reflections. The circuits of CM and DM currents can be quite different. However, in the case of the shielded cable, the frequencies of the main oscillatory modes are approximately the same, because electromagnetic waves propagate in the material with similar permittivity. The velocity of the electromagnetic waves can be expressed as

$$v = \frac{1}{\sqrt{\varepsilon_0 \varepsilon_r \mu_0 \mu_r}}, \tag{18.5}$$

where ε_0 is the dielectric constant, ε_rr is relative permittivity, μ_0 is the permeability of the free space, μ_r is relative permeability.

For the unshielded cable, the electromagnetic wave connected with DM noise propagates in the solid material as in the case of a shielded cable. Therefore, the spectrum of EMI noise remains unchanged. Otherwise, the electromagnetic wave that constitutes the CM leakage current propagates mainly in air, which results in an increasend wave velocity. This in turn causes a higher frequency of the main oscillation mode. The magnitude of CM spectra is much smaller because of decreased transverse line-to-ground parasitic capacitances.

The observations done on the basis of the above-presented spectra have been confirmed by measurements in the time domain (note the different current scales for DM and CM currents the in case of the unshielded cable). The waveforms of phase voltage, phase current, DM and CM noise currents are shown in Fig. 18.24.

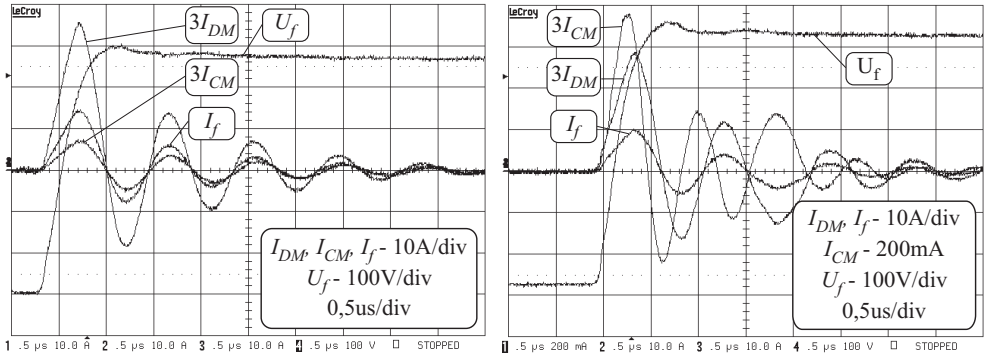


Fig. 18.24. Phase voltage, phase current, DM and CM noise currents in a shielded and unshielded cable

18.5. EMI mitigating techniques

For economic reasons, EMC should be considered early at the equipment design stage (Tihanyi, 1995). As a first step, simple methods of EMI mitigation are available employing good engineering practice such as wire separation and layout, shielding, HF effective earthing. If the basic engineering techniques give no satisfactory results, EMI filters can be an effective solution. However, as we showed (Kempski *et al.*,

2004), the choice of a method of adverse effects elimination requires caution due to complicated interactions between the components of the drive system.

For comparative analysis between drives with various passive filters, the measurements in a drive comprised of a two-quadrant 7.5 kW frequency converter and a 1.5 kW induction motor have been chosen as the case study.

Figure 18.25 shows the waveforms of CM voltage, shaft voltage and CM current in a motor PE wire in the chosen test arrangement without filters.

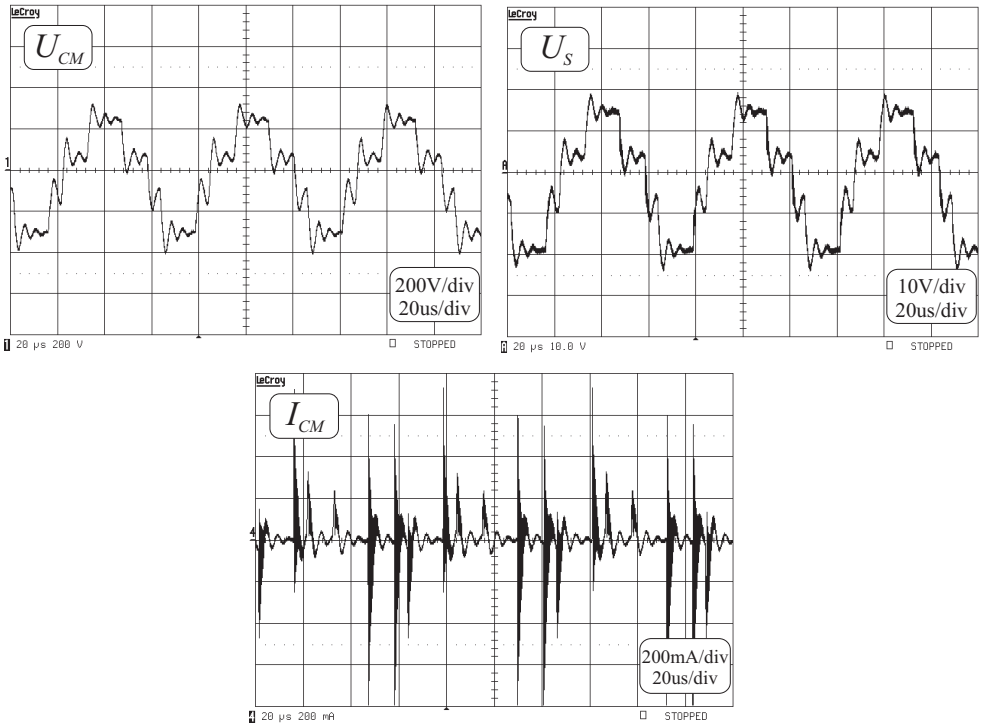


Fig. 18.25. CM voltage, shaft voltage, CM current

The spectra of CM currents on both sides of the converter are presented in Fig. 18.26.

As inductive filters, simple commonly used series reactors, CM choke and a slightly more sophisticated EMC transformer (Ogasawara and Akagi, 1996) have been taken into consideration. Additionally, a CM voltage filter having sinusoidal output voltages has been investigated.

18.5.1. Series reactors

Series reactors are the most popular—recommended by manufacturers—technique of reducing the $\mathrm{d}v/\mathrm{d}t$ of inverter output voltages and ripples of phase currents. Figure 18.27 shows the waveforms of a phase current in a drive without filters and with series reactors.

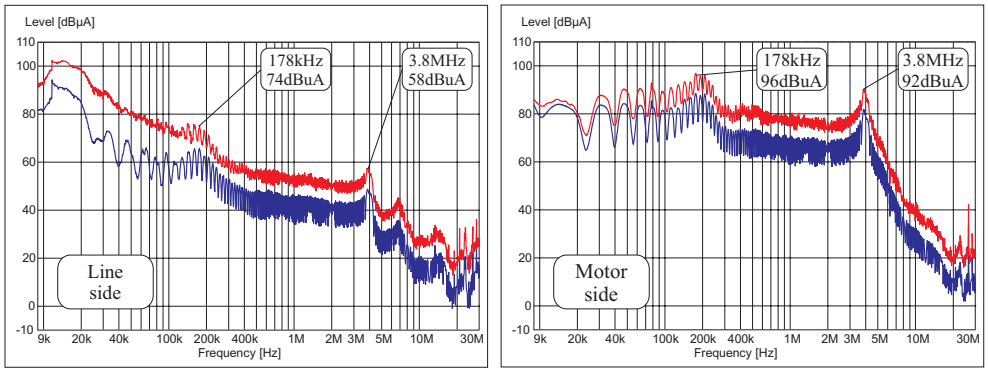


Fig. 18.26. Spectra of CM currents on the line and motor side

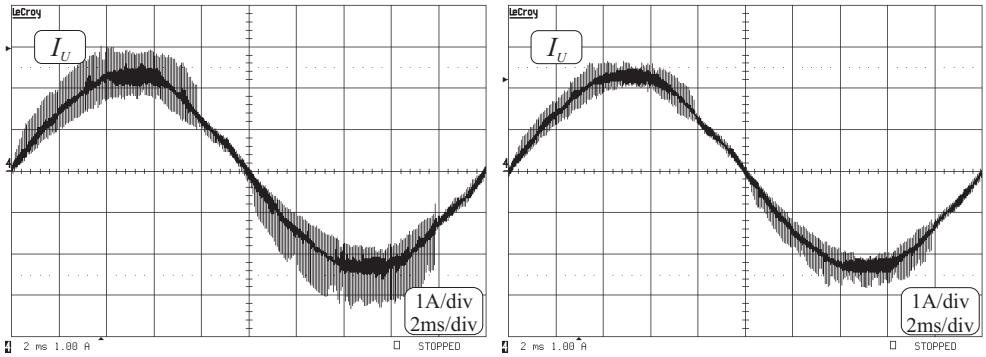


Fig. 18.27. Phase current in a drive without passive filters and with line reactors

Figure 18.28 shows the waveforms of CM voltage, shaft voltage and CM current, respectively, in a motor PE wire in the modified circuit.

In the waveforms there appear weakly damped low frequency oscillations. Additionally, in the CM current waveform high frequency spikes are visible at each switching event.

18.5.2. CM choke

Another often recommended measure to diminish CM currents is a CM choke. The CM choke has three windings wound on a common toroidal ferrite core. The relevant waveforms are presented in Fig. 18.29.

The shapes of the waveforms, are similar to those that have been obtained in the drive with line reactors. The only differences between them are the lower frequency of oscillations and significantly reduced high frequency spikes.

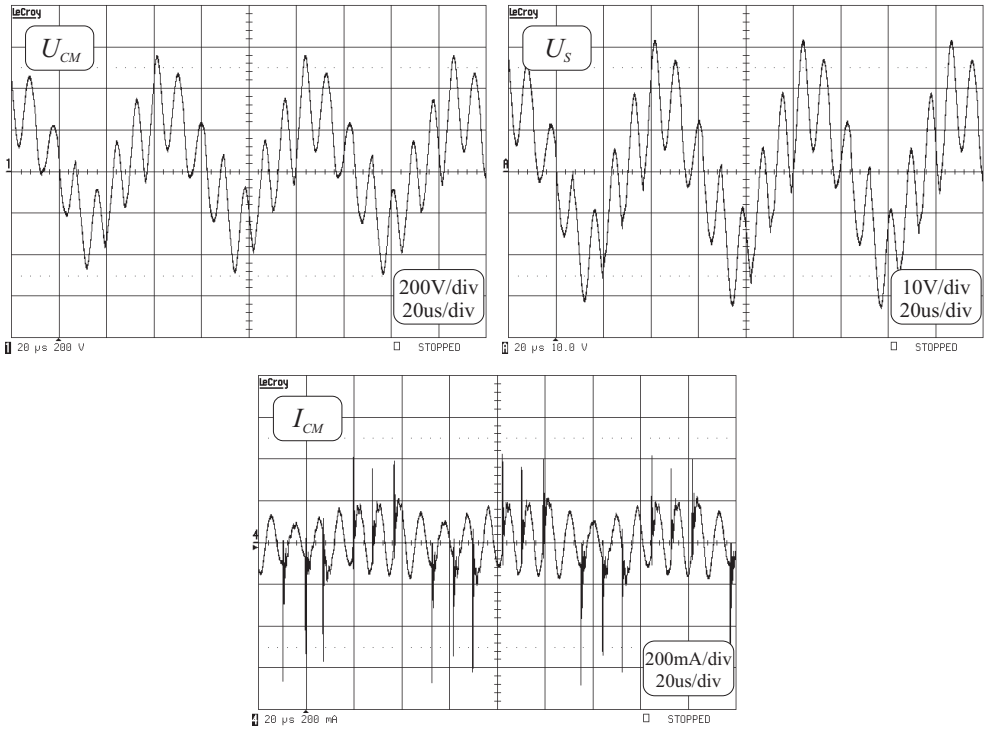


Fig. 18.28. CM voltage, the shaft voltage and the CM current in a drive with line reactors

18.5.3. CM transformer

A CM transformer is, in fact, a CM choke with an additional tightly coupled secondary winding shorted by a damping resistor. The value of the damping resistor should be selected to achieve an aperiodic decay form of a CM current. This assures the lack of oscillation in stepped waveforms of the CM voltage and the shaft voltage as well, Fig. 18.30.

18.5.4. Comparison of the influence of passive EMI filters on internal EMC of drives

Figure 18.31 shows the waveforms of CM currents (with their RMS values) in a motor PE wire in drive without filters and with the above-described filters.

The effect of both a CM choke and line reactors on CM current waveforms consists in an insertion of an additional inductance in the CM current path. The differences are caused by various inductances and different values of parasitic capacitances of the devices. Increased values of time constant and a characteristic impedance of the resonant circuit permit a significant reduction of the higher frequency components of the CM current on the motor side. However, due to a weaker damping factor the RMS value is not reduced significantly. High frequency spikes visible especially in

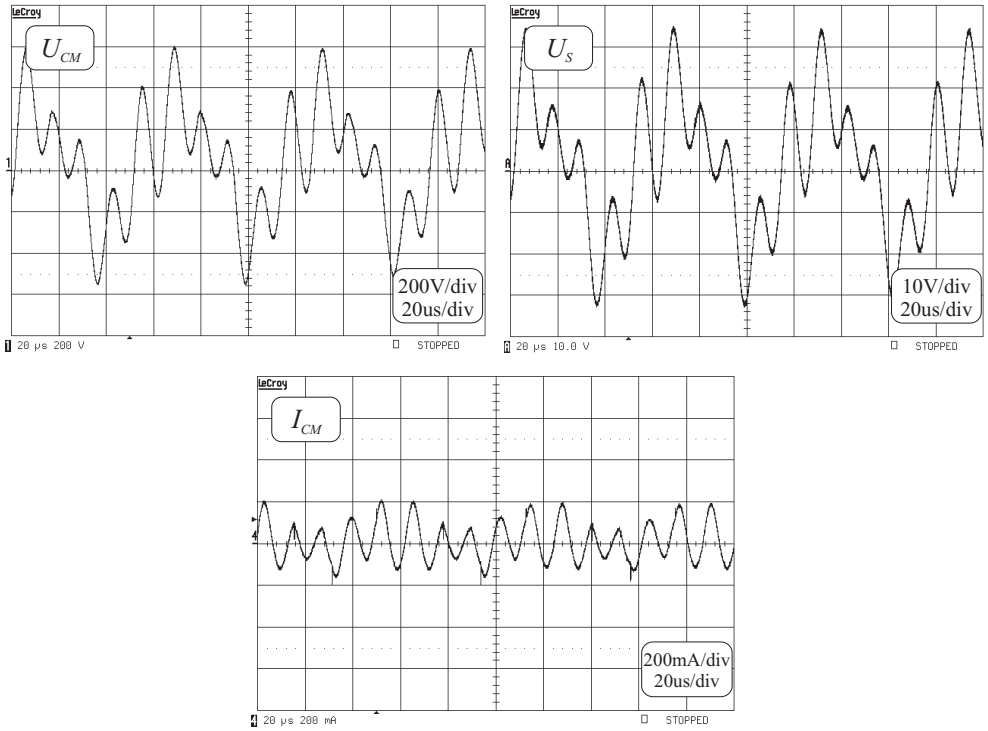


Fig. 18.29. CM voltage, the shaft voltage and the CM current in a drive with a CM choke

the CM current in the drive with line reactors are caused by parasitic turn-to-turn capacitances of inductors, which constitute a part of the CM current path.

An aperiodic decay form of the CM current in the drive with the CM transformer assures a reduction of its RMS value. The comparative results obtained for CM currents in a motor PE wire in the frequency domain are shown in Fig. 18.32 (IF BW = 9 kHz in the whole frequency range).

All of the applied filters attenuate the high frequency component (3.8 MHz) in the original spectrum of the system. However, due to a lower damping factor of a CM current path in a drive with line reactors or a CM choke, we have observed a high level of peaks at new low resonant frequencies (in the CISPR A range). The reason for additional high frequency resonant peaks is the presence of turn-to-turn parasitic capacitances of inductors. In the drive with a CM transformer, an increase in the damping factor has been achieved, which results in a suppression of the CM current level at both resonant frequencies and the spreading of the spectrum across a wide frequency range.

It would appear that the best solution to meet the requirements of the standard EN 61800-3 (CISPR B frequency range) might be a CM choke. However, the series resonance between the inductance of a CM choke and a motor-to-ground parasitic capacitance is responsible for the peak at frequency 80 kHz. In contrast, a CM transformer acts in the entire conducted emission range (CISPR A and CISPR B).

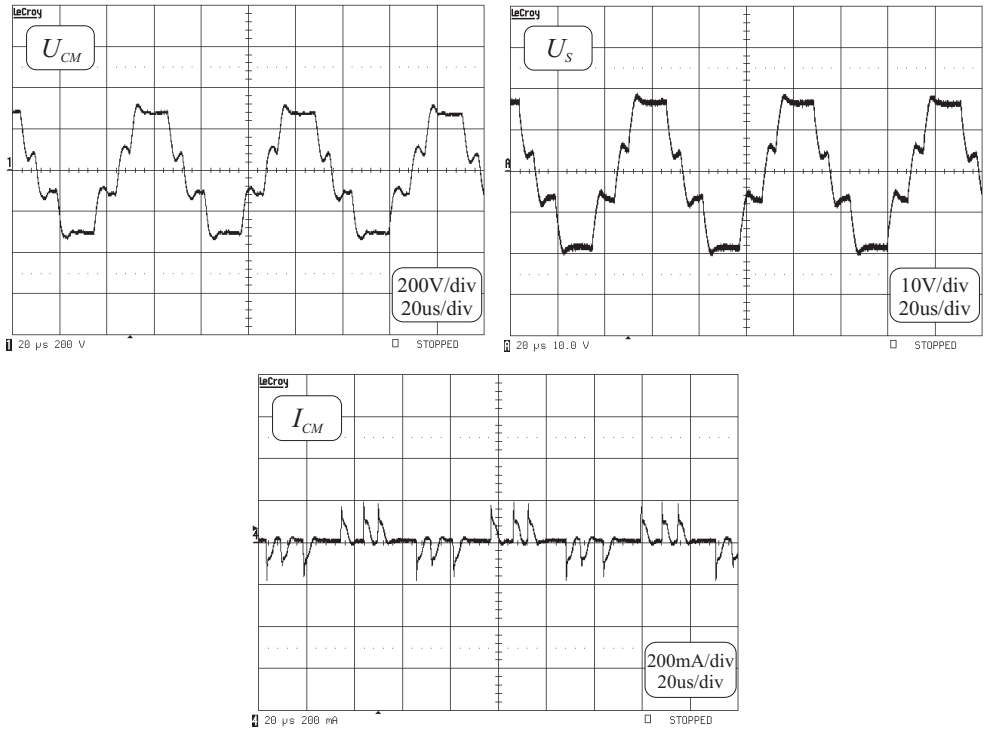


Fig. 18.30. CM voltage, the shaft voltage and the CM current in a drive with a CM transformer

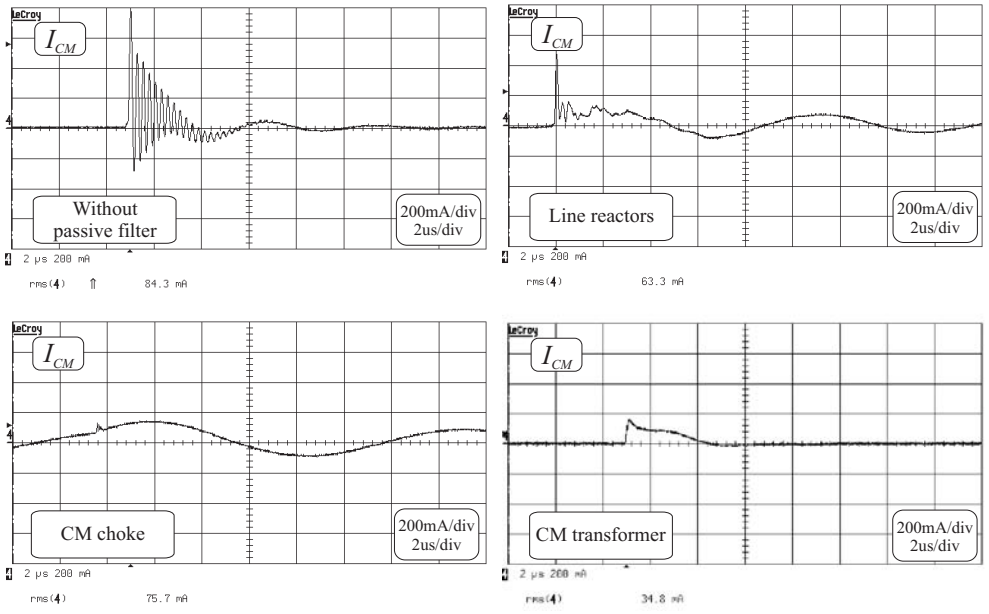


Fig. 18.31. Influence of passive filters on the CM current shape and its RMS value

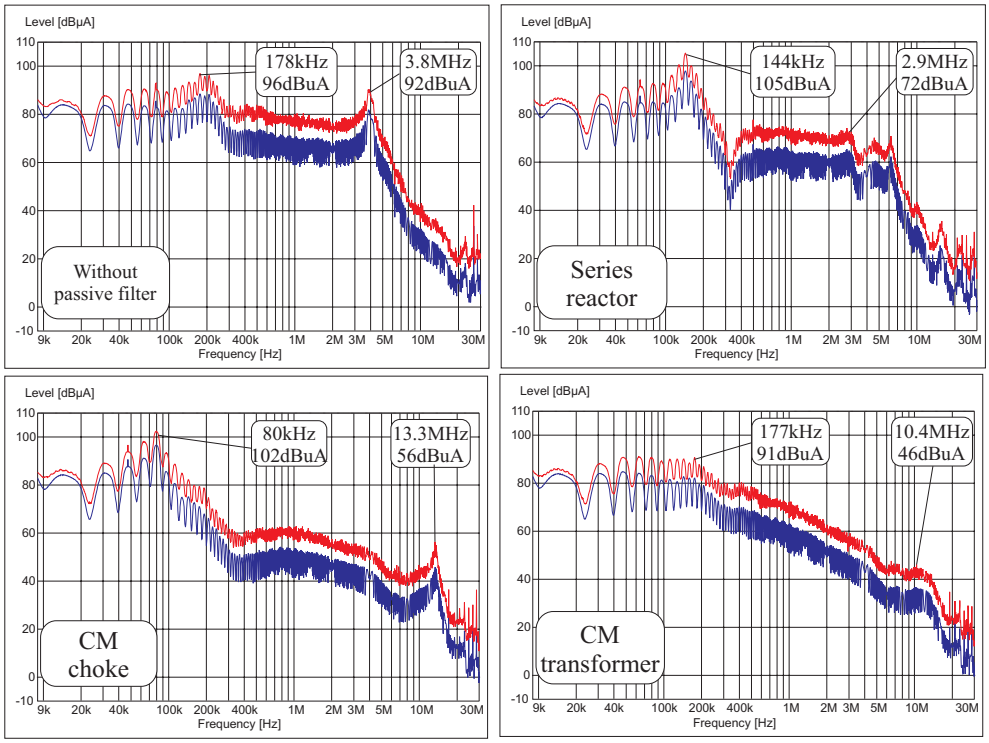


Fig. 18.32. Influence of passive filters on the spectrum of the CM current

The reason for this is shown in Fig. 18.33. The insertion loss of the CM choke reveals that it is, in fact, a second order resonant filter with a resonant frequency determined by the inductance of a CM choke and its parasitic turn-to-turn capacitance (parallel resonance). The insertion loss of the CM transformer is relatively low and has a flat shape. However, the damping resistance in an additional winding of transformer damps oscillations in all resonant circuits in the CM current path.

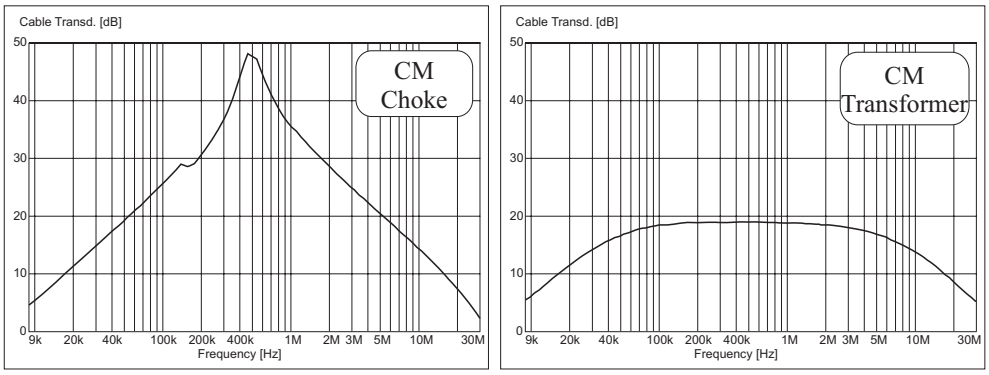


Fig. 18.33. Insertion loss of a CM choke and a CM transformer

Another aspect of internal compatibility is the possibility of damage to motor bearings caused by bearing currents. The risk of the appearance of bearing currents depends on the level of the shaft voltage that, in turn, depends on the level of the CM voltage.

Figure 18.34 shows three dimensional distributions of EDM current amplitudes in puncture awaiting times. The presented results are based on statistical analyses of large data of EDM currents, which have been measured in a special arrangement.

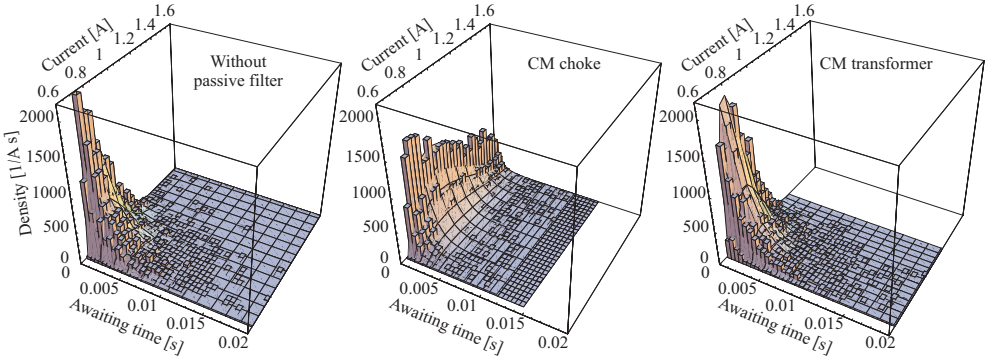


Fig. 18.34. 3D distributions of EDM currents in a drive with different passive filters

As we might expect from the highest level of the shaft voltage in the drive with the CM choke we observe the greatest risk of frequent occurrence of EDM currents of very high amplitudes in this drive arrangement. The attenuation of resonance oscillations in the drive with the CM transformer can decrease this risk, even in comparison with the drive without passive filters.

18.5.5. Zero CM voltage sinusoidal filter

Almost the entire compensation of the CM voltage at motor terminals is possible by means of the so-called sinusoidal filter, which was proposed by Akagi (Akagi *et al.*, 2002; Akagi and Tamura, 2006). It is a combination of common and differential mode gamma filters. The large inductance of series connected inductors (mainly of a CM choke) in an inverter-motor path causes a voltage drop, which is almost equal to the common mode voltage at the output of the inverter.

The cancellation of the CM voltage and, consequently, the shaft voltage permits the exclusion of the risk of EDM currents. Relevant measurement results of the CM voltage in the stator winding neutral point, shaft voltage, CM current in a motor PE wire, both in time and in frequency domain, are shown in Fig. 18.35.

Figure 18.36 shows that line-to-line voltages at the output of the sinusoidal filter are really sinusoidal. This practically excludes overvoltages at motor terminals in the system with a long cable. However, as we stated in our previous work (Kempski *et al.*, 2003; 2005), there are some problems due to magnetic saturation of cores in four-quadrant drives and in drive systems with hysteresis regulators e.g. DTC (Direct Torque Control).

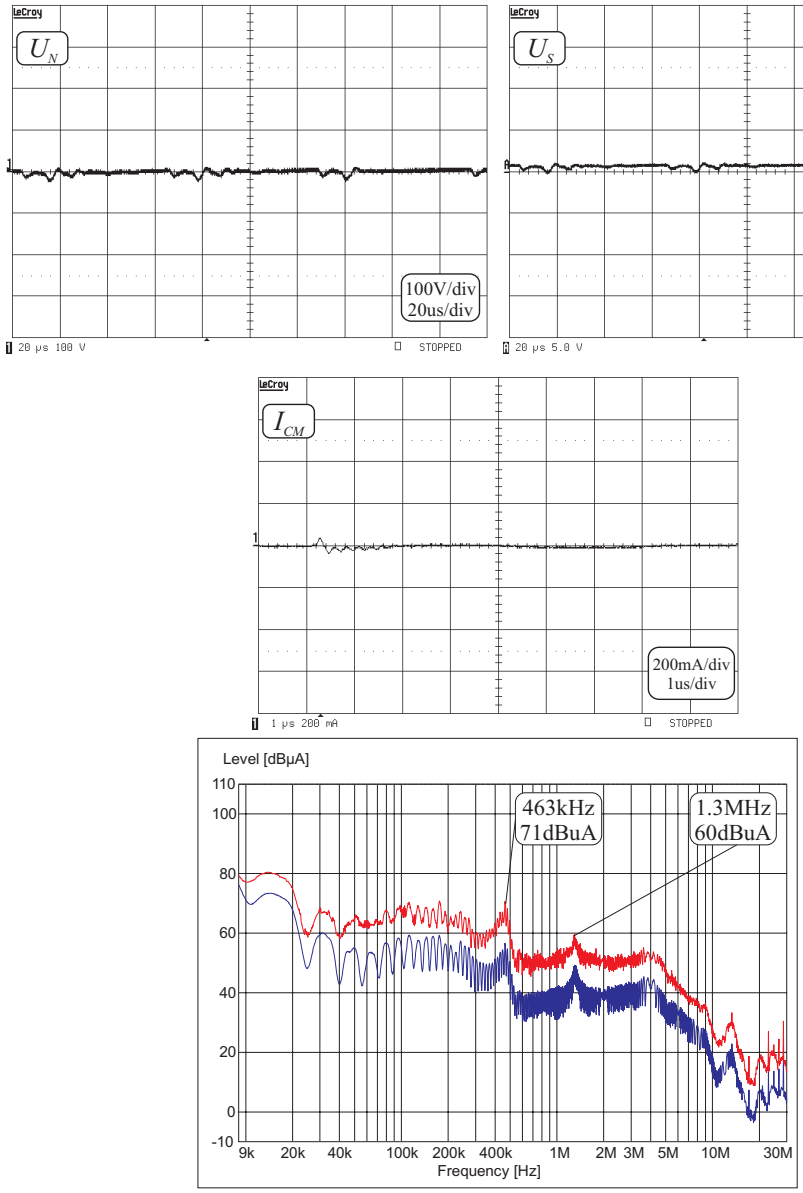


Fig. 18.35. CM voltage, the shaft voltage, CM currents (in both time and frequency domains) in a drive with a sinusoidal filter

18.6. Conclusions

The apparent anomalies have given EMC undeserved reputation for "black magic". However, the improvement in measurement techniques allows EMC to become an almost exact science. Specific EMC measuring methods require sophisticated test

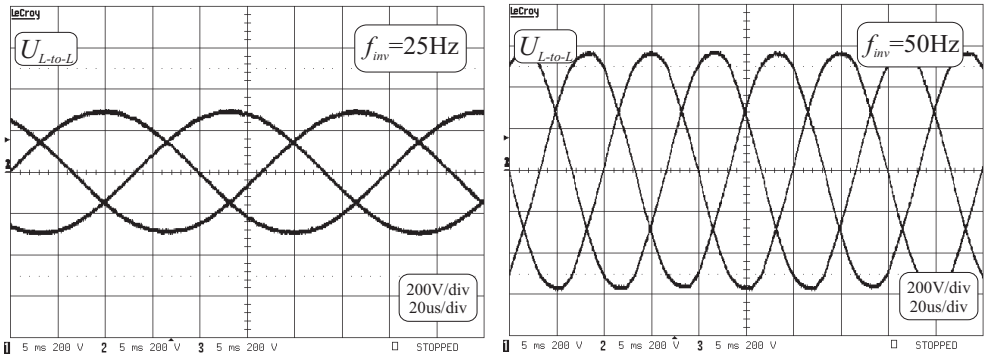


Fig. 18.36. Line-to-line voltages at motor terminals in a drive with a sinusoidal filter

equipment, but it is a prerequisite condition for understanding the physical phenomena which have been earlier considered as "anomalies".

EMC problems in a system comprising power electronic converters are unusually serious because of high $\mathrm{d}v/\mathrm{d}t$ rates of converter voltages and strong parasitic couplings. The complexity of parasitic electromagnetic phenomena in adjustable speed drives causes a need to take various aspects of electromagnetic compatibility into consideration simultaneously. This was the main reason why converter drives have been chosen as an exemplification of EMC problems in power electronics. In this chapter, phenomena linked with external and internal compatibility such as EMI currents, bearing currents and transmission line effects have been examined. On this basis the method of mitigating EMI problems have been proposed and investigated.

If one of the mitigation techniques is applied on the basis of preliminary measurements, it is strongly recommended to conduct all of the measurements again to ensure the effect of the mitigation technique is as desired, because of the interdependencies of the related phenomena.

The commonly used, most simple passive filters for the reduction of CM currents are line reactors and CM chokes. However, they can increase conducted emission measured by the LISN in the CISPR A frequency range, due to the lowering of the damping factor and the decreasing of the resonant frequency of the CM current path. Additionally, a higher level of weakly damped oscillations in the CM voltage waveform increases the risk of EDM currents. The weaknesses of a CM choke may be partly overcome by introducing an additional winding shorted by a damping resistor in the CM transformer.

Using the sinusoidal filter, it is possible to compensate the CM voltage at motor terminals and, consequently, to cancel the shaft voltage and the risk of EDM currents. However, in some applications there are some problems due to magnetic saturation of the core of the series inductor of the filter.

References

Akagi H., Hasegawa H. and Doumoto T. (2002): *Design and performance of a passive EMI filter for use with voltage source PWM inverter having sinusoidal output voltage and zero common-mode voltage.* — Proc. 33-rd IEEE *Power Electronics Specialists Conference*, Vol. 3, Cairns, Australia, pp. 1543–1550.

Akagi H. and Tamura S. (2006): *A Passive EMI Filter for Eliminating Both Bearing Current and Ground Leakage Current From an Inverter-Driven Motor.* — IEEE Trans. *Power Electronics*, Vol. 21, No. 5, pp. 1459–1469.

Jin M. and Weiming M. (2004): *A new technique for modeling and analysis of mixed mode conducted EMI noise.* — Proc. 35-th IEEE *Power Electronics Specialist Conf.*, Aachen, Germany, pp. 1134–1140.

Jin M., Weiming M. and Lei Z. (2004): *Determination of noise source and impedance for conducted EMI prediction of power converters by lumped circuit model.* — Proc. 35-th IEEE *Power Electronics Specialist Conf.*, Aachen, Germany, pp. 3028–3033.

von Jouanne A., Zhang H. and Wallace A.K. (1998): *An evaluation of mitigation techniques for bearing currents, EMI and overvoltages in ASD applicatios.* — IEEE Trans. *Industry Applications*, Vol. 34, No. 5, pp. 1113–1121.

Kempski A. (2001): *Capacitively coupled discharging currents in bearings of induction motor fed from PWM (pulse width modulation) inverters.* — J. Electrostatics, Vols. 51–52, pp. 416–423.

Kempski A. (2003): *Electromagnetic compatibility (EMC) external and internal of the systems containing power electronics converters.* — Pomiary, Automatyka, Kontrola, No. 2–3, pp. 33–36, (in Polish).

Kempski A. (2005): *Conducted electromagnetic emission in power converter drives.* — University of Zielona Góra Press, (in Polish).

Kempski A., Smoleński R. and Strzelecki R. (2002): *Common mode current paths and their modeling in PWM inverter-fed drives.* — Proc. 33-rd IEEE Power Electronics Specialists Conf. Cairns, Australia, Vol. 3, pp. 1551–1556.

Kempski A., Smoleński R. and Bojarski J. (2005): *Statistical model of electrostatic discharge hazard in bearings of induction motor fed by inverter.* — J. Electrostatics, Vol. 63, pp. 475–480.

Kempski A., Smoleński R., Piontek S. and Klytta M. (2000): *Internal compatibility of induction motor inverter drives.* — Proc. *Polish-German Symp.*, Zielona Góra, Poland, Part 1, pp. 255–258.

Kempski A., Smoleński R., Kot E. and Strzelecki R. (2005): *Series passive compensation of common mode voltage in multilevel inverter drives.* — Proc. 36-th IEEE *Annual Power Electronics Specialists Conf.*, Recife, Brazil, pp. 1833–1838.

Kempski A., Strzelecki R. and Smoleński R. (2004): *The influence of passive EMI filters on various aspects of electromagnetic compatibility.* — Proc. 35-th IEEE *Power Electronics Specialist Conf.*, Aachen, Germany, pp. 970–975.

Kempski A., Strzelecki R., Smoleński R. and Benysek G. (2003): *Suppression of conducted EMI in four-quadrant AC drive system.* — Proc. 34th IEEE *Power Electronics Specialists Conf.*, Acapulco, Mexico, pp. 1121–1126.

Macdonald D. and Gray W. (1999): *PWM related bearing failures.* — IEEE Industry Applications Magazine, Vol. 5, No. 4, pp. 41–47.

Magnusson P.C., Alexander G.C., Tripathi V.K. and Weisshaar A. (2001): *Transmission Lines and Wave Propagation*. — London: CRC Press.

Mohan N., Undeland T.M. and Robbins W.P. (1995): *Power Electronics*. — New York: John Wiley & Sons Inc.

Moreira A.F., Lipo T.A., Venkataramanan G. and Bernet S. (2002): *High-frequency modeling for cable and induction motor overvoltage studies in long cable drives*. — IEEE Trans. Industry Applications, Vol. 38, No. 5, pp. 1297–1306.

Ogasawara S. and Akagi H. (1996): *Modeling and damping of high-frequency leakage currents in PWM inverter-fed AC motor drive systems*. — IEEE Trans. Industry Application, Vol. 32, No. 5, pp. 1105–1113.

Palis F., Mecke R., Mecke H. and Rummel T. (1997): *Influence of system parameters on EMC behaviour of IGBT inverters*. — Proc. 7-th European Conf. *Power Electronics and Applications*, Trondheim, Norway, Vol. 2, pp. 810–814.

Qu S. and Chen D. (2002): *Mixed-mode EMI noise and its implications to filter design in offline switching power supplies*. — IEEE Trans. Power Electronics, Vol. 17, No. 4, pp. 502–507.

Ran L., Casadei D., Clare J., Keith J.B. and Christopoulos C. (1998): *Conducted electromagnetic emissions in induction motor drive system*. — IEEE Trans. Power Electronics, Vol. 13, No. 4, pp. 757–775.

Shen W., Wang F., Boroyevich D. and Liu Y. (2004): *Definition and acquisition of CM and DM EMI noise for general-purpose adjustable speed motor drives*. — Proc. 35-th IEEE *Power Electronics Specialist Conf.*, Aachen, Germany, pp. 1028–1033.

Skibinski G.L., Kerkman R.J. and Schlegel D. (1999): *EMI emissions of modern PWM AC drives*. — IEEE Industry Applications Magazine, Vol. 5, No. 6, pp. 47–81.

Smoleński R. (2003): *Bering currents and methods of their reduction in the drives with PWM frequency converters*. — Ph.D. thesis, University of Zielona Góra Press, (in Polish).

Smoleński R., Bojarski J., Kempski A. and Strzelecki R. (2002): *Statistical method of bearing damage risk estimation in PWM inverter-fed drives*. — Proc. 7-th Int. Conf. *Probabilistic Methods Applied to Power Systems*, Naples, Italy, Vol. 2, pp. 991–996.

Teulings W., Schanen J.L. and Roudet J. (1997): *A new technique for spectral analysis of conducted noise of SMPS including interconnects*. — Proc. 28th IEEE *Power Electronics Specialist Conf.*, St. Louis, Missouri, USA, Vol. 2, pp. 1516–1521.

Tihanyi L. (1995): *Electromagnetic Compatibility in Power Electronics*. — Sarasota: J.K. Eckert & Company, Inc.

Tse K.K., Chung H.S-H., Hui S.Y. and So H.C. (2000): *Analysis and spectral characteristics of a spread-spectrum technique for conducted EMI suppression*. — IEEE Trans. Power Electronics, Vol. 15, No. 2, pp. 399–409.

Weston D.A. (1991): *Electromagnetic Compatibility*. — Principles and Applications, New York, Marcel Dekker, Inc.

Williams T. and Armstron K. (2000): *EMC for Systems and Installations*. — Oxford: Butterworth-Heinemann Ltd.

Chapter 19

POWER ELECTRONICS SYSTEMS TO IMPROVE THE QUALITY OF DELIVERY OF ELECTRICAL ENERGY

Grzegorz BENYSEK*, Marcin JARNUT*, Jacek RUSIŃSKI*

19.1. Introduction

Electricity is a very useful and popular energy form which plays an increasing role in our modern industrialized society. Scarcer natural resources and the ubiquitous presence of electrical power make it desirable and continuously increase the demand, causing power systems to operate close to their stability and thermal ratings. All the above mentioned reasons together with the high penetration of Distributed Resources (DR) and higher than ever interest in the quality of the delivered energy are the driving forces responsible for extraordinary changes taking place in the electricity supply industry worldwide.

Against this background of rapid changes, the expansion programmes for many utilities are being thwarted by a variety of environmental and regulatory pressures that prevent the building of new transmission lines and electricity generating plants, the construction of which is becoming increasingly difficult.

The one-line diagram shown in Fig. 19.1 illustrates the Electrical Power System (EPS) and its major components: generation, transmission and distribution systems. Electrical power is generated at power stations predominantly by synchronous generators that are mostly driven by steam or hydro turbines. Hence, the electrical power generated at any such a station usually has to be transmitted over a great distance, through the transmission systems, to the distribution systems. The distribution networks distribute the energy from the transmission grid or small/local DR to customers. An in-depth analysis of the options available for maximizing the existing transmission and distribution resources, with high levels of stability and Power Quality (PQ), points in the direction of power electronics (Akagi, 1994; 1995; 1996;

* Institute of Electrical Engineering
e-mails: {G.Benysek, M.Jarnut, J.Rusinski}@iee.uz.zgora.pl

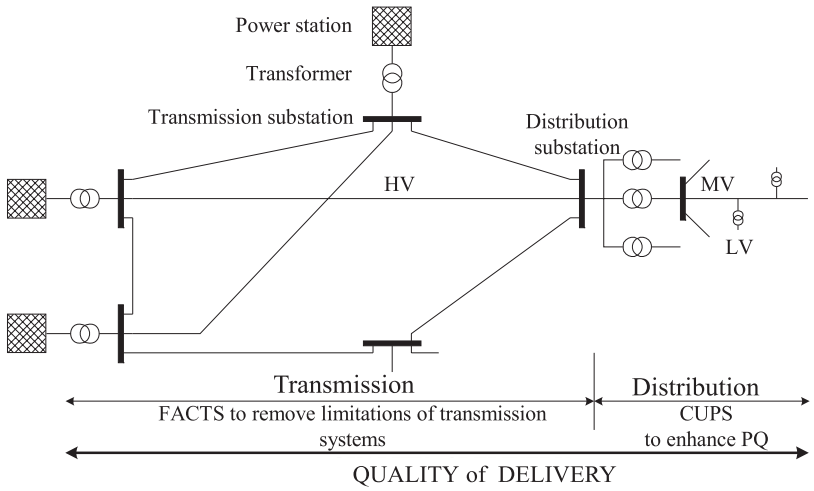


Fig. 19.1. Simplified one-line diagram of the power system

Hingorani, 1993; Gyugyi, 2000; Gyugyi *et al.*, 1995; 1998; IEEE/CIGRE, 1995; Mohan *et al.*, 1995; Song and Johns, 1999). There is general agreement that novel power electronics equipment is a potential substitute for conventional solutions, which are normally based on electromechanical technologies that have slow response times and high maintenance costs (Hingorani and Gyugyi, 2000).

Two kinds of power electronics applications gaining importance in power systems are already well defined: active and reactive power control and power quality improvement. The first application area is for arrangements known as the Flexible Alternating Current Transmission System (FACTS), where the latest power electronic devices and methods are used to control the transmission side of the network (Hingorani, 1993, 1998; Gyugyi *et al.*, 1995; Renz, 1999; Schauder *et al.*, 1998b). The second application area is for devices known as the CUstom Power System (CUPS), which focuses on the distribution system supplying the energy to end-users and is a technology created in response to reports of poor power quality of supply affecting factories, offices, and homes (Akagi, 1995; Akagi and Fujita, 1995; Aredes *et al.*, 1998; Fujita and Akagi, 1998; Ghosh and Ledwich, 2001; 2002; Hingorani, 1998; Jeon and Cho, 1997; Meckien and Strzelecki, 2002).

Transmission systems. Traditional solutions for upgrading the electrical transmission system infrastructure have been primarily in the form of new power plants, new transmission lines, substations, and associated equipment. However, as experience has proven, the process of authorizing, locating, and constructing new transmission lines has become extremely difficult, expensive and time consuming. It is envisaged that, alternatively, FACTS controllers can allow the same objectives to be met with no major alterations to system layout.

The potential benefits of employing FACTS controllers include the reduction of operation and transmission investment costs and implementation time compared to the construction of new transmission lines, increased system security and reliability,

increased power transfer capabilities, and an overall enhancement of the quality of the electrical energy delivered to customers (CIGRE, 1999; IEEE/CIGRE, 1995).

The application of power electronics devices to power transmission has a long tradition. It started with High Voltage Direct Current (HVDC) transmission systems. Thyristor-based HVDC installations provided a means for interconnecting power systems with different operating frequencies – e.g., 50/60 Hz, for interconnecting power systems separated by the sea and for interconnecting weak and strong power systems (Arrillaga, 1999; Hingorani, 1996). The most recent development in HVDC technology is the HVDC system based on solid-state voltage source converters, which permits independent, fast control of active and reactive powers (Asplund *et al.*, 1998).

In addition to the above, there are other power electronics controllers that are members of the FACTS family. One such a device is the Thyristor Controlled Braking Resistor (TCBR), which allows a quantity of real power to be dissipated in the resistor during the fault and therefore restricts machine acceleration and increases a system stability. A Thyristor Controlled Phase Angle Regulator (TCPAR) injects voltage in quadrature with the line voltage. Therefore, by adjusting the magnitude of the injected voltage, the phase angle between the sending and receiving end voltages can be adjusted, thus increasing the stability limit allowing the system to operate at a higher power angle, provided the thermal limit is not reached.

With respect to FACTS equipment, Voltage Source Converter (VSC) technology, – which utilizes self-commutated thyristors/transistors, such as Gate Turn Off (GTO), the Integrated Gate Commutated Thyristor (IGCT), and the Insulated Gate Bipolar Transistor (IGBT) – has been successfully applied in a number of installations worldwide for STATic synchronous COMpensators (STATCOMs), several of which have recently been completed in the U.S., (Reed *et al.*, 2001) – as well as for Unified Power Flow Controllers (UPFCs) (Renz *et al.*, 1998; Schauder *et al.*, 1998a).

The above mentioned transmission system installations appear in addition to the earlier generation of power electronics systems that utilize line-commutated thyristor technology for Static Var Compensators (SVCs) (Ekanayake and Jenkins, 1996; Gyugyi, 1988; IEEE, 1987; 1995), Thyristor Switched Series Capacitor (TSSC) and Thyristor Controlled Series Compensators (TCSCs) (Chistl *et al.*, 1992; Keri *et al.*, 1992).

The newest member of the FACTS family is an Interline Power Flow Controller (IPFC), proposed for providing flexible power flow control in a multi-line power system (Strzelecki *et al.*, 2001a; 2002; 2004d; 2004e; 2005b). In an IPFC, two or more parallel lines are compensated by Static Synchronous Series Compensators (SSSCs), which are connected to a common Direct Current (DC) link. Thus SSSCs can provide series compensation to the line to which they are connected. In addition, they can also transfer real power between the compensated lines. This capability makes it possible to equalize both real and reactive power between the lines, to transfer power from an overloaded line to an underloaded one and to damp out system oscillations resulting from a disturbance.

It can be concluded that the flexible Alternating Current (AC) transmission technology permits a greater control of power flow. Since these devices provide very fast power swing damping, the power transmission lines can be securely loaded up to their thermal limits.

Distribution systems. As with FACTS devices in transmission systems, power electronics devices can be applied to power distribution systems to increase the reliability and quality of power supplied to customers – to increase the PQ (Strzelecki and Benysek, 2004b; Strzelecki *et al.*, 2003h; 2003i; Thomsen, 1999). The devices applied to power distribution systems for the benefit of customers (end-users) are called custom power systems. Through this technology the reliability and quality of the power delivered can be improved in terms of reduced interruptions as well as reduced voltage and current variations and distortions. The proper use of this technology will benefit all industrial, commercial and domestic customers.

Custom power devices are basically a compensating type, used for active filtering, load balancing, power factor correction and voltage regulation. Active filtering, which predominantly is responsible for the elimination of harmonic currents and voltages, can be both shunt and series. Some CUPS devices with active filtering functions are used as load compensators, in which mode they correct the imbalance and distortions in the load currents, such that the compensated load draws a balanced sinusoidal current from the AC system. Some other devices are operated to provide a balanced, harmonic free voltage to the customers. Below are described selected devices that are members of the CUPS family.

A Parallel Active Power Filter (PAPF) (Greczko *et al.*, 2001; Kot *et al.*, 2000; Moran *et al.*, 1997; Peng, 1998; Peng *et al.*, 1986; Singh *et al.*, 1999; Strzelecki and Sozański, 2001; Strzelecki and Supronowicz, 1999; 2000), called also Distribution STATic synchronous COMpensator (DSTATCOM) (Ghosh and Ledwich, 2002), is a device that can complete current compensation, i.e., power factor correction, harmonic filtering, load balancing, and which can also perform voltage regulation. A Series Active Power Filter (SAPF), called also the Dynamic Voltage Restorer (DVR) (Ghosh and Ledwich, 2002; McHattie, 1998), is a device that can provide voltage based compensation and therefore can protect the sensitive loads from sags/swells and interruptions in the supply side. Still, when there is distortion in the source voltage, the SAPF provides a harmonic free voltage to the customers. A Unified Power Quality Conditioner (UPQC) (Aredes and Heumann, 1996; Elnady *et al.*, 2001; Peng and Lai, 1996; Pengcheng, 2003) has the same structure as that of a UPFC (Chen *et al.*, 2000; Fujita *et al.*, 1999) and can complete both current and voltage compensation at the same time.

Voltage Active Power Filters (VAPFs) (Liang and Nwankpa, 2000; Moran *et al.*, 1989; Strzelecki *et al.*, 2003a; 2003b; 2003c; 2003d; 2003e; 2003f; 2003g; 2004a; 2004b; 2004c; 2004f; 2005c) present a different way for power quality improvement. In a VAPF, PQ improvement is possible because the parallel connected VSC acts as a sinusoidal voltage source with fundamental frequency, and therefore the conditioner is suited to fulfil a wide range of different tasks:

- to prevent "dirty" loads from polluting the electrical distribution network;
- to protect sensitive loads from line disturbances as voltage sags and distortions.

On the basis of the above discussion it can now be stated that the major aim of this chapter is to introduce power delivery problems and to discuss the solutions of some of these technical difficulties using power electronics based devices. To achieve this goal, generally in the first part power electronics devices for transmission control

will be introduced. Special attention will be paid to the newest and most promising members of the FACTS family to interline power flow controllers. Their principle of operation, basic properties and control will be presented. This part deals also with a method of probabilistic dimensioning of the devices for transmission control. In the next part, custom power solutions to some of the power quality problems of distribution systems will be introduced. As earlier, special attention will be paid to three devices, the UPQC and the VAPF, with the latter being most promising because of its variety. Finally, the last part shows directions for some future investigations.

19.2. Modern power electronics systems for transmission control

Generally, FACTS devices can be divided into three major categories: series devices; shunt devices; and combined devices.

As a series device, a variable impedance (capacitor, reactor etc.) or power electronics based variable source which injects the voltage in series with the line could be utilized. When the injected voltage is in phase quadrature with the line current, the series source exchanges (supplies or consumes) only reactive power and thus predominantly affects active power in the line. On the other hand, if the injected voltage is in phase with the line current, the series source handles (supplies or consumes) active power but predominantly affects reactive power in the line. However, in the latter case an external source of active power is needed (it could be an Energy Storage System (ESS) or a shunt connected variable source).

As a shunt device, variable impedance (capacitor, reactor etc.) or a variable source which injects the current into the system at the point of common coupling may be utilized. As in the case of series devices, if the injected current is in phase quadrature with the line voltage, the shunt device exchanges only reactive power with the line. Other phase relationships will also cause active power exchange. But this time there is no need for an extra source, because a shunt connected source can produce active power itself.

As a combined device, unified series and shunt variable sources may be utilized (UPFC). In this type of device, the current is injected into the system with the shunt part, and the voltage with the series part. Because both parts are unified, there can be real power exchange between parts through the power link.

Unified power flow controllers can control active and reactive power transmitted through the transmission line. However, one of the disadvantages of this solution is the need to equip every transmission line with an independent arrangement. This feature, then, is not attractive from the economical point of view, especially in meshed systems.

The problem with power flow control in a number of separate lines can be solved by using so-called interline power flow controllers. Generally speaking, the IPFC is a combination of two or more static synchronous series compensators which are coupled via a common DC link to facilitate bi-directional flow of active power between the AC terminals of SSSCs. All SSSC devices are controlled to provide independent series reactive compensation for the adjustment of active power flow in each line and to maintain the desired distribution of reactive power flow among the lines. However, if the IPFC arrangement consists only of SSSC devices, there is danger that reactive

power flow control in one line will deteriorate the distribution of reactive power flow in the others. This happens because series connected converters cannot internally generate a voltage in phase with the line current; in other words, they cannot provide series active compensation (deliver active power to the transmission line). Therefore, the active power needed to control reactive power in one transmission line must be provided from other lines, which leads to considerable reactive power growth. This problem can be overcome in many ways, for example, by increasing the size of the energy storage element (connecting batteries or super-capacitors), by connecting to the common DC element distributed resources and, finally, by connecting one, common for all series converters, STATCOM to provide shunt reactive compensation and supply or absorb the overall active power deficit of SSSCs.

19.2.1. SSSC based interline power flow controllers

The IPFC employs a number of series DC/AC converters, namely SSSCs, each providing series compensation for a different line. However, standing alone SSSCs are unable to control the reactive power flow and thus the proper load balancing of the line. Therefore, in an IPFC, series VSCs are tied together at their DC link capacitors. Therefore, converters do not only provide series reactive compensation but can also be controlled to supply active power to the common DC link from its own transmission line or absorb active power from the DC link to its own transmission line (asynchronous tie) – series VSCs provide reactive power flow. Thus IPFC power flow control capability is the same as that of a UPFC device. The only difference is that the active power demand of a given converter is compensated by another series converter from another line instead of a shunt converter as in UPFCs.

The general structure of an IPFC is shown in Fig. 19.2. For simplicity the transmission system consists of two lines. *Line1* is represented by a reactance X_1, a sending end bus voltage $\bar{V}_1$, a receiving end bus voltage $\bar{V}_{21}$ and a series injected voltage $\bar{V}_{IPFC1}$. Similarly, *Line2* is represented by X_2, and voltages $\bar{V}_1$, $\bar{V}_{22}$ and $\bar{V}_{IPFC2}$.

As has been explained above, an IPFC series converter injects a controllable voltage, for given *Line1*, $\bar{V}_{IPFC1}$. The injected series voltage can be decomposed into two components: $\bar{V}_{IPFC1q}$, in quadrature, and $\bar{V}_{IPFC1p}$, in phase with the line current. The first component, produced internally by the converter, provides series reactive compensation and is proportional to the reactive power $Q_{\Sigma 1}$ exchanged between the series converter and the line (Strzelecki *et al.*, 2004e):

$$Q_{\Sigma 1} = \mathrm{Im}\left\{\bar{V}_{IPFC1}\bar{I}_1^*\right\} = V_{IPFC1q}I_1, \tag{19.1}$$

where $\bar{I}_1^*$ is the conjugate of $\bar{I}_1$.

To provide series active compensation, the series converter has to produce the second component in phase with the line current, proportional to the active power $P_{\Sigma 1}$ exchanged with the transmission line. The active power exchanged with *Line1* is given by the following equation (Strzelecki *et al.*, 2004e):

$$P_{\Sigma 1} = \mathrm{Re}\left\{\bar{V}_{IPFC1}\bar{I}_1^*\right\} = V_{IPFC1p}I_1. \tag{19.2}$$

However, this power cannot be internally produced by the series converter, it has to be absorbed (or supplied) through the DC tie from (or to) the other transmission

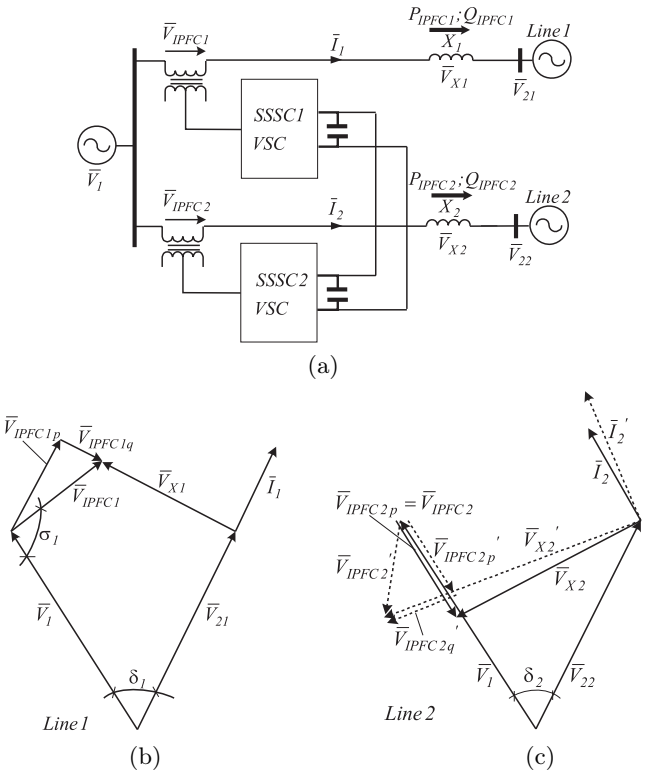


Fig. 19.2. (a) Single line diagram of an IPFC with two series VSCs, (b) possible vector diagram for *Line1*, (c) possible vector diagram for *Line2*

lines. Because to maintain a constant DC voltage, the net active power through the DC link has to be zero (neglecting losses), thus the active power supplied by one of the transmission lines must be equal to that injected into the others. Therefore the operating constraint representing active power exchange between or among the two series converters (Fig. 19.2) via the common DC link is given by Strzelecki *et al.* (2004e):

$$|P_{\Sigma 1}| = |P_{\Sigma 2}| \,. \tag{19.3}$$

The phasor diagrams in Fig. 19.2 mirror the situation when transmission *Line1* has to be controlled by *SSSC1* somehow to secure only active power flow to the receiving end bus. Assuming that the series converter *SSSC2* is secondary in relation to *SSSC1* and thus has to maintain a constant DC voltage (has to exchange with the DC tie required active power), one can see that in transmission *Line2* it is possible to both the satisfy active power demands and control, with a limited degree of freedom, its own reactive power flow (degree of freedom depends on the number of transmission lines the participating in satisfying active power demands of their master and/or on parameters of transmission lines):

$$|I_1 V_{IPFCp1}| = |I_2 V_{IPFCp2}| = \left|I_1' V_{IPFCp2}'\right| \,. \tag{19.4}$$

The active and reactive powers injected into the n-th transmission line by the SSSC voltage sources are given by, respectively (Strzelecki *et al.*, 2004e; 2005b),

$$P_{\Sigma n} = \frac{V_{IPFCn} V_{2n}}{X_n} \sin\left(\delta_n + \sigma_n\right) - \frac{V_1 V_{IPFCn}}{X_n} \sin\left(\sigma_n\right), \tag{19.5}$$

$$Q_{\Sigma n} = \frac{V_{IPFCn}^2}{X_n} + \frac{V_1 V_{\Sigma n}}{X_n} \cos\left(\sigma_n\right) - \frac{V_{IPFCn} V_{2n}}{X_n} \cos\left(\delta_n + \sigma_n\right). \tag{19.6}$$

From these figures an observation can be made that not only the phasor of the injected voltage but also its amplitude influences active power produced by the series converter. Therefore the $\bar{V}_{IPFCn}$ voltage must be controlled in order to secure an active power balance in the DC link, keeping in mind that its maximum achievable value $V_{IPFCn\,\max}$ is determined by the device rating. The rotation of the series injected voltage phasor with its maximum achievable value $V_{IPFCn\,\max}$ from 0 to 360° determines the control area within the boundary circle (Strzelecki *et al.*, 2004e):

$$\left(V_{IPFCn}\,\cos\sigma_n\right)^2 + \left(V_{IPFCn}\sin\sigma_n\right)^2 = V_{IPFCn\,\max}^2, \tag{19.7}$$

as in Fig. 19.3.

The inspection of the figures shows that if the series injected voltage is controlled to keep its end on a line that is parallel to the voltage drop across the line impedance X_n (so-called "voltage compensation line" (Strzelecki *et al.*, 2001a)), then there is produced a constant real power demand $P_{\Sigma n} = \text{const}$, independent of σ_n. Therefore, moving the converter's operating point from one "voltage compensation line" to another it is possible to change its real power demands. Considering two transmission lines equipped with an IPFC, where an arbitrary series converter in *Line1* is treated as a master in relation to the series converter in *Line2*, in order to satisfy active power demands of which operates along a selected "voltage compensation line", the slave converter has to operate along a complementary "voltage compensation line" with respect to the line defining zero active power demand ($P_{\Sigma n} = 0$). Only then will the net active power through the DC link (neglecting losses) be zero and the DC voltage

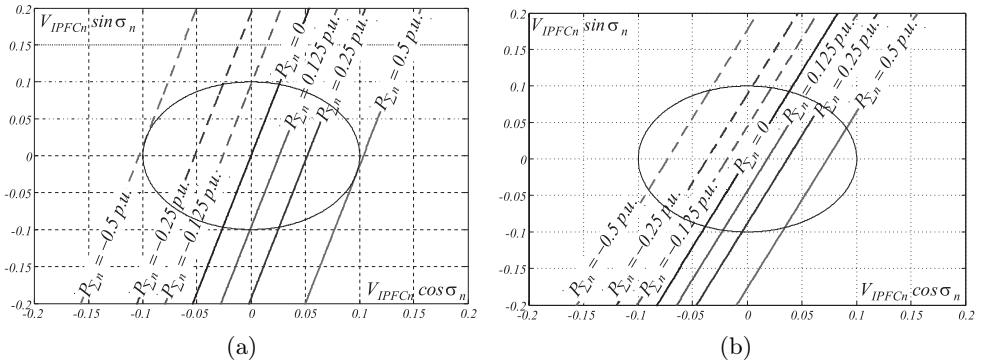


Fig. 19.3. Variation of the series injected voltage: (a) $\delta_n = \pi/6$, $X_n = 1$ p.u, (b) $\delta_n = \pi/4$, $X_n = 1$ p.u.

constant. From the figures an observation can be made that the complementary lines of the two converters must be in the opposite direction. Additionally it is possible to claim that maximum achievable real power exchanged with other transmission lines, limited by the converter's rating, strongly depends on the transmission angle δ_n and the transmission line reactance X_n. This is true because in lines with a relatively large δ_n angle or low X_n impedances, the line current increases and, in consequence, $P_{\Sigma n}$.

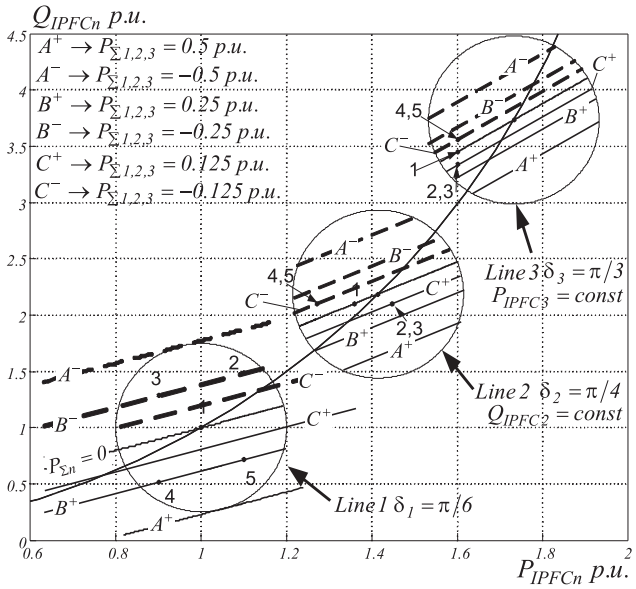


Fig. 19.4. Behavior examples of an IPFC compensating three independent transmission lines; 1, 2, 3, 4, 5 – operating points

Additionally, Fig. 19.4 presents the results of theoretical investigations for an IPFC compensating three independent transmission lines. Studying the above results one can say that while slave converters secure active power balance in the DC link, there is still a relatively large range for active and reactive power compensation. For example, the reactive power compensation range for the converter in transmission *Line2* at the operating point 5 is about 0.25 p.u., while for the uncoupled device it is about 0.27 p.u. Reasons for this are the relatively large transmission angles of the slave converters in relation to the master's transmission angle (generally speaking, large active powers transmitted through slave lines in relation to active powers transmitted through master lines).

In the sense of maximum influence on transmission characteristics and, consequently, transient stability, the IPFC controllable parameters (for given line n, respectively V_{IPFCn} and σ_n) should be determined to achieve a maximum impact on transmittable active power. Therefore the optimal parameters should be determined from the following equations: $\partial P_{IPFCn}/\partial V_{IPFCn} = 0$, $\partial P_{IPFCn}/\partial \sigma_n = 0$. Because the first equation does not have a solution, the maximum impact is achieved if the V_{IPFCn} parameter is set to its maximum value, $V_{IPFCn\,\max}$, determined by the device

rating. However, the second dependency has a solution as follows:

$$\frac{\partial P_{IPFCn}}{\partial \sigma_n} = \frac{V_{2n} V_{IPFCn}}{X_n} \cos\left(\delta_n + \sigma_n\right) = 0 \Rightarrow \sigma_n \rightarrow \left\{-\delta_n \pm \pi/2\right\}. \tag{19.8}$$

On the basis of the above one can say that to achieve a maximal influence of the IPFC device on the transmitted power, the injected voltage $\bar{V}_{IPFCn}$ must be constrained to stay in quadrature with the sending end voltage $\bar{V}_1$ (lead or lag sending end voltage). Therefore on the basis of the equation (19.8) and taking into consideration the equation (19.8) it is evident that the transmission characteristics are the same as for the UPFC device; their up or down displacement depends on the exchanged active power. The above presented capabilities of the IPFC to control the transmitted power can be utilized to increase the transient stability limit as well to improve dynamic stability.

19.2.2. Combined interline power flow controllers

The general form of a combined IPFC is shown in Fig. 19.5 (Strzelecki *et al.*, 2005b). It employs a number of series DC/AC converters, namely SSSCs, each providing series active and reactive compensation for a different transmission line and one shunt DC/AC converter, namely a STATCOM, coupled to the IPFC's common DC link. The shunt converter provides shunt reactive compensation and supplies or absorbs the overall active power deficit of the combined SSSC's and in this way keeps the voltage across the DC link constant. Therefore, the combined IPFC device can control a total of five power system quantities including the voltage magnitude at the bus and independent active and reactive power flow of the transmission lines. In other words, combined IPFC devices possess all the properties of UPFC arrangements but extend the concepts of power flow control to a multilane.

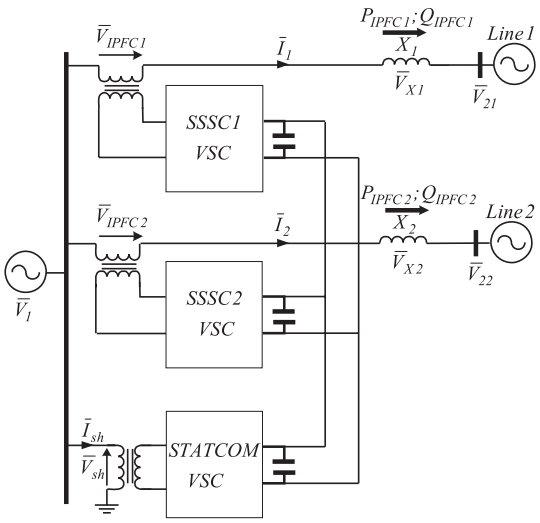


Fig. 19.5. Single line diagram of a combined IPFC

This scheme requires then a rigorous maintenance of the overall power balance at the common DC terminal. The operating constraint representing the active power exchange among the converters via the common DC link is (Strzelecki *et al.*, 2004e):

$$P_{sh} - \sum_{n=1}^{N} P_{\Sigma n} = 0, \tag{19.9}$$

where $P_{\Sigma n}$ is active power supplied/absorbed by the series VSC in n-th transmission line ($n \in 1, 2, \ldots, N$), $P_{sh} = \mathrm{Re}\{\bar{V}_{sh}\bar{I}^{*}_{sh}\}$ is active power supplied/absorbed by the shunt VSC.

Additionally, when the IPFC parameters are controlled in the sense of maximum influence on transmission characteristics and, consequently, on transient stability, the above dependency can be rewritten as

$$P_{sh} = \sum_{n=1}^{N} \frac{V_{IPFCn}}{X_n} \left(V_{2n} - V_1 \cos\left(\delta_n\right)\right). \tag{19.10}$$

Control structure. An overall control design for an IPFC is proposed and shown in Fig. 19.6, which, for simplicity, only shows the control design for two transmission lines where the shunt converter secures only the constant DC link voltage (Strzelecki *et al.*, 2005b). For clarity, only the most important features are shown in this figure, while less important signal processing and limiting functions have been omitted. It should be noted that data for the receiving end bus voltage and current in every transmission line, together with the DC link voltage, are needed for this control design. If the receiving end bus is a local bus and the currents and voltage vectors are local signals, then, theoretically, there is no time delay in getting these signals.

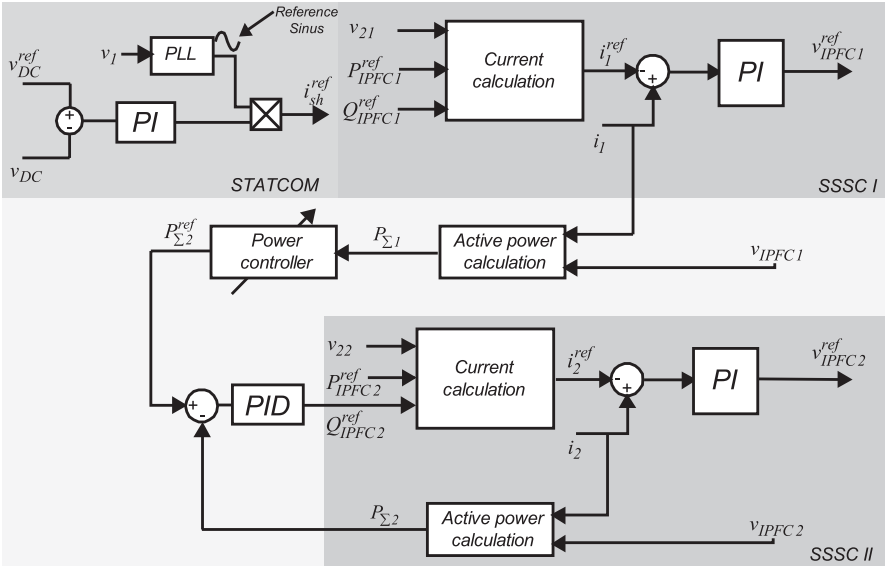


Fig. 19.6. Overall control design for a combined IPFC (phase a)

Because the shunt VSC (STATCOM) secures constant DC link voltage, both series VSCs can almost freely (the SSSC's operation limit is related only to its rating) control power at both receiving end busses. Therefore the controlled voltages V_{IPFCn} injected by both SSSCs are determined through comparison currents, as was described earlier in this section.

However, it will not be possible to secure series active compensation in a situation when the active power demand from both series VSCs exceeds the rating of the STATCOM. In this case, the reactive power flow of both transmission lines will be suboptimized at the nearest value from the reference input along a particular reactive power compensation line.

The above problem, among others, can be solved by the so-called "power controller". Assuming that transmission *Line2* is secondary in relation to *Line1*, to secure in prime *Line1* series active compensation, the "power controller" has to equalize the active power demands of the primary VSC between the STATCOM and the secondary series VSC. In the simplest situation, when the "power controller" transfer function is set to be "*1*", the primary VSC's active power demands will be secured exclusively by the slave VSC. But in the case of many transmission lines, the "power controller" transfer function should be selected in accordance with $P_{IPFCslave}/P_{IPFCmaster}$. The biggest weight, i.e., the biggest contribution to the active power in the DC link should be given to transmission lines with the biggest $P_{IPFCslave}/P_{IPFCmaster}$ ratio.

Computer simulations. In Fig. 19.7, the SSSC's active and reactive power control capabilities, where a constant DC link voltage is maintained by the STATCOM (shunt connected VSC), were investigated.

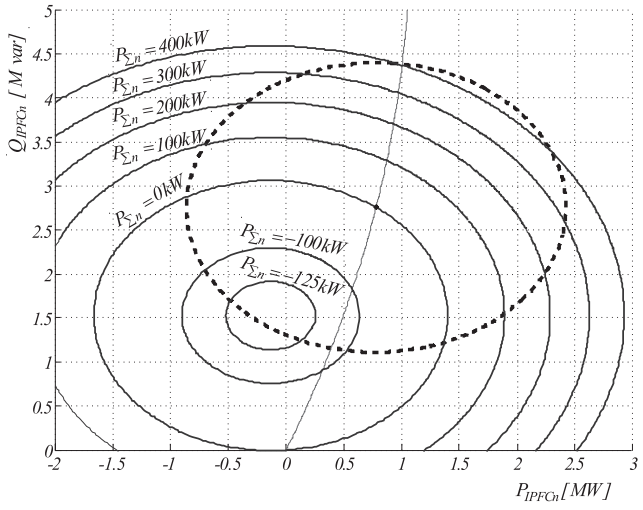


Fig. 19.7. SSSC's power flow control capability with constant DC link voltage maintained by the STATCOM

For clarity, the compensating lines show the SSSC's active and reactive power flow control capabilities with the constant active power $P_{\Sigma n}$ exchanged with the DC link. Additionally, the dashed circle shows SSSC's operation limit related to its power

rating. As one can see, the SSSC allows controlling the active and reactive power delivered to the receiving end bus even without the necessity of exchanging the active power through the DC link $P_{\Sigma n} = 0$, or with a constant demand for a active power $P_{\Sigma n}$.

19.2.3. Interline power flow controllers – probabilistic dimensioning

To date, the choice of power ratings of the shunt converter in an IPFC system has been based on the maximum active power demand of individual series converters (Gyugyi *et al.*, 1998). This deterministic approach has not taken into account the probabilistic nature of the electrical quantities in an EPS. In this section, a new probabilistic approach to assess the power ratings of an IPFC system is proposed. The approach exploits the inherent random nature of electrical quantities such as the voltage and power in a distributed power system (Popczyk, 1991), resulting in considerable reduction of power rating, and hence costs, for the IPFC system, when compared with that provided by the deterministic approach (Strzelecki and Benysek, 2004a; 2004b).

From the previous sections we know that the shunt converter provides shunt reactive compensation, and supplies or absorbs the overall active power deficit of combined SSSCs and in this way keeps the voltage across the DC link constant. Additionally, when the IPFC parameters are controlled, in the sense of a maximum influence on transmission characteristics and, consequently, on transient stability, though without reactive compensation, the power rating of the shunt converter in IPFC system coupling N transmission lines has to be the sum of the maximum active power demand of individual lines. Thus

$$P_{sh} = \sum_{n=1}^{N} P_{\Sigma n}^{\max} = \sum_{n=1}^{N} \left(\frac{V_{IPFCn} V_{2n}}{X_n} \left(1 - \frac{V_1}{V_{2n}} \cos\left(\delta_n\right) \right) \right)^{\max}. \tag{19.11}$$

The overall power of a shunt converter in IPFC system coupling N transmission lines has been determined by the maximum possible active power flowing within different lines. Such an approach is deterministic since the maximum values are readily definable. However, in reality, electrical quantities such as the voltage and current in distributed power systems all manifest a degree of randomness, which can be statistically modeled by some well-known distribution density functions (Popczyk, 1991). Using this probabilistic approach, a new power rating for the shunt converter in an IPFC system can be assessed. Since the new approach embraces the intrinsic natural random variation of electrical quantities, a more realistic power rating than that derived by the classical deterministic approach should be found.

To make a comparison between power ratings of shunt converters obtained by the deterministic and the probabilistic approach, it should be noted that:

- for given N transmission lines, the particular random variables have the same type of distribution,
- power in a deterministic approach is determined according to (19.11),
- power in a probabilistic approach is determined according to (19.12) as follows (Strzelecki *et al.*, 2002):

$$P_{sh}^{prob} = \max\left(\text{abs}\left\langle P_{sh,a}^{prob}, P_{sh,b}^{prob}\right\rangle\right), \tag{19.12}$$

where $P_{sh,a}^{prob}$, $P_{sh,b}^{prob}$ are limits determined for the maximum probability density with a 99.9% level of confidence, as shown in Fig. 19.8.

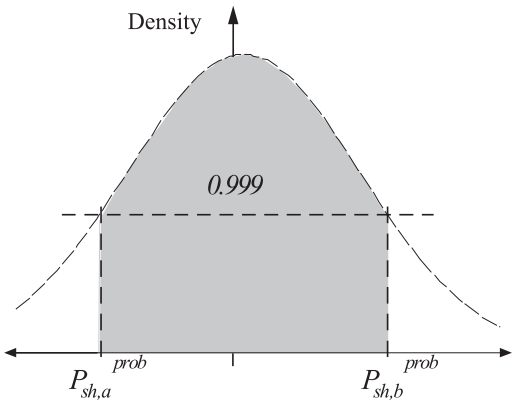


Fig. 19.8. Method for power range determination in the probabilistic approach

Figure 19.9 depicts a series of probabilistic modeling results for an N transmission line system. These results represent the *resultant* distributions of the power rating of the system based on different distribution parameters for V_1 and V_{2n}. These include normal and uniform distributions, and whether they are symmetrical or asymmetrical in relation to the nominal value. As for δ_n, X_n and V_{IPFCn}, uniform distribution is used in all cases shown in Fig. 19.9 for the reasons of the worst-case condition.

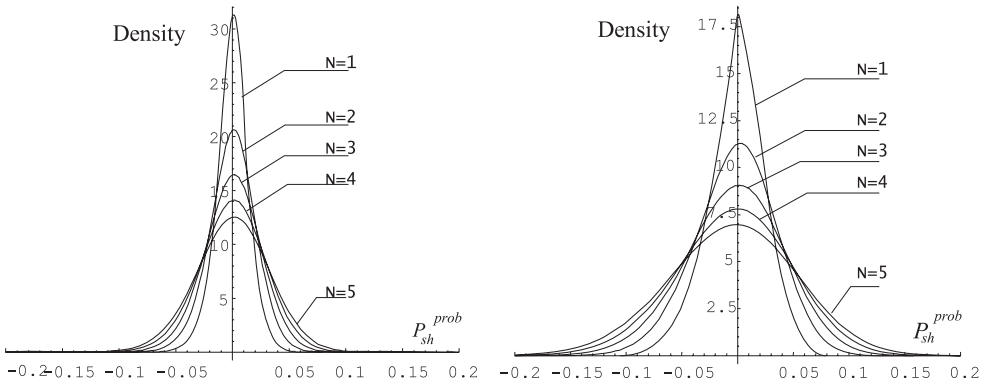


Fig. 19.9. Resultant distributions of the shunt converter's power rating based on its respective distribution parameters: (a) $V_1 \sim N_{1a,b}\,(1; 0.00235)$ and $V_{2n} \sim N_{2n_{a,b}}\,(1; 0.00235)$, (b) $V_1 \sim U_{(0.85V_{1\,\text{nom}}, 1.15V_{1\,\text{nom}})}$ and $V_{2n} \sim (0.85V_{2n,\text{nom}}, 1.15V_{2n,\text{nom}})$

The curve showing the variation of power saving with the number of inverters for each of the four cases is plotted in Fig. 19.10. It can be seen that a maximum

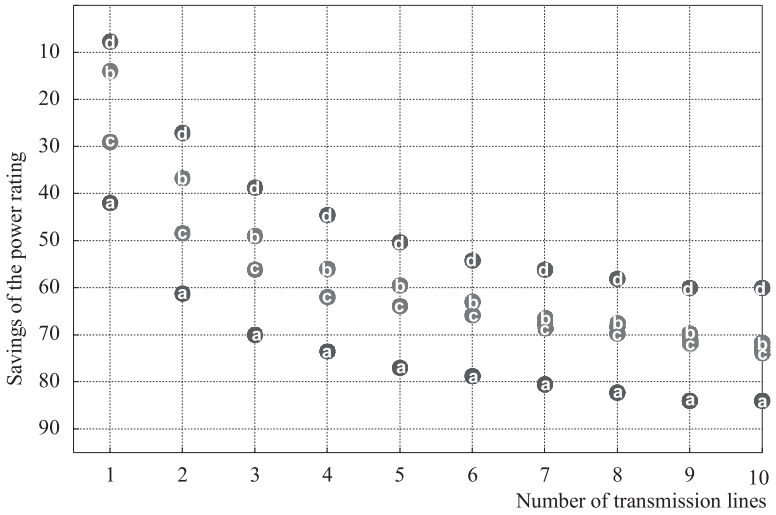


Fig. 19.10. Savings of the parallel inverter's power rating for a given number of transmission lines

saving occurs when the distributions are normal, as shown in the 'a' and 'c' lines; and least if the distribution is uniform, as shown in the 'b' and 'd' lines. Also, the higher the number of transmission lines, the greater the saving. By way of example, using the new approach, 72.1% of saving in the power rating can be obtained for a 5 transmission lines system when the quantities are normally distributed.

IPFC experimental investigations. The experimental investigations were undertaken on a system in Poland over a *4-week* period. Individual transmission lines were connected at the 110 kV/15 kV substation of the power system. Extensive measurements at the 15 kV voltage lines were undertaken, at the rate of 128 measurements per 20 ms, using the modern network analyzer SKYLAB HT9032. The analyzer allows high-speed measurements in accordance with the EN 50160 standard, and can perform many tasks including averaging and harmonics analysis. The substation was feeding normal domestic electrical loads to a residential area. Thus, these measurements would contain inherent random distributions.

The basic electrical quantities measured over a *28-day* period were then processed to produce distribution density curves for some key parameters.

By using numerical methods, and accounting for experimental correlations with the measured quantities, the experimental results for the resultant distributions of the power rating P_{sh}^{prob} of an N transmission line system were determined. Figure 19.11 shows the distributions for the case when the number of transmission lines equals five and ten, respectively. The experimental results show that the distributions appear to be predominately normal rather than uniform, when compared with the theoretical predictions shown in Fig. 19.9. Using the equation (19.12), the experimental results for the variation of savings with the number of lines are determined as shown in Fig. 19.12, with a level of confidence of 0.99.

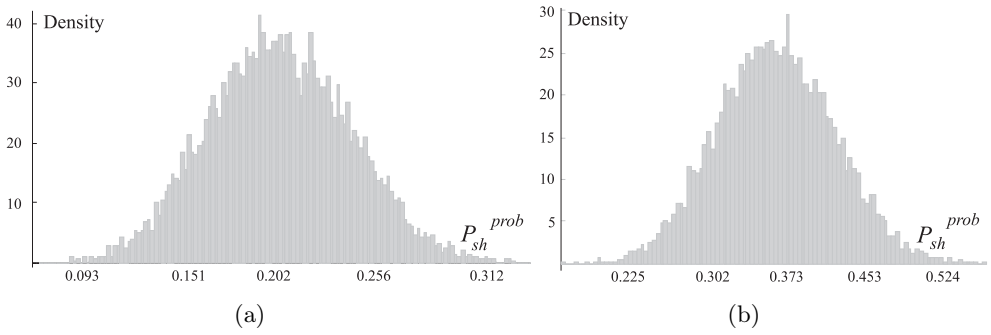


Fig. 19.11. Resultant distributions of the shunt converter's power rating: (a) $N = 5$, (b) $N = 10$

From the curve in Fig. 19.12, it can be confirmed for a one transmission line system ($N = 1$), the IPFC behaves as an "ordinary" UPFC, the power savings are at a level of 7%.

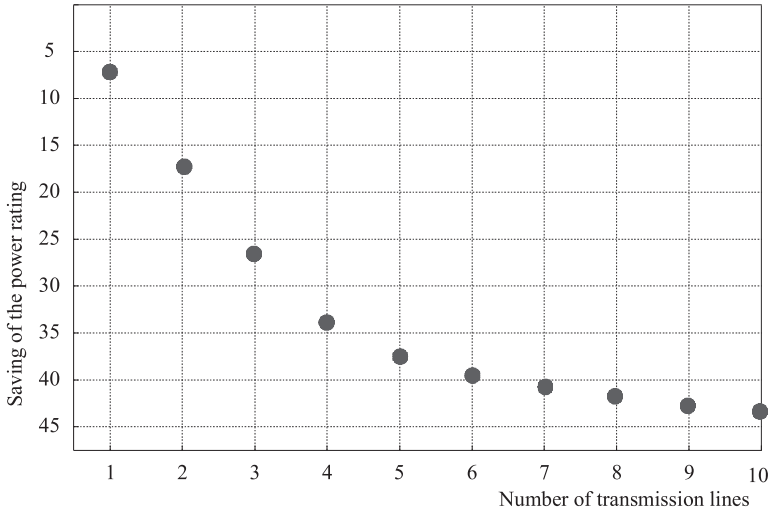


Fig. 19.12. Experimental results for the savings of power rating due to the number of transmission lines

Increasing the number of transmission lines will result in an increase in power rating savings, at about 43% when $N = 10$. It does appear that the experimental savings are less than the theoretical values. This may be due to the actual distributions being exactly of the normal type. Also, the system under the test was supplied with a number of wind turbines, which invariably provide power to the system in a sporadic pattern, thus affecting the theoretical predictions. The pattern between the curves, however, is compellingly similar.

19.3. Compensating type custom power systems

In a similar way as with FACTS devices in transmission systems, custom power systems can be applied to power distribution systems to improve the reliability and quality of the power delivered to end-users. There are widely varying application requirements, such as single or three phase, current or voltage based compensation, therefore the selection of the CUPS for a particular application is as important a task for end-users as it is for utilities.

19.3.1. Single phase UPQC

Figure 19.13 shows selected UPQCs equipped with transformers which are suited to fulfill both voltage and current compensation (Strzelecki *et al.*, 2003f; 2003i). These goals can be achieved with different connections of the DC link capacitor C to the network and different types of connected sources, the voltage and/or current.

Inspection of the figures below shows that the full-bridge circuit requires the largest number of switches – *8* – which is undoubtedly its major disadvantage. However, the source capacitance is not divided. The smallest number of power control devices is required in the half bridge UPQC device. Although there are only two bridge branches, the DC link capacitance consists of two independent capacitors. It is undisputed that one of the most important features of this solution is the possibility to take advantage of very popular three-phase intelligent power modules.

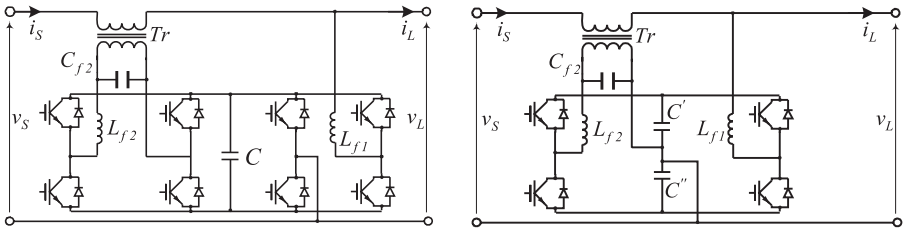
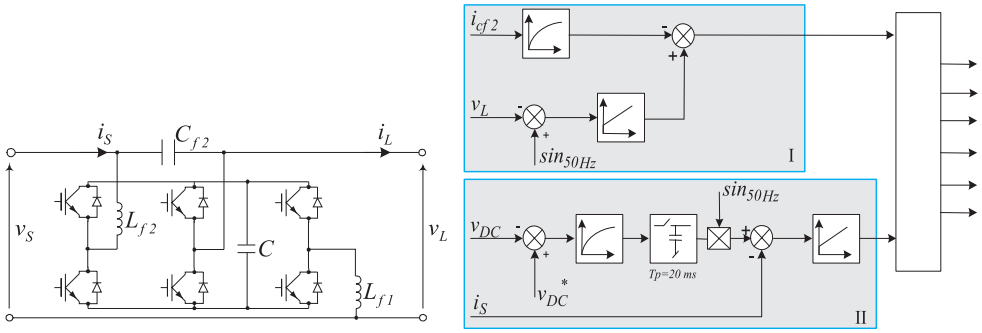


Fig. 19.13. UPQCs on the basis of: (a) full-bridge circuit, (b) half-bridge circuit

The above described configurations require transformers as an interface between the UPQC and the network. This heavy feature is eliminated in the configuration described below (Meckien *et al.*, 1999; 2003), see Fig. 19.14. The examined device is a combination of a series voltage and parallel current sources with a common DC link and can be built on the foundation of three-phase intelligent power modules, which is one of its major advantages. To the disadvantages of this solution one can add the need for a rather large capacitance C_{f2} and the fact that the DC link voltage is not well matched to the required voltage of the series connected source.

The control algorithm consists of two major parts (Strzelecki *et al.*, 2003f; 2003i). The first part, where the *PI* regulator plays the major role, realises load voltage stabilization. The regulator's input signals are the load voltage v_L and the reference sinusoidal wave, and then its output signal is compared to the initially filtered ca-



(a) (b)

Fig. 19.14. (a) Single phase transformer-less UPQC, (b) its control algorithm

pacitor C_{f2} current. This last activity diminishes oscillations in the capacitor C_{f2} voltage.

The second part realises parallel current source control in such a way as to keep the DC link voltage constant. The DC link voltage v_{DC} is compared to its reference value, and after initial filtering the error signal is inserted into the sample-hold block. Because the error signal determines the network current magnitude, thus the invariability of this signal, during one period, guarantees the sinusoidal network current. For the network fundamental frequency 50 Hz this signal is sampled per 20 ms, synchronously with the network voltage 0 pass. The stored current magnitude related signal after multiplication by the reference sinusoidal course becomes the reference curve.

To verify the properties of the proposed solution, a scaled down hardware model was developed. By studying the courses presented in Fig. 19.15 it is possible to say that the examined device fulfils its functions: the network current becomes sinusoidal, even in the case of a strongly deformed load current; the load voltage becomes stabilized in the situation of network voltage magnitude variations.

19.3.2. Three phase UPQC

The 3-phase UPQC presented in Fig. 19.16 consists of series and parallel active power filters, connected by a common DC circuit (Strzelecki *et al.*, 2005a). The SAPF – the v'_C voltage source – filters harmonics and stabilizes at the point of measurement the load voltage v_L, during network voltage v_S changes and deformations. The PAPF – the i'_C current source – filters passive components in the load current i_L. Small filters, L_V–C_V as well as L_I–C_I (along with the dumping circuits R'_V–C'_V as well as R'_I–C'_I) provide the filtration of the harmonics related to PWM control.

The main controller controls the V_{DC} voltage and on the basis of the measured instantaneous i_L, v_S values as well as the references V^*_{DC} and v^*_L calculates the reference current i^*_C and voltage v^*_C waves. The calculations are realized in d–q coordinates, rotating with a frequency $\omega_S = 2\pi/T_S$, where T_S is the period of the network voltage. Unsettling in active power balance causes V_{DC} voltage changes, thus to stabilize this voltage, in the i_C current the controller forces an additional component, in phase with the v_I voltage (in d–q coordinates this is most often a periodic wave).

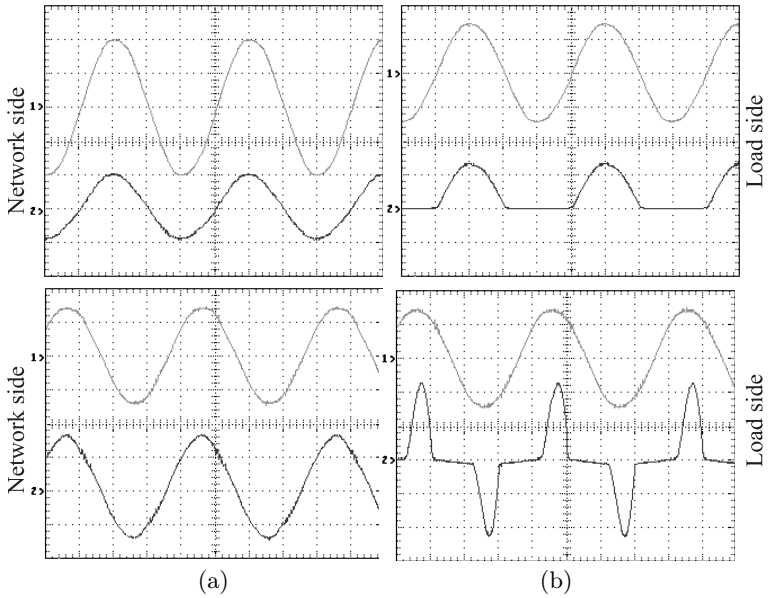


Fig. 19.15. Experimental waveforms: (a) one pulse rectifier, (b) two pulse rectifier with the capacitor filter *Ch1:*50 V/div ⇒ *voltage*, *Ch2:*1 A/div ⇒ *current*; time 5 ms/div

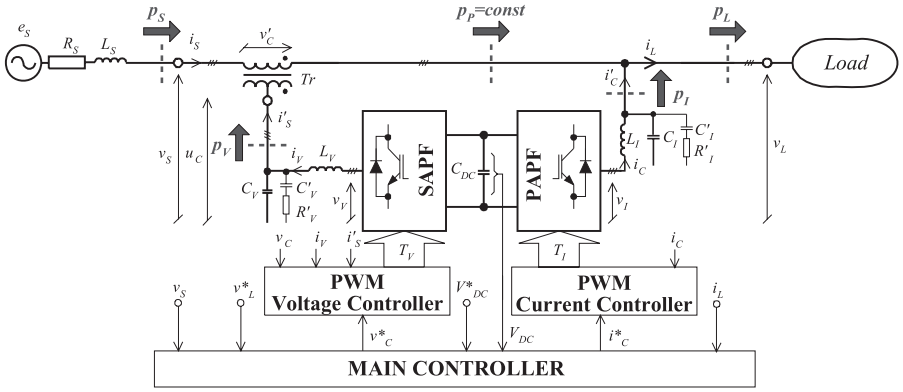


Fig. 19.16. Scheme of the investigated UPQC

Experimental results. In Fig. 19.17, exemplary experimental current and voltage courses, illustrating UPQC steady state filtration capabilities, are introduced. The experimental courses concern the case of an arrangement loaded with a controlled 7 kW, *6-pulse* rectifier ($\alpha = 0°$, $THD(I_L) \approx 28\%$) and supplied with a deformed network voltage, a distortion coefficient $THD(V_S) \approx 9\%$. In this case, both the network current i_S as well as the load voltage v_L are practically sinusoidal. Load changes mainly infect PAPF and SAPF nominal powers (Fig. 19.18). In the general case, PAPF nominal power depends on distortion power, reactive power and load asymmetry power; additionally, SAPF nominal power depends on active components of the load current,

and deformations and asymmetry of the network voltage. Experimental dependencies, introduced in Fig. 19.18, concern the case when the UPQC is loaded with a *6-pulse* rectifier and supplied with a sinusoidal ($THD(V_S) \approx 4\%$), symmetrical, nominal voltage.

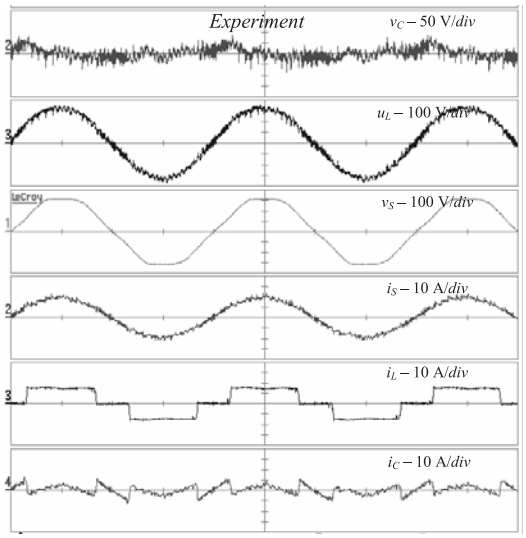


Fig. 19.17. Experimental courses in respect to network voltage and load current harmonics filtration

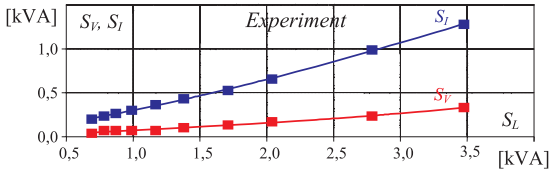


Fig. 19.18. $SAPF(S_V)$ and $PAPF(S_I)$ nominal powers as a function of the load apparent power ($\alpha = 0^\circ$)

Steady states in the experimental investigations contained also an estimation of UPQC potential for the load current as well as network voltage symmetrization. The investigations were carried out in the case of a lowered network (supply) voltage. As the asymmetric load, two *2-pulse* rectifiers, 1.1 kVA each, were used. The asymmetric network (supply) voltage was obtained with the assistance of three independent autotransformers. Exemplary current and voltage courses are introduced in Figs. 19.19 and 19.20.

19.3.3. Voltage active power filter

The basic equivalent circuit of a VAPF is presented in Fig. 19.21 (Strzelecki *et al.*, 2003a, 2004f). In this solution the sine wave voltage source v_V is connected in parallel with the load. Next, the series inductance X_S connects the network voltage v_S to

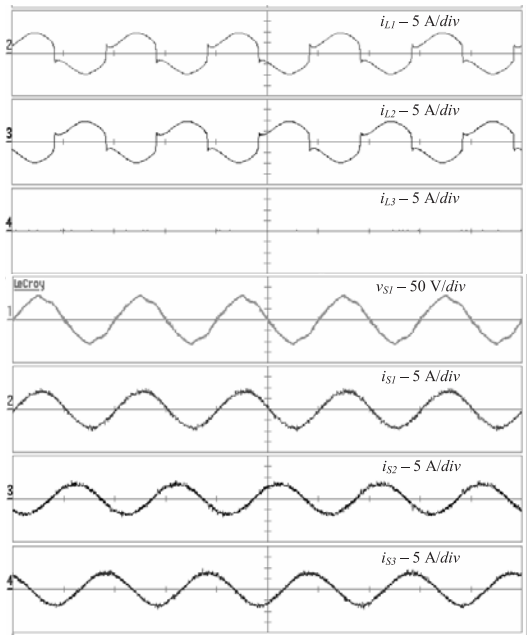


Fig. 19.19. Experimental courses in the situation of a symmetric supply and a non-linear and asymmetric load

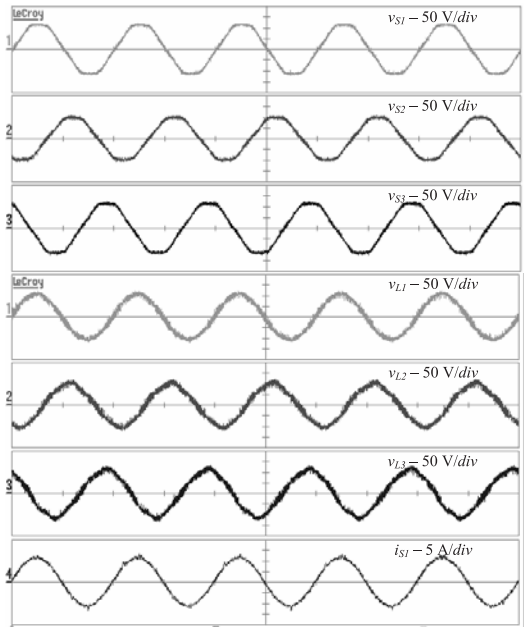


Fig. 19.20. Experimental courses in the situation of an asymmetric supply and a symmetric resistive load

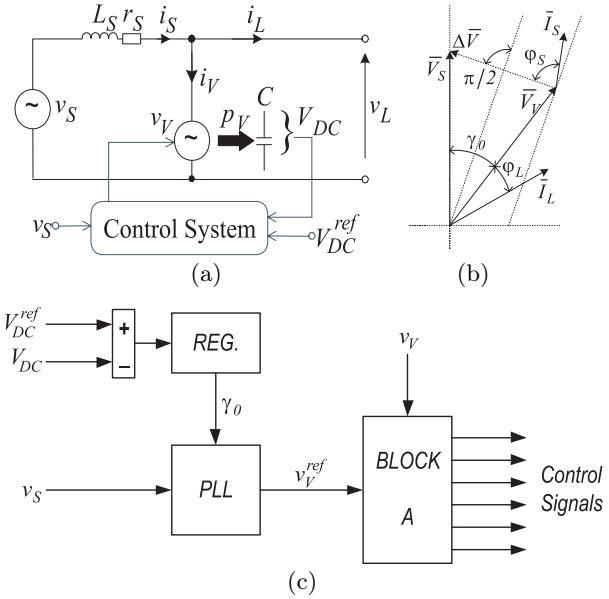


Fig. 19.21. VAPF: (a) One line scheme, (b) vector diagram, (c) simplified control algorithm

the rest of the arrangement. The general function of the voltage source v_V is the removal of the distortions produced by the load from the network, as well as the removal of the load from the distortions generated by the network. The separation is possible because the stabilized and sinusoidal v_V voltage, shifted with the angle γ_0 with regard to the network voltage v_S, secures the sinusoidal and almost in phase with the voltage v_S network current i_S (the v_V voltage source absorbs in the natural way current harmonics produced by the load) and also the stabilized and sinusoidal load voltage.

In the simplified control algorithm, Phose Locked Loop (PLL) generates a signal v_V^{ref} whose frequency is equal to that of the network and whose phase is shifted in relation to v_S with the angle γ_0 proportional to the load active power P_L. The block A stabilizes the load voltage at the reference value by means of a closed-loop control error between v_V and the reference voltage v_V^{ref}. To keep the total DC link voltage at constant value a regulator REG was implemented.

Harmonics separation. The VAPF equivalent circuits for different conditions are presented in Fig. 19.22 (Strzelecki *et al.*, 2003b). In those circuits, X_{Sh} represents series reactance for higher harmonics and v_{Vh} represents the VSC output voltage in the conditions of a distorted network voltage v_{Sh}. Additionally, i_{Sh} and i_{Lh} represent respectively network and load current higher harmonics. The inspection of Fig. 19.22(a) clearly shows that neglecting both sinusoidal sources v_S and v_V, and because $X_{Sh} \gg 0$, the load current is given by $i_{Lh} = i_{Vh}$. On the basis of the above one can conclude that the VAPF under consideration has the effect of "separating" the network from the non-linear loads. In the case of a distorted network voltage (see Fig. 19.22(b)), when the VSC – except that which is sinusoidal with fundamental

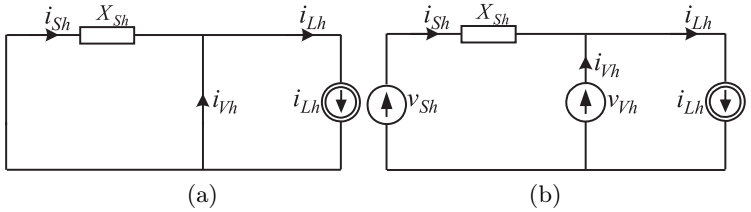


Fig. 19.22. VAPF equivalent circuits for higher harmonics in the case of: (a) sinusoidal network voltage, (b) distorted network voltage

frequency – produces also higher harmonics components just to meet the condition $v_{Sh} = v_{Vh}$, then the load current is given by $i_{Lh} = i_{Vh}$ and the VAPF acts as early "separation" of the network from the non-linear loads.

Reactive power compensation. In the case when the V_V and V_S amplitudes are the same ($V_V = V_S = V$) and neglecting r_s, the angle between the network voltage $\bar{V}_S$ and the current $\bar{I}_S$ is $\gamma_0/2$ and, in consequence, the input power factor PF_S is not unity. On the basis of the above the input power factor can be defined (Strzelecki *et al.*, 2003g; 2004f)

$$PF_S = \sin\left(\frac{\gamma_0}{2} + \varphi_S\right) \underset{r_S \to 0}{\approx} \cos\left(\gamma_0/2\right) = \cos\left(\frac{1}{2}\,\text{arc sin}\,\frac{P_L}{P_{\max}}\right), \tag{19.13}$$

where $P_{\max} = V^2/(\omega L_S)$.

Figure 19.23 presents the relation between the input power factor PF_S and the load active power P_L for different $P_{\max}$ (in relation to the reference power $P_{U(\max)}$). Examining the curves below one can see that when decreasing P_L, the input power factor PF_S increases. Additionally, only in the case of very slight loads is it possible to reach PF_S near unity.

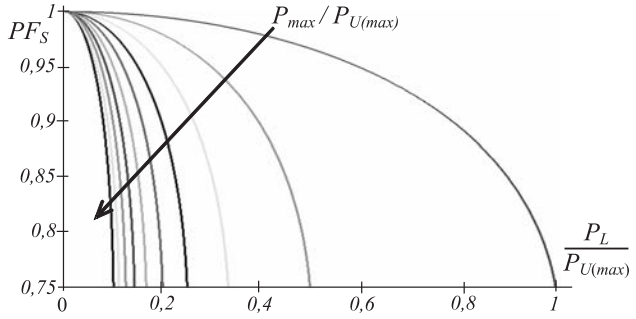


Fig. 19.23. Relation between PF_S and P_L; where $P_{\max}/P_{U(\max)} = 1/10, 2/10, \ldots, 1$

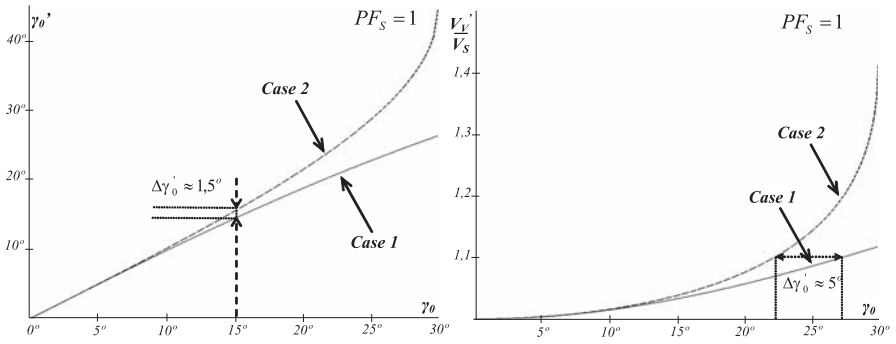
Power factor improvement can be reached, for example, by increasing the V_V voltage amplitude above the network voltage V_S and shifting the angle γ_0 to a new value γ_0' just to keep the following rule true: $P_S = P_L = \text{const}$ (Case 1). On the basis of the above and the assumption $r_s \to 0$, one can define the following (Strzelecki *et al.*,

2003g):

$$V'_{V\ \mathrm{rms}} = V_{S\ \mathrm{rms}}\sqrt{1+\sin^2\gamma_0}, \quad \gamma'_0 = \arcsin\left(\sin\gamma_0 \Big/ \sqrt{1+\sin^2\gamma_0}\right). \tag{19.14}$$

Another case is also possible, i.e., when maintaining unity, the input power factor load active power is variable (Case 2). Then the voltage produced by the VAPF could be defined by the equations (Strzelecki *et al.*, 2003g):

$$V'_{V\ \mathrm{rms}} = V_{S\ \mathrm{rms}} \Big/ \cos\left(\frac{\arcsin(2\sin\gamma_0)}{2}\right), \quad \gamma'_0 = \frac{\arcsin(2\sin\gamma_0)}{2}. \tag{19.15}$$



(a) (b)

Fig. 19.24. Changes of: (a) angle γ'_0, (b) voltage V_V

Examining the above dependencies and curves in Fig. 19.24, one can claim that for low γ_0 angles both cases are pretty much the same. Increasing the γ_0 value above 15°, γ_0 in Case 1 is evidently lower and, in consequence, the V_V voltage and reactive power exchanged between the VAPF and the supply network are lower.

Single phase VAPF – experimental results. To verify the properties of the VAPF device, a down scaled hardware model, presented in Fig. 19.25, was developed. In this circuit, a two level PWM modulation VSC as the v_V voltage source was implemented with a Γ passive filter on its output.

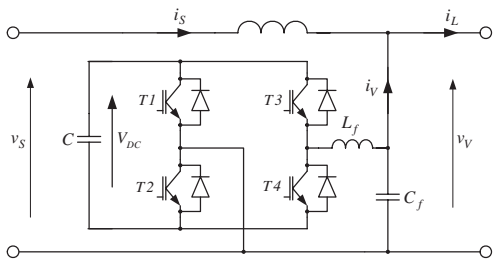


Fig. 19.25. Single phase VAPF

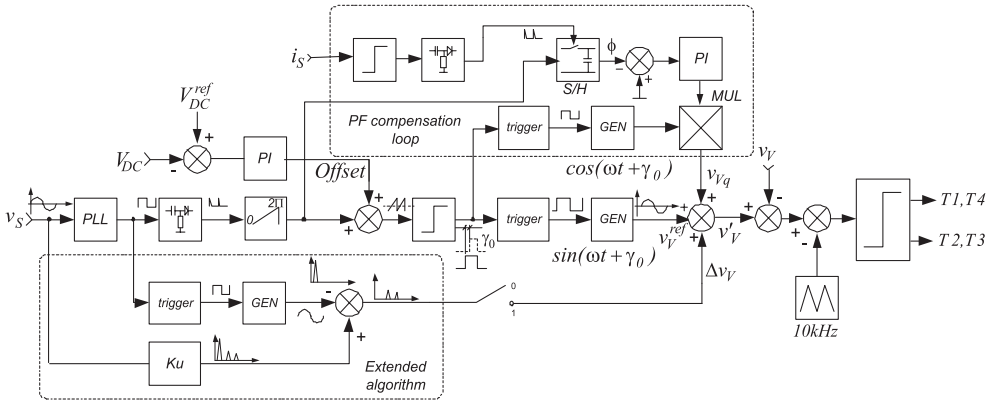


Fig. 19.26. Control diagram

Table 19.1. *THD* coefficient; values obtained for the extended control algorithm are given in parentheses

Load		*THD* [%]			
		i_S	v_S	i_L	$v_V = v_L$
Linear	0.4 kW	13.7 (9.9)	3.6 (6.1)	2.8 (6.3)	4.5 (7.7)
	1 kW	7.9 (5.9)	5.6 (9.2)	4.4 (5.9)	9.1 (15.6)
Non-linear	0.4 kW	11.2 (3.5)	5.1 (6.1)	60.0 (60.8)	8.0 (8.6)
	1 kW	7.7 (3.5)	5.2 (6.6)	44.0 (45.0)	8.6 (8. 8)

Additionally, Fig. 19.26 presents a control diagram (Strzelecki *et al.*, 2003b; 2003g; 2004f). To regulate the DC circuit voltage a *PI* controller was implemented which uses the error between the reference V_{DC}^{ref} and the actual DC voltage V_{DC} as a feedback signal. Next, the control algorithm generates a signal v'_V whose frequency is equal to that of the network and whose phase is shifted in relation to v_S with the angle γ'_0 proportional to the load active power P_L. The reference load voltage is secured by means of a closed-loop control error between v_V and the reference voltage v'_V.

In the case of a distorted network voltage v_S, to avoid network current distortion, there is a need to use an extended algorithm. The extended part has to extract from the distorted network voltage the unneeded components and add to the already shifted basic component (at 50 Hz frequency) of the network voltage. Unfortunately, this solution leads to a distorted load voltage.

Figures 19.27 and 19.28 demonstrate the harmonics separation capability. As one can see, the load current contains a large amount of harmonics due to a two pulse rectifier with capacitor filter; however, the network current is sinusoidal. Because of the distorted network voltage, the extended control algorithm can additionally improve the shape of the network current.

Additionally, Table 19.1 collects *THD* coefficients at characteristic points of the VAPF device.

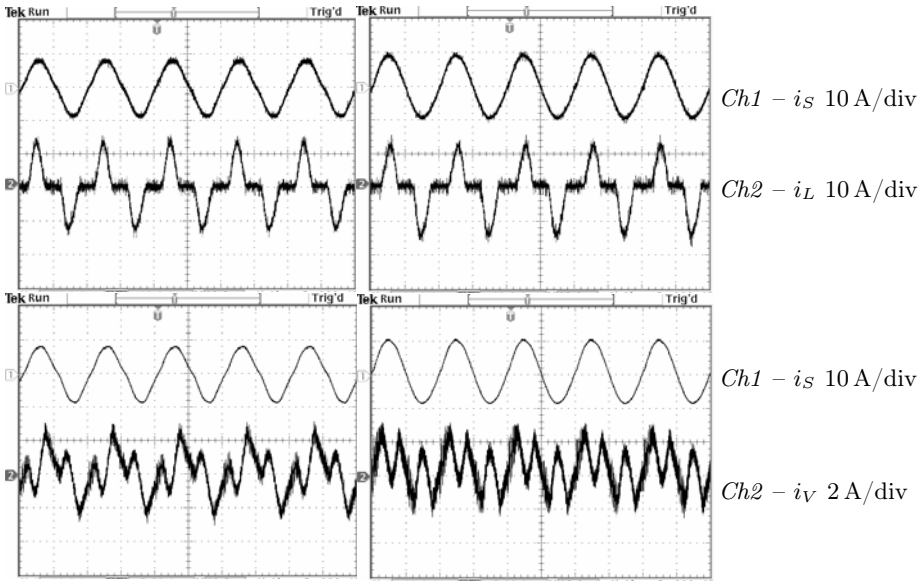


Fig. 19.27. Experimental waveforms ($\Delta t = 10\,\text{ms/div}$) for the non-linear load $P_L = 1\,\text{kW}$: (a) basic control algorithm, (b) extended control algorithm

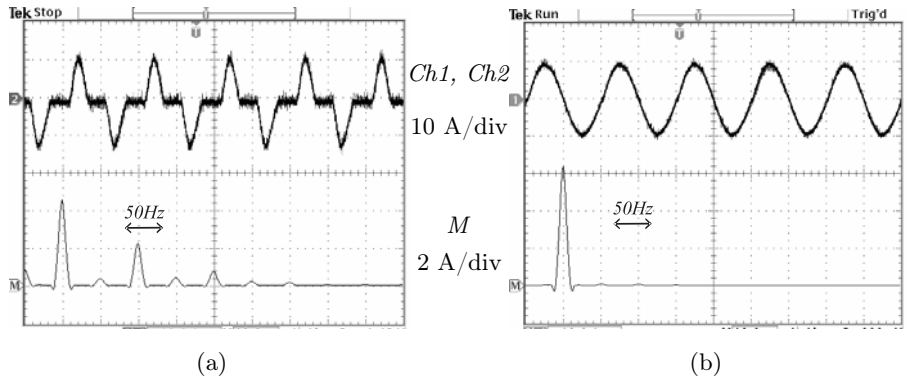
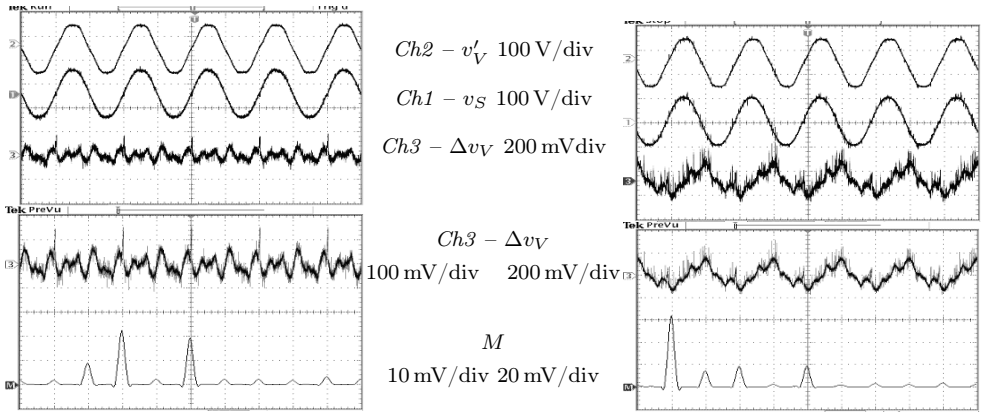


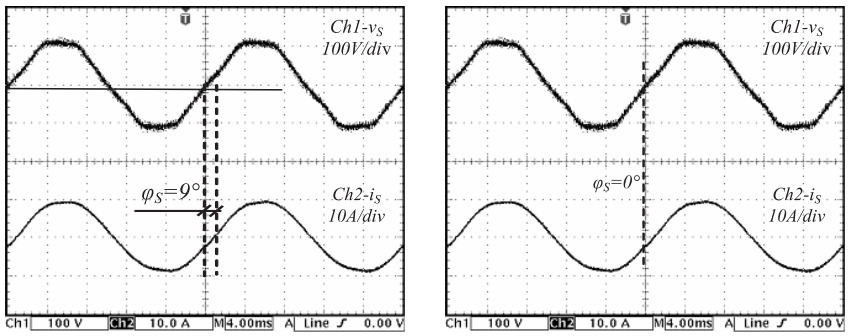
Fig. 19.28. Experimental waveforms ($\Delta t = 10\,\text{ms/div}$) and spectrum for the non-linear load $P_L = 1\,\text{kW}$ (extended control algorithm): (a) load current, (b) network current

Figure 19.29 demonstrates the arrangement capabilities in the case of network voltage magnitude variations. It can be observed that in the case of nominal amplitude, the extended part generates only the signal Δv_V proportional to the network voltage higher harmonics; however, in the case of variable amplitude (3% under the nominal value), the algorithm additionally generates a basic frequency component and thus secures the stabilized load voltage.



(a) (b)

Fig. 19.29. Experimental waveforms ($\Delta t = 10\,\text{ms/div}$) and spectrum in the case of network voltage magnitude variations; non-linear load $P_L = 1\,\text{kW}$ (extended control algorithm): (a) nominal network voltage, (b) reduced network voltage



(a) (b)

Fig. 19.30. VAPF – input power factor correction ($\Delta t = 4\,\text{ms/div}$): (a) without PFS correction, (b) with PFS correction

For the linear load $P_L = 1\,\text{kW}$, the angle between the network voltage $\bar{V}_S$ and the current $\bar{I}_S$ is $\varphi_S = 9°$ and the input power factor PF_S is 0.98. However, activating the power factor compensation loop and thus increasing the V_V voltage amplitude 1.3% above the V_S voltage amplitude increases this value to unity, see Fig. 19.30.

Three phase multilevel VAPF – experimental results. To verify the properties of the three phase VAPF device, a down scaled hardware model was developed. In this circuit, a four level cascaded multilevel converter as the v_V voltage source was implemented. During the study the DC voltages were even, $V_{DC1} = V_{DC2} = V_{DC3} = V_{DC4}$, and on the converter's output a passive Γ filter was implemented.

This time the control algorithm can be divided into two major parts. The first one generates a signal v'_V shifted in relation to v_S with the angle γ'_0 proportional to

the load active power P_L, as was the for a single phase device. Thus in the steady state the total V_{DC} voltage is constant and, in consequence, the average active power exchanged between the VAPF and the network is zero (under the assumption that commutation losses are diminished) (Strzelecki *et al.*, 2003c; 2003d; 2003g; 2004f).

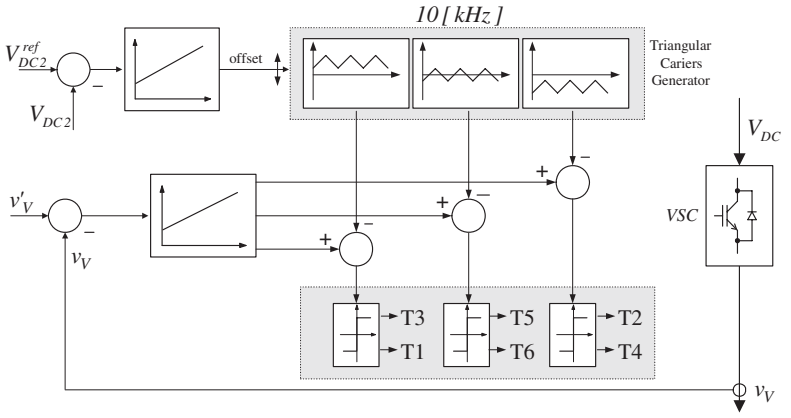


Fig. 19.31. Algorithm for balancing DC voltages (phase a)

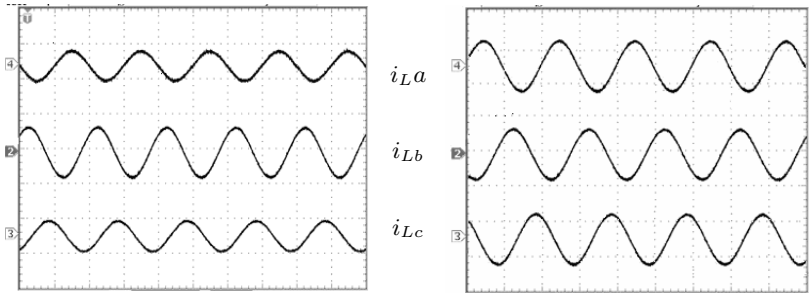
As has been stated earlier, the γ'_0 angle changes have an impact on the real power absorbed or supplied by the VAPF, thus it is possible to regulate the total DC voltage. However, to avoid the problem of unbalanced voltages on the selected capacitors (V_{DC2}, V_{DC3} and V_{DC4}), the control algorithm has to be equipped with an additional second part, see Fig. 19.31. The second part secures the constant and balanced voltages V_{DC2}, V_{DC3} and V_{DC4} as a result of a variable switching strategy of the transistors. A changeable switching strategy can be achieved adding a suitable constant component to the triangular carriers (this does not cause changes on the converter's output voltages or currents). The required constant component can be obtained from the comparison of the reference voltage, in a phase a that will be V_{DC2}^{ref}, with the actual measured value V_{DC2}. The constant component obtained in this manner shifts the triangular waves and in this way changes the switching strategy, which finally leads to the equalization of the DC voltages. Additionally, in (Kot and Benysek, 2001; Strzelecki *et al.*, 2001b), appropriate methods of avoiding problems of unbalanced DC voltages for other types of multilevel converters are introduced.

In the figures below experimental waveforms obtained for two different load types, linear (R–L load) and non-linear (6-pulse rectifier with R–L load), are presented.

Figure 19.32 demonstrates the arrangement's capability for balancing the network in the conditions of unbalanced loads. Because the VAPF produces sinusoidal balanced voltages, network currents are, also balanced. This condition is satisfied whether the load currents are balanced or not, because the network currents are determined only by the network voltages, the voltage as well as the X_S reactance. However, in the conditions of unbalanced loads, the i_V currents are also unbalanced, which results in a 100 Hz component in the DC voltages.

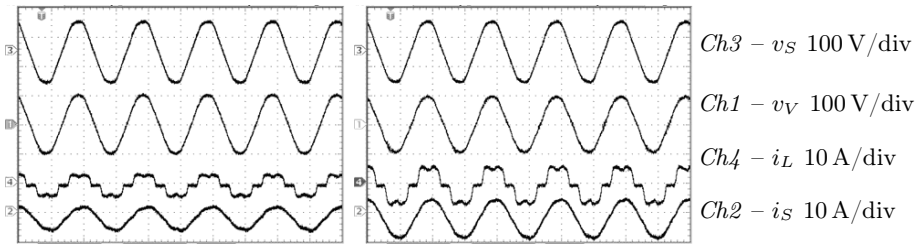
Figure 19.33 demonstrates the harmonics separation capability. As one can see, the load current contains a large amount of harmonics due to the six pulse rectifier with a resistive-inductive load; however, the network current is almost sinusoidal.

Additionally, Table 19.2 presents the *THD* coefficients at characteristic points of the VAPF device.



(a) (b)

Fig. 19.32. Experimental current waveforms ($\Delta t = 10\,\text{ms/div}$, $10\,\text{A/div}$) for a linear unbalanced load: (a) load side, (b) network side



(a) (b)

Fig. 19.33. Experimental waveforms ($\Delta t = 10\,\text{ms/div}$) for a non-linear balanced load: (a) $P_L = 0.8\,\text{kW}$, (b) $P_L = 1.2\,\text{kW}$

Table 19.2. *THD* coefficients

		THD [%]			
		i_S	v_S	i_L	$v_V = v_L$
Non-linear	$P_L = 0.8$ kW	3.3	3.3	25.3	2.9
load	$P_L = 1.2\,\text{kW}$	2.6	3.5	24.2	3.7

Furthermore, Fig. 19.34 demonstrates, in the case of the non-linear balanced load, the DC voltages.

19.4. Future works

As distributed resources hardware becomes more reliable and economically feasible, there is a trend to connect DR units to the existing utilities to serve different purposes and offer more possibilities to end-users, such as:

- improving the availability and reliability of electrical power,

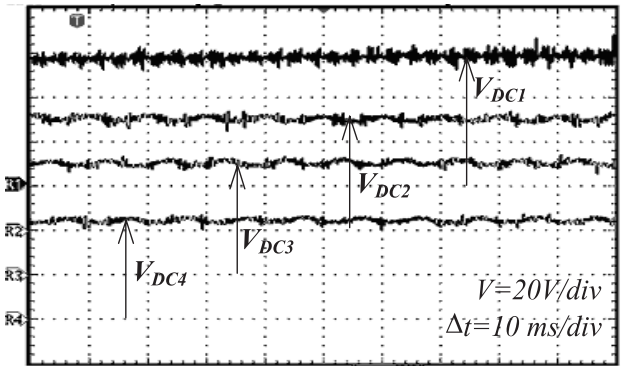


Fig. 19.34. DC voltages

- peak load shaving,
- energy cost savings,
- selling power back to utilities or other users,
- reactive power compensation,
- mitigation of harmonics and a voltage sag.

However, a wide range of system issues arise when DR units attempt to connect to the EPS. The major issues regarding the connection of DRs include protection, power quality, and system operation. These issues are barriers that limit the penetration of DRs into the EPS. Future electrical power systems should be versatile and flexible so that electrical energy could be freely generated, transmitted, distributed, and consumed.

The overall general aim of future investigations is to lower or eliminate these technological barriers by addressing the connection issues between DRs and the EPS. The main objectives will be to investigate the impact of DRs on the EPS, and the reciprocal impact and effects of EPS events on the operation of DRs.

The investigations will have three main scientific objectives that concern acquiring an understanding of the integration and connection between the DR and the EPS. These are:

- in-depth investigation into the impact of DRs on the EPS, and the effects of EPS events on the operation of DR units,
- in-depth investigation into various DRs/EPS connection issues due to various distributed sources, such as fuel cells and the renewables,
- in-depth investigation into advanced coordinated control of DRs within the EPS.

These scientific objectives will provide underpinnings that are vital for the exploitation and advancement of DR technology. There are three main technological objectives involved in acquiring the capabilities of integrating the DR and the EPS, introduced below.

Definition of DR/EPS connection interfaces. At present, distribution system voltage regulation design is based on relatively predictable daily and seasonal changes in load patterns. Without DRs, power flow is mostly unidirectional, and monotonically

decreasing in real power magnitude with increasing the distance from the substation. The addition of DRs units to the EPS can radically shift power flow patterns, which leads to complex and unpredictable load flow dynamics. This will make it difficult to maintain adequate voltage regulation, among other things. It is therefore essential to formulate new definitions for the DRs/EPS connection interface by developing some specific requirements from the point of view of a system's voltage regulation, transient response and fault behavior; reclosing; anti-islanding; power systems dynamics and stability.

Design and construction of the DRs/EPS interface. This concerns the design and construction of an interface that possesses all the quality of delivery requirements. Industrial and domestic applications require the DRs/EPS connection interface to generate high-quality electrical power, possibly close to the perfect sinusoid. This requires the invention of new power electronics topologies and hardware to provide solutions for a low harmonic content and minimum ElectroMagnetic Interference (EMI). It is proposed that while the existing matrix and multilevel converter techniques are used as the baseline, novel topologies and control must be developed to address specific system requirements within a DR, and characteristics of the DR sources these converters use. Matrix converter techniques offer the ability to convert directly from AC to AC without the intermediate DC stage. The particular advantage of this approach is the elimination of the DC link capacitor. Because of high generator output frequency, integral pulse operation may be possible, leading to a significant reduction in losses. Multilevel converters offer, among other significant advantages, reduced harmonic distortion and EMI. Other advantages inherent in the topology are reduced switching losses and the possibility of using lower rate power devices. Lower device ratings imply faster switching speed and lower on-state losses. Special attention will be paid to cascaded multilevel converters, with several separate DC sources. This arrangement should be the perfect DRs/EPS interface, because several batteries, fuel cells, solar cells, wind turbines, and micro-turbines can be connected through a multilevel converter to feed a load or the AC grid without voltage balancing problems.

Additionally, the development of DRs/EPS connection interfaces equipped with storage technologies will be the object of future investigations (Kolluri, 2002). Special attention will be paid to the batteries of super-capacitors.

The above mentioned DRs/EPS connection interfaces, equipped with modern power electronics arrangements (e.g., multilevel converters) and ESS provide significant opportunities concerning the control of distributed power systems. These topologies are attractive for controlling the frequency, voltage output (including the phase angle), real and reactive power flow at a DC/AC interface, system dynamic behavior, and for reducing power quality problems, such as voltage harmonics, voltage imbalance, or a sag.

Because of this, a new power delivery concept has arisen and will be investigated. This concept comes from the assumption that in the near future more DRs will be equipped with modern power electronics interfaces, "wind" with multi-terminal DC systems, etc. Therefore, the duties of FACTS or CUPS arrangements will be intercepted by DRs with modern interfaces, as in Fig. 19.35.

Development of a Coordinated Control System (CCS) in DRs. To enable future EPS systems to absorb high penetration of DRs, the connection interface must in-

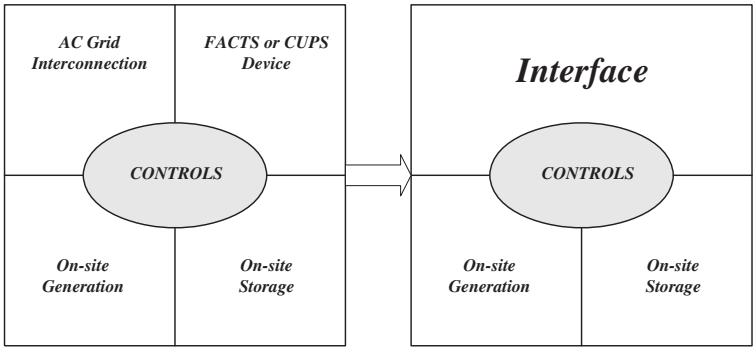


Fig. 19.35. New power delivery concept

corporate a CCS that makes the overall distribution network act proactively and collectively with Load Top Changer (LTC), Static Var Regulator (SVR), load changes and other environment distributions. The dispersed nature of the CCS within the future DRs/EPS means that modern communications and real time control technologies should be employed. The CCS would be based on a collection of local controllers connected together via the Internet. These local controllers would be used to monitor and log the power quality output from DRs.

References

Akagi H. (1994): *Trends in active power line conditioners.* — IEEE Trans. Power Electronics, Vol. 9, No. 3, pp. 263–268.

Akagi H. (1995): *New trends in active filters.* — Proc. 6th European Conf. *Power Electronics and Applications*, Sevilla, Spain, pp. 17–26.

Akagi H. (1996): *New trends in active filters for power conditioning.* — IEEE Trans. Industry Applications, Vol. 32, No. 6, pp. 1312–1322.

Akagi H. and Fujita H. (1995): *A new power line conditioner for harmonic compensation in power systems.* — IEEE Trans. Power Delivery, Vol. 10, No. 3, pp. 1570–1575.

Aredes M. and Heumann K. (1996): *A unified power flow controller with active filtering capabilities.* — Proc. 8th Int. Conf. *Power Electronics and Motion Control*, Prague, Czech Republic, Vol. 3, pp. 139–144.

Aredes M., Heumann K. and Watanabe E. (1998): *An universal active power line conditioner.* — IEEE Trans. Power Delivery, Vol. 13, No. 2, pp. 1453–1460.

Arrillaga J. (1999): *HVDC Transmission.* — London: IEE Publications.

Asplund G., Eriksson K. and Svensson K. (1998): *HVDC light: DC transmission based on voltage sourced converters.* — ABB Review, No. 1, pp. 4–9.

Chen Y., Wolanski Z. and Oil B. (2000): *Unified power flow controller (UPFC) based on chopper stabilized diode-clamped multilevel converters.* — IEEE Trans. Power Electronics, Vol. 15, No. 2, pp. 258–267.

Chistl N., Hedin R., Sadek K. and Lutzelberger P. (1992): *Advanced series compensation (ASC): with thyristor controlled impedance.* — Int. Council on Large Electric Systems paper 14/37/38-05, Paris session.

CIGRE (1999): *Custom power: State of the art.* — Working Group 14.31.

Ekanayake J. and Jenkins N. (1996): *A three-level advanced static var compensator.* — IEEE Trans. Power Delivery, Vol. 11, No. 1, pp. 540–545.

Elnady A., Goauda A. and Salama M. (2001): *Unified power quality conditioner with a novel control algorithm based on wavelet transform.* — Proc. Canadian Conf. *Electrical and Computer Engineering*, Quebec City, Canada, Vol. 2, pp. 1041–1045.

Fujita H. and Akagi H. (1998): *Unified power quality conditioner: the integration of series and shunt active filter.* — IEEE Trans. Power Electronics, Vol. 13, No. 2, pp. 315–322.

Fujita H., Watanabe Y. and Akagi H. (1999): *Control and analysis of a unified power flow controller.* — IEEE Trans. Power Electronics, Vol. 14, No. 6, pp. 1021–1027.

Ghosh A. and Ledwich G. (2001): *A unified power quality conditioner (UPQC) for simultaneous voltage and current compensation.* — Electric Power Systems Research, Vol. 59, No. 1, pp. 55–63.

Ghosh A. and Ledwich G. (2002): *Power Quality Enhancement Using Custom Power Devices.* — Boston: Kluwer Academic Publishers.

Greczko E., Benysek G. and Kot E. (2001): *Properties of the active filters constructed on the base of the multilevel converters.* — Technicna Elektrodinamika, No. 3, pp. 13–18, (in Russian).

Gyugyi L. (1988): *Power electronics in electric utilities: static var compensators.* — Proc. IEEE, Vol. 76, No. 4, pp. 483–494.

Gyugyi L. (2000): *Converter-based FACTS technology: electric power transmission in the 21st century.* — Proc. Int. *Power Electronics Conf.*, Tokyo, Japan, Vol. 1, pp. 15–26.

Gyugyi L., Kalyan K. and Schauder C. (1998): *The interline power flow controller concept: a new approach to power flow management in transmission systems.* — IEEE Trans. Power Delivery, Vol. 14, No. 3, pp. 1115–1122.

Gyugyi L., Schauder S., Williams S. and Rietmann T. (1995): *The unified power flow controller: a new approach to power transmission control.* — IEEE Trans. Power Delivery, Vol. 10, pp. 1085–1097.

Hingorani N. (1993): *Flexible AC transmission systems.* — IEEE Spectrum, Vol. 30, No. 4, pp. 41–48.

Hingorani N. (1996): *High-voltage DC transmission: A power electronics workhorse.* — IEEE Spectrum, Vol. 33, No. 4, pp. 63–72.

Hingorani N. (1998): *Power electronics in electric utilities: role of power electronics in future power systems.* — Proc. IEEE, Vol. 76, No. 4, pp. 481–482.

Hingorani N. and Gyugyi L. (2000): *Understanding FACTS: Concepts and Technology of Flexible AC Transmission Systems.* — New York: IEEE.

IEEE (1987): *Application of static var systems for system dynamic performance.* — Special publication No. 87TH1087-5-PWR.

IEEE (1995): *Static var compensator: models for power flow and dynamic performance simulation.* — IEEE Trans. Power Systems, Special Stability Controls Working Group, Vol. 9, No. 1, pp. 229–240.

IEEE/CIGRE, (1995): *FACTS overview.* — IEEE Service Center, Special issue 95-TP-108, Piscataway, New York.

Jeon S. and Cho G. (1997): *A series-parallel compensated uninterruptible power supply with sinusoidal input current and sinusoidal output voltage.* — Proc. 28th IEEE *Power Electronics Specialists Conf.*, St. Louis, USA, pp. 297–303.

Keri A., Ware B., Byron R. and Chamia M. (1992): *Improving transmission system performance using controlled series capacitors.* — Int. Council on Large Electric Systems paper 14/37/38-07, Paris session.

Kolluri S. (2002): *Application of distributed super-conducting magnetic energy storage systems (D-SMES) in the energy system to improve voltage stability.* — Proc. IEEE-Power Engineering Society Winter Power Meeting, pp. 838–841.

Kot E. and Benysek G. (2001): *Analysis of DC link capacitor voltage balance in cascade parallel active power filters.* — Proc. 2nd Int. Conf. *Power Electronics Devices Compatibility*, Zielona Góra, Poland, pp. 120–126.

Kot E., Baranowski A. and Benysek G. (2000): *Comparative analysis of the parallel active filters on base of the multilevel inverters.* — Proc. 11th Int. Conf. *Electrical Drives and Power Electronics*, Zagreb, Croatia, pp. 38–43.

Liang Y. and Nwankpa C. (2000): *A power-line conditioner based on flying-capacitor multilevel voltage-source converter with phase-shift SPWM.* — IEEE Trans. Industry Applications, Vol. 36, No. 4, pp. 965–971.

Meckien G. and Strzelecki R. (2002): *Single phase active power line conditioners – without transformers.* — Proc. 10th Int. Conf. *Power Electronics and Motion Control*, Cavtat & Dubrovnik, Croatia, pp. 44–50.

Meckien G., Mućko J. and Strzelecki R. (2003): *Single phase, transformer-less electrical energy conditioner.* —Przegląd Elektrotechniczny, No. 2, pp. 121–123, (in Polish).

Meckien G., Strzelecki R. and Klytta M. (1999): *Single-phase ALPC controllers.* — Proc. 1st Int. Conf. *Power Electronics Devices Compatibility*, Zielona Góra, Poland, pp. 48–58.

McHattie R. (1998): *Dynamic voltage restorer: The customer's perspective.* — Institute of Electrical Engineers Colloquium on Dynamic Voltage Restorers, digest no. 98/189.

Mohan N., Undeland T. and Robbins W. (1995): *Power Electronics, Converters, Applications, and Design.* — New York: John Wiley & Sons, (2nd edition).

Moran L., Ziodas P. and Joos G. (1989): *Analysis and design of a novel 3-phase solid-state power factor compensator and harmonic suppressor system.* — IEEE Trans. Industry Applications, Vol. 25, pp. 609–619.

Moran L., Fernandez L. and Dixon J. (1997): *A simple and low-cost control strategy for active power filters connected in cascade.* — IEEE Trans. Industrial Electronics, Vol. 44, No. 5, pp. 402–408.

Peng F. (1998): *Application issues of active power filters.* — IEEE Industry Application Magazine, Vol. 4, No. 5, pp. 21–30.

Peng F., Akagi H. and Nabae A. (1986): *A study of active power filters using quad-series voltage-source PWM converters for harmonic compensation.* — IEEE Trans. Industry Applications, Vol. 22, No. 3, pp. 460–465.

Peng F. and Lai J. (1996): *Application considerations and compensation characteristics of shunt active and series active filters in power systems.* — Proc. 7th Int. Conf. *Harmonics and Quality Power*, Las Vegas, USA, pp. 12–20.

Pengcheng Zhu. (2003): *A novel control scheme in 2-phase unified power quality conditioner.* — Proc. 29th Annual Conf. IEEE *Industrial Electronics Society*, Jacksonville, USA, Vol. 2, pp. 1617–1622.

Popczyk J. (1991): *Probabilistic Models in Electrical Systems.* — Warsaw: WNT, (in Polish).

Reed G., Paserba J., Croasdaile T. and Takeda M. (2001): *STATCOM application at VELCO Essex substation.* — Proc. IEEE-Power Engineering Society Transmission & Distribution Conf. and Exposition, Atlanta, USA, pp. 1133–1138.

Renz B. (1999): *AEP unified power flow controller performance.* — IEEE Trans. Power Delivery, Vol. 14, No. 4, pp. 1374–1381.

Renz B., Keri A., Mehraban A. and Kessinger J. (1998): *World's first unified power flow controller on the AEP system.* — Int. Council on Large Electric Systems paper 14-107, Paris session.

Schauder C., Gyugyi L., Lund M. and Hamai D. (1998a): *Operation of the unified power flow controller (UPFC) under practical constraints.* — IEEE Trans. Power Delivery, Vol. 13, pp. 630–639.

Schauder C., Stacey E., Lund M. and Gyugyi L. (1998b): *AEP UPFC project: Installation, commissioning and operation of the* ±160 *MVA STATCOM (phase 1).* — IEEE Trans. Power Delivery, Vol. 13, No. 4, pp. 1530–1535.

Singh B., Al-Haddad K. and Chandra A. (1999): *A review of active filters for power quality improvement.* — IEEE Trans. Industrial Electronics, Vol. 46, No. 5, pp. 960–971.

Song Y. and Johns A. (1999): *Flexible AC Transmission Systems (FACTS).* — London: IEE Power and Energy series.

Strzelecki R., Benysek G. and Bojarski J. (2001a): *Interline power flow controller.* — Proc. 5th Domestic Conf. *Control in Power Electronics and Electric Drives*, Arturówek, Poland, Vol. 2, pp. 591–596, (in Polish).

Strzelecki R., Benysek G., Rusiński J. and Kot E. (2001b): *Analysis of DC link capacitor voltage balance in multilevel active power filters.* — Proc. 9th European Conf. *Power Electronics and Applications*, Leoben, Austria, pp. 295–301.

Strzelecki R., Benysek G., Fedyczak Z. and Bojarski J. (2002): *Interline power flow controller-probabilistic approach.* — Proc. 33rd IEEE *Power Electronics Specialists Conf.*, Cairns, Australia, Vol. 2, pp. 1037–1042.

Strzelecki R., Benysek G., Rusiński J. and Jarnut M. (2003a): *Active power filter - new control system and topology.* — Proc. Int. Conf. *Marine Electrical Technologies*, Edinburgh, UK, pp. 99–106.

Strzelecki R., Supronowicz H., Jarnut M. and Benysek G. (2003b): *1-phase active power line conditioner.* — Proc. 10th European Conf. *Power Electronics and Applications*, Toulouse, France, pp. 495–501.

Strzelecki R., Jarnut M., Kot E. and Benysek G. (2003c): *Multilevel voltage source power quality conditioner.* — Proc. 34th IEEE *Power Electronics Specialists Conf.*, Acapulco, Mexico, pp. 1043–1048.

Strzelecki R., Benysek G., Jarnut M. and Kot E. (2003d): *Controlled voltage sources in electrical energy conditioning arrangements.* — Proc. 4th Domestic Conf. *Selected Problems of Electrical Engineering and Electronics*, Jadwisin, Poland, Vol. 2, pp. 287–294.

Strzelecki R., Dębicki H., Benysek G. And Jarnut M. (2003e): *Control in simple scheme of the network voltage conditioner.* — Proc. 10th Domestic Conf. *Basic Problems of Power Electronics and Electro-mechanics*, Wisła, Poland, Vol. 18, pp. 91–94.

Strzelecki R., Tunia H., Jarnut M. and Benysek G. (2003f): *Transformerless 1-phase active power line conditioners.* — Proc. 34th IEEE *Power Electronics Specialists Conf.*, Acapulco, Mexico, pp. 321–326.

Strzelecki R., Jarnut M., Kot E. and Benysek G. (2003g): *Voltage source power line conditioners.* — Proc. 3rd Int. Workshop *Compatibility in Power Electronics*, Gdańsk, Poland, pp. 152–161.

Strzelecki R., Benysek G. and Noculak A. (2003h): *Utilization of the power electronics arrangements in electrical power system.* — Przegląd Elektrotechniczny, No. 2, pp. 41–48, (in Polish).

Strzelecki R., Jarnut M. and Benysek G. (2003i): *Active electrical energy conditioners for individual customers.* — Proc. Domestic Conf. *Progresses in Applied Electrical Engineering*, Kościelisko, Poland, Vol. 1, pp. 27–34.

Strzelecki R., Jarnut M. and Benysek G. (2004a): *Modified voltage source active power filter.* — Proc. 11th Int. Conf. *Power Electronics and Motion Control*, Riga, Latvia, pp. 876–892.

Strzelecki R., Jarnut M. and Benysek G. (2004b): *Voltage power line conditioner VPLC - input power factor correction solutions.* — Proc. 9th Int. *Baltic Electronics Conf.*, Tallinn, Estonia, pp. 339–342.

Strzelecki R., Jarnut M. and Benysek G. (2004c): *Voltage source conditioners property.* — Proc. 6th Int. Conf. *Unconventional Electromechanical and Electrical Systems*, Alushta, Ukraine, pp. 767–772.

Strzelecki R., Smereczyński P. and Benysek G. (2004d): *Static properties of the interline power flow controllers.* — Technicna Elektrodinamika, No. 1, pp. 63–69.

Strzelecki R., Smereczyński P. and Benysek G. (2004e): *Interline power flow controllers - energetic properties.* — Proc. 11th Int. Conf. *Power Electronics and Motion Control*, Riga, Latvia, pp. 893–899.

Strzelecki R., Benysek G., Jarnut M. and Kot E. (2004f): *Voltage source power line conditioners.* — Quality and Utilization of Electrical Energy, Vol. 10, pp. 13–24, (in Polish).

Strzelecki R. and Benysek G. (2004a): *Probabilistic method for power flow controllers dimensioning.* — Proc. 4th Int. Conf. *Electric Power Quality and Supply Reliability*, Pedase, Estonia, pp. 149–156.

Strzelecki R. and Benysek G. (2004b): *Conceptions and properties of the arrangements in distributed electrical power systems.* — Proc. 3rd Domestic Conf. *Materials and Technologies in Electrical Engineering*, Gorzów Wlkp., Poland, pp. 241–248, (in Polish).

Strzelecki R. and Sozański K. (2001): *Active filters in supplying networks and drives.* — Proc. 4th Int. Conf. *Modern Supplying Arrangements in Power Systems*, Świerże Górne, Poland, pp. 17.1–17.12, (in Polish).

Strzelecki R. and Supronowicz H. (1999): *Harmonic Filtration in AC Power Systems.* — Toruń: Adam Marszałek Publishing House (2nd edition), (in Polish).

Strzelecki R. and Supronowicz H. (2000): *Power factor in AC supply systems and improvements methods.* — Warsaw: Publishing House of the Warsaw University of Technology, (in Polish).

Strzelecki R., Benysek G., Rusiński J. and Dębicki H. (2005a): *Modeling and experimental investigation of the small UPQC systems.* — Proc. 4th Int. Workshop *Compatibility in Power Electronics*, Gdynia, Poland, pp. 162–177.

Strzelecki R., Smereczyński P. and Benysek G. (2005b): *Interline Power Flow Controller – properties and control strategy in dynamic states*. — Proc. 4th Int. Workshop *Compatibility in Power Electronics*, Gdynia, Poland, pp. 178–185.

Strzelecki R., Benysek G., Jarnut M. and Rusiński J. (2005c): *Voltage power line conditioner – voltage restoring mode*. — Proc. 11th European Conf. *Power Electronics and Applications*, Dresden, Germany, pp. 1167–1173.

Thomsen P. (1999): *Application and control of CUPS in the distribution grid*. — Institute of Energy Technology, Aalborg University, Vol. 3, pp. 2–11.