Appl. Math. and Comp. Sci., 1996, vol.6, No.1, 159–168

CREST FACTOR MINIMIZATION: A NEURAL NETWORK APPROACH

NIKOS E. MASTORAKIS*

In this paper, a state-space model is proposed for the minimization problem of the crest factor in the case of a multisine signal. This state-space model provides a stable neural network and it can be studied using computer simulation.

1. Introduction

The crest factor (CF) of a signal is defined as the ratio of its peak value and its root mean-squared (RMS) value. The number of averages required to measure a signal with specified accuracy is proportional to the square of the crest factor (Shoukens *et al.*, 1988). This shows the importance of CF minimization in measurement problems. Up to now, the exact solution for the CF minimization problem has not been given yet. A survey of the existing techniques can be found in (Van der Ouderaa *et al.*, 1988b). Van den Bos (1985), Kahane (1980) and Overton (1982) presented some very interesting results. In (Van der Ouderaa *et al.*, 1988b), one can find a summary of the random method, the Rudin-Shapiro polynomial method (Rudin, 1959; Shapiro, 1951), the Schroeder method (Schroeder, 1970), the Newman and Littlewood methods (Littlewood, 1966; Newman, 1965), and the Van den Bos and Krol method (Van den Bos, 1987; Van den Bos and Krol, 1979). Another algorithm is that of Gerchberg and Saxton (1972). An extension of this algorithm in the time domain of band-limited Fourier signals was given by Van der Ouderaa *et al.* (1988a). Boyd (1986) uses some results from the mathematical literature in order to generate signals with very low crest factors. In (Van der Ouderaa and Renneboog, 1988), the Schroeder and random methods are compared with a time-frequency-domain swapping algorithm for the crest factor reduction of logtones used in 1/3 octave analysis. In (Guillaume *et al.*, 1991), the minimization of the crest factor is attempted using the L_p error criterion and the Levenberg-Marquardt routine. The method of the present paper is based on that of (Guillaume *et al.*, 1991), but instead of the Levenberg-Marquardt routine a neural-network (Zhang and Constantinides, 1992) is proposed for the minimization of the CF.

Before presenting the state-space model for this neural network, the following definition is given.

* Hellenic Naval Academy, Department of Engineering, Terma Hatzikyriakou, 18539, Piraeus, Greece

Definition 1. The crest factor CF_x of the function $x(t)$ is defined as

$$CF_x = \frac{\max\limits_{t\in[0,T]} |x(t)|}{\sqrt{\dfrac{1}{T}\displaystyle\int_0^T |x(t)|^2\,dt}} \tag{1}$$

This number best represents the amount by which a signal is compressed. Apart from that, the number of averages required to measure a signal with a specified accuracy is proportional to the square of the crest factor (Shoukens *et al.*, 1988).

It is well-known that the ℓ_p-norm of a function $x(t)$ defined on the interval $[0, T]$ is given by

$$\ell_p(x) = \left(\frac{1}{T}\int_0^T |x(t)|^p\,dt\right)^{1/p}, \qquad p \geq 1 \tag{2}$$

For continuous functions, the uniform norm is also introduced

$$\ell_\infty(x) = \max_{t\in[0,T]} |x(t)|$$

In the notation above, one can write

$$CF_x = \frac{\ell_\infty(x)}{\ell_2(x)} \tag{3}$$

2. CF Minimization

A multisine signal is defined as

$$x(t) = \sum_{u=1}^{N_u} \alpha_u \cos\left(2\pi k_u \frac{t}{T} + \phi_u\right) \tag{4}$$

where k_u are the harmonic numbers, $k_u \in \mathbb{N}$, and $0 < k_1 < k_2 < \ldots < k_{N_u} = k$. The ℓ_2-norm (i.e. the RMS value) is dependent only on the amplitudes α_u and thus, in the special case of a given α_u, the CF minimization is actually the minimization of $\ell_\infty(x)$. Following the approach presented in (Guillaume *et al.*, 1991), in order to minimize $\ell_\infty(x)$ with respect to ϕ_u, we minimize the corresponding 'discrete' norm $L_p(x)$ with respect to ϕ_u and then we let $p \to \infty$. We recall that

$$L_p(x) = \left(\frac{1}{N}\sum_{n=0}^{N-1} |x_n|^p\right)^{1/p}, \qquad L_\infty(x) = \max_{n=0,\ldots,N-1} |x_n| \tag{5}$$

where $x_n = x\left(\frac{nT}{N}\right)$.

In (Guillaume *et al.*, 1991), it is proved that $\ell_p(x) = L_p(x)$ if $p = 2r$ $(r \in \mathbb{N})$, $N > pk + 1$, and the interval $[0, T]$ is divided into N equal subintervals. If these conditions are not satisfied, the discrete norm is only an approximation of the continuous one.

It is obvious that the L_p-minimization problem reduces to the minimization of $\Lambda_p = L_p^p$, i.e.

$$\Lambda_p = \frac{1}{N} \sum_{n=0}^{N-1} |x_n|^p \tag{6}$$

We minimize Λ_p with respect to ϕ_u with p even and for an increasing sequence of p's with $p \to \infty$. Therefore, we have to solve the problem of minimizing $\Lambda_r = \frac{1}{N} \sum_{n=0}^{N-1} x_n^{2r}$.

Here, a neural network is proposed for the above non-linear programming problem. The transient behaviour of the proposed neural network is defined by the following vector equation:

$$\frac{\mathrm{d}\phi}{dt} = -\nabla_\phi \Lambda_r \tag{7}$$

where $\phi = [\phi_1, \ldots, \phi_{N_u - 1}]$, since it is supposed, as a result of normalization, that $\phi_{N_u} = 0$.

The state-space equations for this network are

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = -\frac{\partial \Lambda_r}{\partial \phi_i}, \qquad i = 1, \ldots, N_{u-1} \tag{8}$$

The stability of this network is proved as follows. The function Λ_r is selected as a Lyapunov function of the system. It is differentiable, positive-definite, and $\mathrm{d}\Lambda_r/\mathrm{d}t = -(\nabla\Lambda_r)^2 \leq 0$. Therefore the proposed network is stable. In the equilibrium point, we have

$$\frac{\partial \Lambda_r}{\partial \phi_i} = 0, \qquad i = 1, \ldots, N_{u-1} \tag{9}$$

which constitutes the first-order conditions of the minimization problem under consideration. Therefore the minimum of Λ_r is easily obtained by this network equilibrium point.

An improved version of eqns. (8) is

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = -\mu_i \frac{\partial \Lambda_i}{\partial \phi_i} = 0, \qquad i = 1, \ldots, N_{u-1}$$

where $\mu_i = 1/\tau_i > 0$ are referred to as learning rates of the neural network, and τ_i are proper time constants. In general, each differential equation has its own learning rate. By their suitable choice we can provide an appropriate scaling and therefore reduce the stiffness of the differential equations and improve the convergence rate of the system.

For well-scaled problems all learning rates could be identical ($\mu_i = \mu_0 = 1/\tau_0$ for all i). By the chain rule, one can find

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = -\mu_i \frac{1}{N} \sum_{n=0}^{N-1} 2r\, x_n^{2r-1} \frac{\partial x_n}{\partial \phi_i}, \qquad i = 1, \ldots, N_{u-1}$$

Note that

$$\frac{\partial x_n}{\partial \phi_i} = -\sin\left(2\pi k_i \frac{n}{N} + \phi_i\right) \alpha_i \tag{10}$$

Thus, we have the following system of $N_u - 1$ differential equations

$$\frac{\mathrm{d}\phi_i}{\mathrm{d}t} = -\mu_i \frac{1}{N} \sum_{n=0}^{N-1} 2r \left[\sum_{u=1}^{N_u} \alpha_u \cos\left(2\pi k_u \frac{n}{N} + \phi_u\right)\right]^{2r-1} \alpha_i \left[-\sin\left(2\pi k_i \frac{n}{N} + \phi_i\right)\right],$$
$$i = 1, \ldots, N_u - 1 \tag{11}$$

The system of differential eqns. (11) is suitable for hardware implementation (Agranat *et al.*, 1990; Kennedy and Chua, 1988; Rodriguez-Vazquez *et al.*, 1990; Yanai and Sawada, 1990; Zhang and Constantinides, 1992) as well as for computer simulation. This system was simulated on a computer (IBM compatible, 486 DX2, 66MHZ) using the fifth-order Runge-Kutta-Fehlberg method. The results are the same as those of (Guillaume *et al.*, 1991), see Fig. 1.

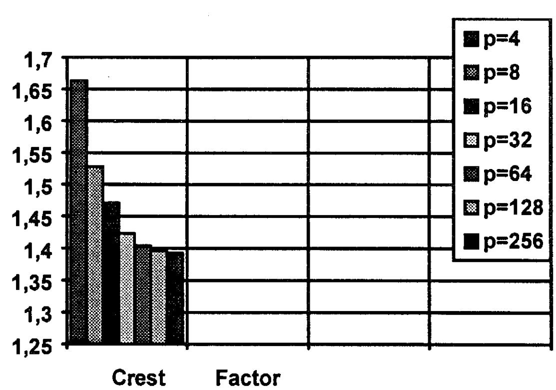

Fig. 1. Crest factor minimization of a multisine signal with $N_u = 31$, $\alpha_u = 1$, $k_u = u$, $u = 1, 2, \ldots, 31$.

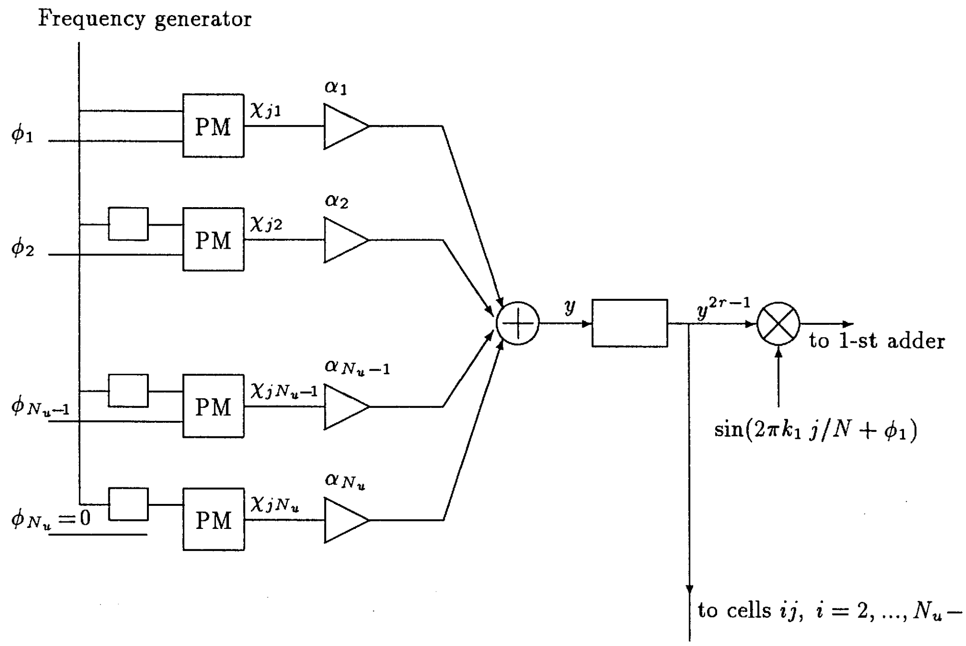

Fig. 2. Implementation of 'cell $1j$', $j = 0, 1, 2, \ldots, N-1$.

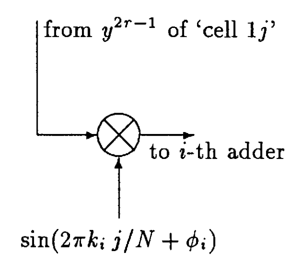

Fig. 3. Implementation of 'cell ij', $i = 2, \ldots, N_u - 1$, $j = 0, 1, 2, \ldots, N-1$.

3. Hardware Implementation

In this paragraph, a hardware implementation of the proposed neural network is presented. The neural network consists of elementary cells. In Figs. 2 and 3, the quantities 'cell ij' are shown, where $i = 1, 2, \ldots, N_u - 1$ and $j = 0, 1, \ldots, N-1$. The input for the elementary 'cell $1j$' consists of the variables $\phi_1, \ldots, \phi_{N_u-1}$, which, for brevity, are denoted by ϕ (note that $\phi_{N_u} = 0$). Each cell $1j$ is composed of N_u

phase modulators (PM). To the input of each PM, we connect a frequency shifting circuit which, in Fig. 2, is denoted by a small box before the box of the PM. So, in the 'cell $1j$', the output of each PM is $\chi_{ju}(t) = \cos\left(2\pi k_u j/N + \phi_u(t)\right)$, where $u = 1, 2, \ldots, N_u$. The box after the adder gives the $(2r-1)$-th power of the input signal. In the sequel, multiplication by another $\sin(\cdot)$ is required and the signal is fed to the first (general) adder. The triangles denote various amplifiers. A simplified structure of the 'cells ij' ($i > 1$) is given in Fig. 3.

The complete implementation is shown in Fig. 4, where one should notice that finally the necessary feedback to the initial variables $\phi_1, \ldots, \phi_{N_u-1}$ is created. Each integrator is supplied with an appropriate initial condition which is actually the initial value of the corresponding variable ϕ_i, $i = 1, 2, \ldots, N_u - 1$.

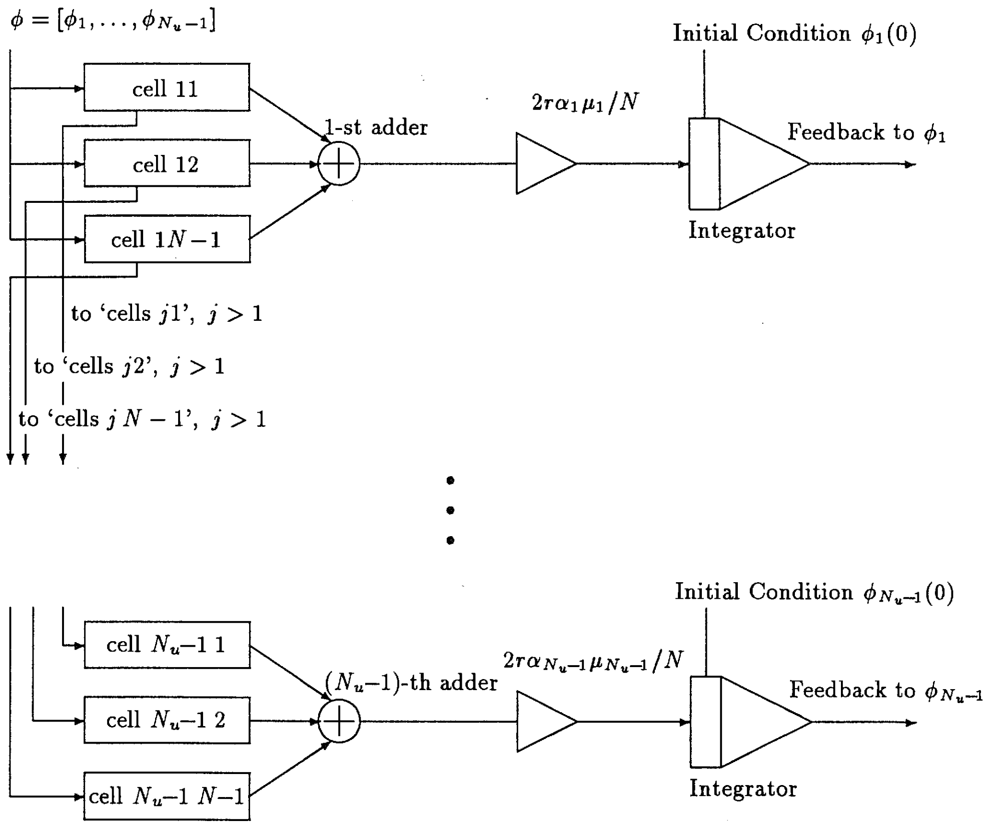

Fig. 4. The proposed network.

Remark. Many research papers have been devoted to optimization via neural networks techniques recently. One can see (Agranat *et al.*, 1990; Bouzerdoum and Pattison, 1993; Kennedy and Chua, 1988; Lillo *et al.*, 1993; Rodriguez-Vazquez *et al.*, 1990; Yanai and Sawada, 1990; Zhang and Constantinides, 1992).

In particular, for the CF minimization, the advantage of the hardware implementation is that for many cases we have to solve the same problem for different α_u or k_u. Thus, in this case, it is advantageous to have a permanent circuit structure for evaluation of optimal $\phi_1, \ldots, \phi_{N_u-1}$. On the other hand, the neural network, compared with other discrete methods, offers a rapid computation of $\phi_1, \ldots, \phi_{N_u-1}$. The reason is that, in this case, the variation in $\phi_1, \ldots, \phi_{N_u-1}$ always follows directly the direction of the vector $-\nabla_\phi \Lambda_r$. The same is not true, in general, for the discrete methods (see e.g. the method of steepest descent in any standard textbook on optimization techniques, (Luenberger, 1972; Murray, 1972)). A geometrical interpretation of the above ideas in a special case where $N_u = 3$ is shown in Figs. 5 and 6.

4. Conclusion

A stable neural network is proposed for the minimization problem of the crest factor of a multisine signal. Computer simulations are used and the results are compared with those of (Guillaume *et al.*, 1991). It should be noticed that the laboratory implementation offers a real, efficient, and fast computation of the optimal $\phi_1, \ldots, \phi_{N_u-1}$. Thereafter, computation of the crest factor is reduced to eqns. (3)–(6). Since the problem always includes a multisine signal with different α_u and k_u, the hardware implementation of a neural network is of great importance. Some other relevant studies can be found in (Agranat *et al.*, 1990; Bouzerdoum and Pattison, 1993; Kennedy and Chua, 1988; Lillo *et al.*, 1993; Rodriguez-Vazquez *et al.*, 1990; Yanai and Sawada, 1990; Zhang and Constantinides, 1992).

Acknowledgement

The author wishes to thank the anonymous reviewer for his comments, which improved the quality of the paper.

References

Agranat A.J., Neugebauer C.F., Nelson R.D. and Yarir A. (1990): *The CCD neural processor: A neural network integrated circuit with 65536 programmable analog synapses.* — IEEE Trans. Circ. Syst., v.37, No.8, pp.1073–1075.

Bouzerdoum A. and Pattison T.R. (1993): *Neural network for quadratic optimization with bound constraints.* — IEEE Trans. Neural Networks, v.4, No.2, pp.293–304.

Boyd S. (1986): *Multitone signals with low crest factors.* — IEEE Trans. Circuits and Systems, v.CAS-33, No.10, pp.1018–1022.

Gerchberg R.W. and Saxton W.O. (1972): *A practical algorithm for the determination of phase from image and diffraction plane pictures.* — Optik, v.35, No.2, pp.237–247.

Guillaume P., Schoukens J., Pintelon R. and Kollar I. (1991): *Crest factor minimization using nonlinear Chebyshev approximation methods.* — IEEE Trans. Instrum. and Measur., v.40, No.6, pp.982–989.

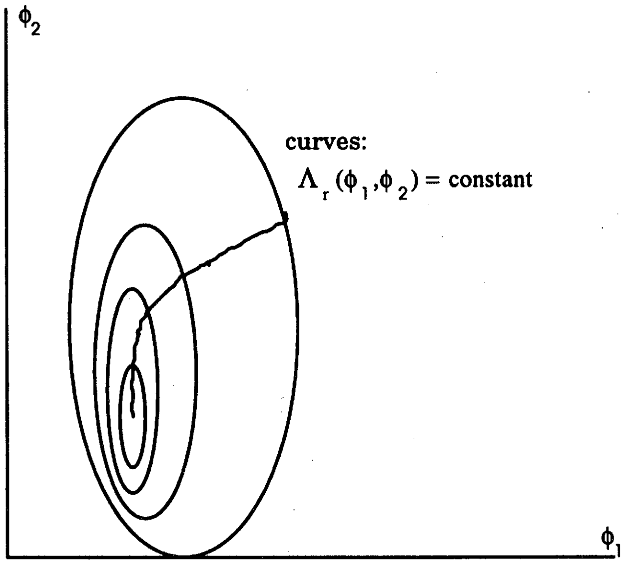

Fig. 5. The neural network leads the state variables directly to a (local) minimum.

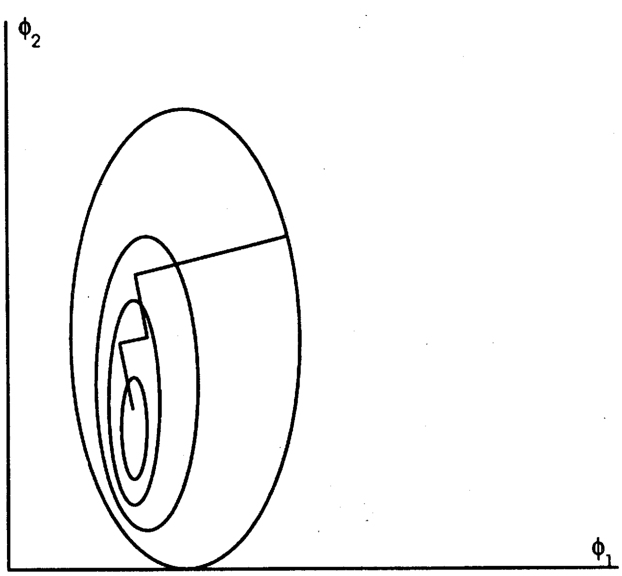

Fig. 6. The classical method of steepest descent (Luenberger, 1972).

Kahane J.P. (1980): *Sur les polynomes a coefficients unimodulaires.* — Bull, London Math. Soc., No.12, pp.321–342.

Kennedy M.P. and Chua L.O. (1988): *Neural networks for non-linear programming.* — IEEE Trans. Circ. Syst., v.35, No.5, pp.554–562.

Lillo W.E., Loh M.H., Hui S. and Zak S.H. (1993): *On solving constrained optimization problems with neural networks: A penalty method approach.* — IEEE Trans. Neural Networks, v.4, No.6, pp.931–940.

Littlewood J.E. (1966): *On polynomials* $\sum^n \pm z^m$, $\sum^n e^{alpha} m^j z^m$, $z = e^{teta}$. — J. London Math. Soc., v.41, pp.367–376.

Luenberger D.G. (1972): *Introduction to Linear and Nonlinear Programming.* — Washington D.C.: Addison-Wesley Publishing Company.

Murray W. (Ed.) (1972): *Numerical Methods for Unconstrained Optimization.* — London and New York: Academic Press.

Newman D.J. (1965): *An L1 extremal problem for polynomials.* — Pr. Amer. Marh. Soc., v.16, pp.1287–1290.

Overton M.L. (1982): *Algorithms for nonlinear* l_1 *and* l_∞ *fitting.* In: Nonlinear Optimization, (M.J.D. Powell, Ed.), New York: Academic Press.

Rodriguez-Vazquez A., Dominquez-Castro R., Rueda A., Huertas J.L. and Sanchez-Sinencio E. (1990): *Non-linear switched-capacitor neural networks for optimization problems.* — IEEE Trans. Circuits Syst., v.37, No.3, pp.384–398.

Rudin W. (1959): *Some theorems on Fourier coefficients.* — Proc. Am. Math. Soc., No.10, pp.855–859.

Schroeder M.R. (1970): *Synthesis of low-peak-factor signals and binary sequences with low autocorrelation.* — IEEE Trans. Information Theory, v.IT-16, No.1, pp.85–89.

Shapiro H.S. (1951): *Extremal problems for polynomials.* — M.Sc. Thesis M.I.T.

Shoukens J., Pintelon R., Van der Ouderaa E. and Renneboog J. (1988): *Survey of excitation signals for FFT based signal analyzers.* — IEEE Trans. Instrum. and Meas., v.IM-37, No.9, pp.342–352.

Van den Bos A. (1985): *Nonlinear least-absolute-values and minimax model fitting.* — Proc. 7th IFAC Symp. *Identification and System Parameter Estimation*, York, UK, pp.173–177.

Van den Bos A. (1987): *A new method for synthesis of low-peak- factor signals.* — IEEE Trans. Acoust., Speech, Signal Processing, v.ASSP-35, No.1, pp.120–122.

Van den Bos A. and Krol R.G. (1979): *Synthesis of discrete-interval binary signals with specified Fourier amplitude spectra.* — Int. J. Control, v.30, No.5, pp.871–884.

Van der Ouderaa E. and Renneboog J. (1988): *Logtone Crest Factors.* — IEEE Trans. Instrum. and Measur., v.37, No.4, pp.656–657.

Van der Ouderaa E., Schoukens J. and Renneboog J. (1988a): *Peak factor minimization using a time-frequency domain swapping algorithm.* — IEEE Trans. Instrum. and Measur., v.37, No.1, pp.145–147.

Van der Ouderaa E., Schoukens J. and Renneboog J. (1988b): *Peak factor minimization of input and output signals of linear systems.* — IEEE Trans. Instrum. and Meas., v.37, No.2, pp.207–212.

Yanai H. and Sawada Y. (1990): *Intergrator neurons for analog neural networks.* — IEEE Trans. Circ. Syst., v.37., No.6, pp.854–856.

Zhang S. and Constantinides A.G. (1992): *Lagrange programming neural network.* — IEEE Trans. Circ. and Syst., v.39, No.7, pp.441–452.

Received: June 7, 1995
Revised: November 17, 1995