Int. J. Appl. Math. Comput. Sci., 2021, Vol. 31, No. 4, 601–611
DOI: 10.34768/amcs-2021-0041

AN EFFECTIVE DATA REDUCTION MODEL FOR MACHINE EMERGENCY STATE DETECTION FROM BIG DATA TREE TOPOLOGY STRUCTURES

IAROSLAV IAREMKO [a], ROMAN SENKERIK [a,*], ROMAN JASEK [a], PETR LUKASTIK [b]

[a]Department of Informatics and Artificial Intelligence
Tomas Bata University in Zlín
nám. T.G. Masaryka 5555, 760 01 Zlín, Czech Republic
e-mail: senkerik@utb.cz

[b]Tajmac-ZPS
třída 3. května 1180, 763 02 Zlín, Malenovice, Czech Republic
e-mail: plukasik@tajmac-zps.cz

This work presents an original model for detecting machine tool anomalies and emergency states through operation data processing. The paper is focused on an elastic hierarchical system for effective data reduction and classification, which encompasses several modules. Firstly, principal component analysis (PCA) is used to perform data reduction of many input signals from big data tree topology structures into two signals representing all of them. Then the technique for segmentation of operating machine data based on dynamic time distortion and hierarchical clustering is used to calculate signal accident characteristics using classifiers such as the maximum level change, a signal trend, the variance of residuals, and others. Data segmentation and analysis techniques enable effective and robust detection of operating machine tool anomalies and emergency states due to almost real-time data collection from strategically placed sensors and results collected from previous production cycles. The emergency state detection model described in this paper could be beneficial for improving the production process, increasing production efficiency by detecting and minimizing machine tool error conditions, as well as improving product quality and overall equipment productivity. The proposed model was tested on H-630 and H-50 machine tools in a real production environment of the Tajmac-ZPS company.

Keywords: OPC UA, OPC tree, PCA, big data analysis, data reduction, machine tool, anomaly detection, emergency states.

1. Introduction

Modern production processes are based on the digitization of machine tools with numerical control (CNC) and programmable logic controllers (PLC) throughout obtaining various data from the machines, their storage, evaluation, display, and analytics performed using big data sets obtained from a wide range of sources using dedicated computer systems. This work presents a method to reduce and analyze data from a large n-dimensional environment into two data components with minimal loss and quality of the original information.

The conceptual structure of collected data from the machine tools given by many records and attributes has triggered the development of several big data platforms as a parallel data analytics algorithm. An essential aspect is the usage of data dimensionality reduction procedures.

The analysis of a large tree hierarchical node and its components is not a trivial matter. An important task investigated in this work is making the monitored data node a single data unit. In such a case, the data tree that looks like a matrix with many rows and columns (the matrix can contain up to tens and hundreds of data columns) is transformed as a wiped data matrix into a single vector without losing the original information. The first milestone of such a project is then to reduce the number of columns in the data set and to lose the smallest amount of information possible at the same time. The essential step is to choose the proper method for data reduction. The most popular techniques and algorithms and their results are listed below (see Section 2).

Research on big data analytics is entering the new phase called fast data, where multiple gigabytes

*Corresponding author

of data are collected in the big data systems every second (ur Rehman *et al*., 2016). Modern big data systems collect complex data streams. The complexity of the systems is given by the volume, velocity, value, variety, variability, and veracity in the acquired data, thus defining the 6Vs of big data. As stated by ur Rehman *et al*. (2016), the reduced and relevant data streams are perceived to be more beneficial than collecting raw, redundant, inconsistent, and noisy data. Another critical perspective for big data reduction is that the vast number of variables and features in big data sets cause the famous phenomenon called the "curse of dimensionality," resulting in requirements for unbounded computational resources to analyze the data.

Figure 1 shows that the original data node of the machine tool contains many different signals. The number of data signals can vary from a few tens to hundreds or even more. In this case, it is not very convenient to process each signal separately to calculate some deviation from the normal state. Another disadvantage is the required agreement in advance on what signals will be processed and in which order. Therefore, this work aims to develop a model that will be able to process at once a large number of data signals obtained using a data collector design, similarly as described by Pathria and Beale (2011). These signals must be collected in a data node and stored in a data repository as in the work of Zhang *et al.* (2019b). The developed model then must be able to analyze these data using mathematical calculations, obtain a deviation from the normal, and the output must be a signal spectrogram. Figure 1 also shows that the 3D data model contains a large amount of data exceeding some maximum and minimum limits, which by mistake can be understood as the detection of an emergency state (or anomaly); nevertheless, as shown in case studies, they do not detect an emergency condition or a deviation from the normal on the machine tool at all.

Thus principal component analysis (PCA) (Abdi and Williams, 2010) as a multi-dimensional reduction method is used here to detect an emergency (anomalous) state in a data node that has a hierarchical graph structure, and it is analyzed as a whole. Therefore, it would be possible to determine what part of the machine has failed precisely (i.e., a place on the spindle or where the machine tool dropped the ball screw, etc.). We assume that the data analysis can also reveal the level of criticality and whether it is required to perform an immediate replacement of the part. For such a type of decision making, the use of vibration selection methods, acceleration, and vibration shifts will be required, and it will be discussed in detail in further scientific research.

Data reduction is also is very useful when PCA is combined in the model with classifiers to determine the emergency state and the correct calculation of signal peaks, with high accuracy on the machine data node.

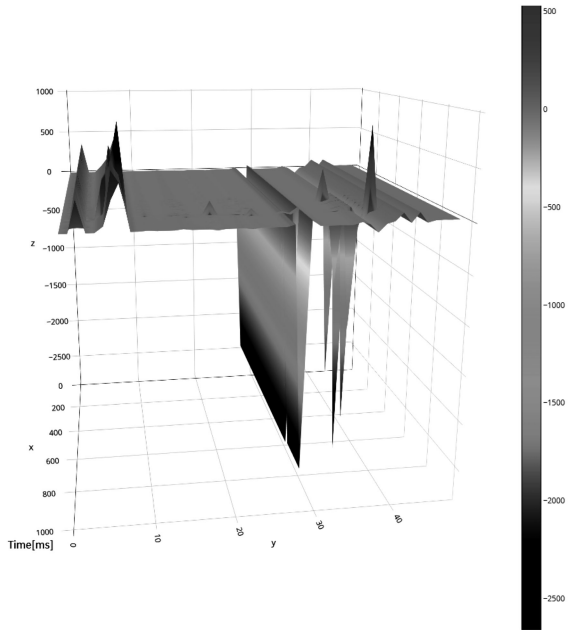

Fig. 1. Condition monitoring node.

For further comparing and confirming that there was a real failure in the data node, it is possible to combine the obtained data analysis with the graph of temperature development measured on the nodes of the machine tool. When the temperature graph shows a sharp temperature jump, assuming that the PCA analysis shows an anomalous state, there was most likely a failure at a particular measurement data node. As far as machine tools are concerned, the PCA reduction-based method must not include quantities or data nodes such as temperature and vibrations. The main goal of the presented research is to create an effective reduction and classification model based on ideas presented by Żabiński *et al*. (2019), Cooper *et al.* (2020a; 2020b) or Zhang *et al.* (2019a) that enables us to process a big data tree topology matrix, and to detect a data anomaly and an emergency state.

1.1. Motivation and originality. The motivation, originality, and contribution of this research can be summarized as follows

- The objective consists in creating a robust efficient system using a parallel modular structure of any number and types of classifiers for faster detection of anomalies and emergency states from big data tree structures.
- Thus originality is given by replacing the DBSCAN algorithm with an elastic modular system of fast classifiers.
- The results also show sufficient accuracy of the proposed model for the use of only one PCA

component, enabling more efficient data processing and prediction.

2. Related work

As stated in the previous section, and considering the clearly defined originality of this paper, the presented research is partly based on that by Hankerson *et al*. (2000), Żabiński *et al*. (2019), Pathria and Beale (2011) or Cooper *et al.* (2020b). Hankerson *et al*. (2000) present developed models and algorithms, which use a similar methodology and a basic concepts like our research. The mentioned paper represents the direction and a basis for creating component analysis for machine tools developed in our work. In the paper by Żabiński *et al*. (2019), a methodology is designed for the evaluation and detection of an emergency condition on a machine tool. The basis of the method is the data reduction of the data node signal. PCA has been used to construct a neural network prediction model (LSTM type) for detecting a future anomaly in the system. The task was to create mathematical models and classifiers for calculating the emergency characteristics of the data signal. In work of Zhang *et al.* (2019b) classifiers have been used to calculate the anomaly in the reduced signal of the rim input data node.

The research presented by Żabiński *et al*. (2019) introduces an algorithm for data anomaly detection based on PCA components and further processing by the DBSCAN algorithm (Ester *et al*., 1996).

In general, The PCA processes a multidimensional matrix and distills it into its main components by capturing the directions of various deviations. As given by Zhang *et al.* (2019b) and Żabiński *et al*. (2019), this means that 70% of the variance of a data set can be captured in one dimension and 95% can be captured in two dimensions; removing all other variables will result in a loss of only 5% of the information compared with the original data. The result is a combination of variables that can best represent data, and from these combined variables implicit information can often be derived that is the basis of the data set.

Within our research, a multidimensional matrix is obtained from a machine tool by collecting all positions, displacements, speeds, and loads (53 variables for a given machine, may be even more or less). All these variables are reduced to two main components, capturing important information from 53 original variables. The data method must identify and decode the relationships because the values in the main signals (components) and must be sensitive to cases where unusual relationships arise between the original 53 signals.

After finding the latent signals, they can be plotted, and it is possible to find that these are relatively consistent signals. With such a cleaned signal, it is possible to start detecting anomalies on a specific time period in the machine tool and combine the obtained PCA intermediate results into a global PCA. Representation of data by several principal components significantly limits data transmissions with minimal degradation of accuracy. It is assumed that most high dimensional data has a lower internal dimensionality allowing a good representation of the lower dimension. The numerous experiments performed revealed that the additional implementation of DBSCAN (Żabiński *et al*., 2019) does not affect the final results, meaning that the results are almost identical, and less time is required for the algorithm to run.

Feature selection is an important factor in the success of the data mining process through selecting the useful or relevant attributes in the data set. Bartenhagen *et al*. (2010) present a comparison between feature selection methods and their impact on learning algorithms. The authors claimed that most methods of selecting the features improved the accuracy and the performance of the classification techniques. Popular feature selection and reduction models, similar to those that are currently used in our research for processing and reduction of large amounts of data, include factor analysis, random forests, the missing values ratio, low variance filter, high correlation filter, backward feature elimination, forward feature construction, latent Dirichlet allocation (LDA), canonical analysis (CCA) or their types KCCA, mCCA, or using (deep) convolutional neural networks (DCNNs/CNNs) and many others (Bansal and Bansal, 2016; ur Rehman *et al*., 2016; Hyndman *et al*., 2015; Schleinkofer *et al*., 2019; Filter and Filter, 2014). The work of ur Rehman *et al*. (2016) presents a review of methods that are used for big data reduction. It also presents a detailed taxonomic discussion of big data reduction methods including network theory, big data compression, dimension reduction, redundancy elimination, data mining, and machine learning methods. The open research issues pertinent to the big data reduction are also highlighted there.

It is the subject of further research and comparison of these methods with each other according to time performance and algorithmic complexity.

3. Reduction and detection model algorithm

This section contains a comprehensive explanation of the developed algorithm (a reduction and detection model), as well as experiments with different parallel frameworks and a comparison of sequential vs. parallel execution of an emergency state classifier ensemble (see Section 3.7) . The whole reduction model can be simplified into the following six steps taking into account the information given in the previous section and a workflow diagram presented in Fig. 2:

1. Get an input big data matrix from input tree

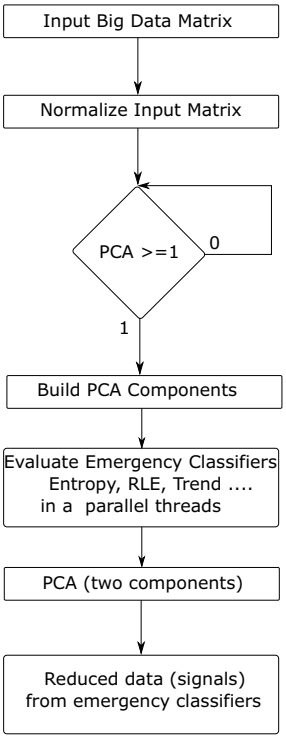

Fig. 2. Research workflow.

hierarchical data.

2. Normalize the input data matrix.
3. Calculate the PCA components.
4. Form the emergency classifier in parallel threads.
5. Calculate resulting PCA components from emergency (crash) classifiers signals.
6. Get the final reduced data matrix.

The complete process (workflow diagram) is also depicted in Fig. 2: each step is described in detail in Sections 3.1 to 3.6. The core algorithm for building the final PCA components and the parallel processing of emergency classifiers is also given as a pseudo-code (see Algorithm 1).

3.1. Input data reading. A big data $m \times n$ matrix of is obtained from input tree topology data, stored in a database as a JSON object. This data does not contain values such as temperature and vibration according to recommendations of Zhang *et al.* (2019a). The next steps represent the z-normalization.

3.2. Data normalization. Z-data normalization (Goldin and Kanellakis, 1995) is necessary as a preprocessing method before further analysis can be performed. Calculations in each column of matrices were performed using the statistical method of the z-score:

$$z = \frac{x - \mu}{\sigma}, \tag{1}$$

where x denotes processed data, μ stands for the mean value of the column in the data file and σ is the standard deviation of the column, that were normalized. If σ of a given column did not equal zero, each individual data point in the column was processed according to (1).

3.3. PCA reduction. Within this workflow step, the PCA-based input data matrix reduction provides a transformation from the $m \times n$ matrix size to a single vector (size n) with all information saved.

PCA is a statistical method suitable for big data reduction and data processing. The data processing is performed by data set removal, observation and conversion to a set of linearly uncorrelated principal components for further analysis. PCA is performed here by using singular value decomposition (SVD) and obtaining the matrix $\mathbf{M}$. The SVD technique has been selected due to the high effectiveness in common parallel frameworks. The matrix $\mathbf{M} \in \mathbb{R}^{m \times n}$ is defined as

$$\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T, \tag{2}$$

where $\mathbf{U} \in \mathbb{R}^{n \times n}$ is a unitary matrix, $\mathbf{\Sigma} \in \mathbb{R}^{m \times n}$, and $\mathbf{V} \in \mathbb{R}^{m \times n}$ is also a unitary matrix. Diagonal items σ_i of $\mathbf{\Sigma}$ are known as singular values of the matrix $\mathbf{M}$, and are similar, but not always the same as eigenvalues. Singular values are equivalent only to eigenvalues when the given matrix is a real, symmetric, square, positive definite matrix.

As already mentioned, PCA can be used to reduce the dimensions and project matrices onto a lower-dimensional subspace, which would be more convenient in finding anomalies in machine tools. The disadvantage of PCA is that each principal component is a linear combination of all the original observations, making it difficult to interpret the results and be prone to outliers.

For the subset of data that has been obtained from the machine tool during the time interval of approximately. 3 hours and analyzed after preliminary processing, the resulting matrix of data signals is $\mathbf{M} \in \mathbb{R}^{10038 \times 52}$.

The next step is to construct the SVD matrix from this not-reduced data (Pathria and Beale, 2011; Zhang *et al.*, 2019b). Very often the situation of a lost weight of values σ_i can be observed. This means that in the matrix $\mathbf{\Sigma}$, which is obtained from the input data hierarchical tree structure, the values of this matrix can lost their weights that are largest singular values and in this case it will be more complicated to predict other singular values for other matrix data. Values σ_i can also be displayed in descending order, and further analysis of the obtained data was then

performed using the first two singular values of the matrix $\mathbf{\Sigma}$ as the x and y axes. The first two singular values were chosen above the others because the results of the matrix $\mathbf{\Sigma}$ are returned in descending order, so that the first two singular values represent two values that change the scale of unitary SVD matrices $\mathbf{U}$ and $\mathbf{V}^T$ to recreate the original matrix. The first two singular values were multiplied by the unitary matrix $\mathbf{U}$ and returned data with a slightly lower number of input data matrix rows with 2 columns. This was also done for unitary matrix $\mathbf{V}^T$, and the results were transposed into the same form as in the previous case (Bansal and Bansal, 2016).

The columns of two matrices $\mathbf{U}$ and $\mathbf{V}^T$ were divided into fields, the values were sent to lists, and the lists were then combined to create data points that will be rendered.

The singular values, according to the graph $\mathbf{U}$, projected all the data in a concatenated matrix, which shows the linear relationship between the data, and individual data points falling above and below the general trend line.

PCA works in conjunction with the SVD method on different unit and diagonal matrices, which gives the optimization problem as follows:

$$\min(\mathbf{L}, \mathbf{S})||\mathbf{L}||_* + \Lambda||S||_1, \tag{3}$$

which is transformed into the form:

$$\mathbf{M} - (\mathbf{L} + \mathbf{S}) \geq \epsilon. \tag{4}$$

In (3), there is a lower order matrix $\mathbf{L}$, which can be factorized as an SVD matrix, $||\mathbf{L}||_*$ is a transformed matrix from matrix $\mathbf{L}$ as the sum of singular values. Λ is the coupling constant between $\mathbf{L}$ and $\mathbf{S}$, $||S||_1$ is the sum of the items of $\mathbf{L}$ and ϵ, where ϵ is a matrix of point error constants that improve noise generated from real data.

3.4. Emergency classifiers. After the data reduction, it is necessary to evaluate emergency signals which include information detecting emergency states from the obtained singular vector (PCA signal vector decomposition). Due to the modular and elastic architecture, any number of PCA components is fed to the input of the emergency classifier calculations.

For efficient execution of the model, it is necessary to use emergency classifiers to determine the detection or anomalous situation on the time series (i.e., on the data node). Emergency (anomaly) classifiers are represented by mathematical statistical quantities obtained with the help of different computational models, and which exclude from the data signal unnecessary values and leave only these, which can be used to construct a signal that has in its system processes or values that are responsible for the state data signal).

Table 1. Execution time of eigenvectors and eigenvalues calculation for different data sizes and the H-630 machine tool.

Time interval [min]	ApacheCommon [ms]	ApacheSpark [ms]
45	72	78
60	61	66
80	78	89
100	90	94
120	78	83
140	83	74
160	91	96

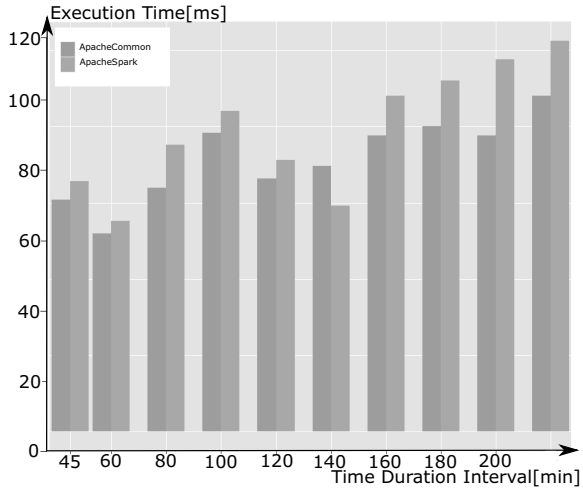

Fig. 3. Comparison of frameworks with respect to time performance.

Regarding these classifiers, further research will be devoted to developing and elaborating classifier models for easier detection of an emergency. Below are just some of the most used classifiers for the emergency state of a data signal.

These signals are entropy (Żabiński *et al.*, 2019; Siboni and Cohen, 2020; Yao *et al.*, 2019), the maximum number of consecutive observations, the maximum level change, the maximum change of rolling means, autocorrelation (ACF1) (Kamat and Sugandhi, 2020; Mühlbauer *et al.*, 2020; Basora *et al.*, 2019), the variance of residuals or a signal trend. As given in Algorithm 1, a reduced data matrix containing n rows and two columns is extended to six dimensions, i.e., from a matrix with two columns and n rows, we obtain a matrix with six columns and n rows. Each column of the reduced data matrix is processed in parallel for a given number of emergency classifiers (six in this paper). For each emergency classifier, both the PCA1 data vector and the PCA2 data vector are evaluated with the classifier, and then we evaluate the simple moving average (SMA) for

Algorithm 1. Parallel evaluation of emergency classifiers and secondary PCA reduction.

INPUT: $\mathbf{PCAComponents} \neq \emptyset$

DECLARE:
PCLength: number of PCA Components (number of columns of input data matrix $\mathbf{PCAComponents}$)
n: Length of data of single PCA Component (number of rows of input data matrix $\mathbf{PCAComponents}$)
k: number of classifiers used
$\mathbf{ClassifiersVectorsDimExt}$: $n \times k \times$ *PCLength* matrix
$\mathbf{ClassifiersVectors}$: $n \times k$ matrix
$\mathbf{FinalComponents}$: $n \times$ *PCLength* matrix

ALGORITHM:
for $i \leftarrow 1$ TO *PCLength* **do**
  $\mathbf{ClassifiersVectorsDimExt}$(All, k) $\leftarrow$ parallel processing of k emergency classifiers for input data $\mathbf{PCAComponents}$(All, i) ; $k = 1, \ldots, k$
end for
for $i \leftarrow 1$ TO k **do**
  $\mathbf{ClassifiersVectors}$(All, k) $\leftarrow$ parallel processing of SMA($\mathbf{ClassifiersVectorsDimExt}$(All, i); $\mathbf{ClassifiersVectorsDimExt}$(All, $i + k$))
end for
$\mathbf{FinalComponents}$ $\leftarrow$ PCAConstructModuleFunction($\mathbf{ClassifiersVectors}$)

OUTPUT: $\mathbf{FinalComponents}$

the corresponding classifier, thus reducing the output to a 1-dimensional vector. This process is repeated for all selected classifiers, resulting in six-dimensional data from the reduced data matrix (PCA components). We can process either 2 PCA components (the default case here) for each classifier using parallel calculation or any selected number of PCA components.

3.5. Secondary PCA reduction. The next step is represented by the secondary PCA-based data reduction, which is applied again for a k-dimensional data according to k emergency signals (classifiers). This process uses the very same algorithm as the primary PCA reduction and in Algorithm 1 it is represented by the function `PCAConstructModuleFunction`.

3.6. Final data. As a result of the secondary PCA reduction, data from emergency signals (classifiers) are transformed into a single data vector of size n. Such a vector represents the output of the proposed reduction and detection model and can be used as an input of another machine/statistical learning-based prediction algorithm for further processing.

Table 2. Execution time of six emergency classifiers for reduced data and for different data sizes.

Execution time building classifiers			
Count of data records (thousands)	Parallel mode [ms]	Sequence mode [ms]	Difference time [ms]
10	12	8	−4
20	10	16	6
50	12	40	28
100	11	55	44
1000	18	414	396
3000	37	1067	1030
6000	78	2035	1957
10000	144	3304	3160
16000	152	3433	3281

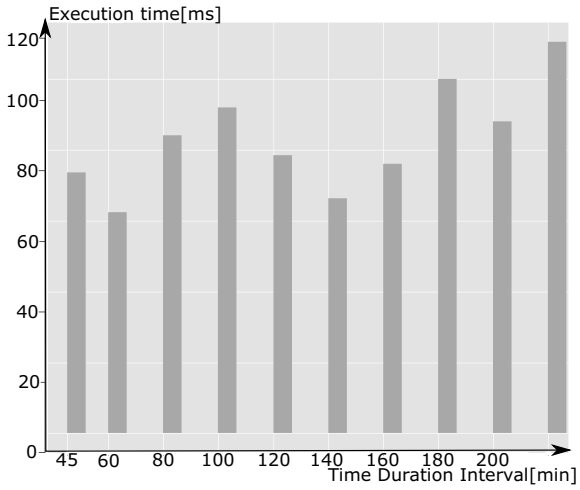

Fig. 4. Total execution time of the algorithm (including preprocessing, classification and data reduction).

3.7. Parallel architecture. We have chosen libraries in the Java platform to calculate decomposition, eigenvectors and eigenvalues (considering the data node structures). Apache Common and Apache Spark were investigated and tested for this purpose. The task was to choose the most reliable implementation for the core calculations. According to the test results presented in Table 1 and Fig. 3, the Apache Common library works faster than Apache Spark. Thus we have decided to select Framework Apache Common as the kernel for calculating decomposition, eigenvectors and eigenvalues. Figure 4 depicts the complete algorithm execution time, including all necessary steps.

Further we have tested the influence of parallel calculation of emergency classifiers on the overall

model performance. Table 2 shows that the parallel processing of multiple classifiers allows us to use bigger data matrices from more data nodes, and to implement complex ensembles of anomaly detectors and classification techniques without a significant increase in computational time.

4. Results analysis

The experiments encompass two case studies, i.e., data from two different machine tools are analyzed. The organization of this section is the following.

Firstly, the illustrative example of distribution analysis of both PCA components (noted PCA1 and PCA2) is given here. Both principal components (PCA1 and PCA2) depicted in histograms, Figs. 5 and 6, are roughly normally distributed. At a first glance, it is not possible to detect significant deviations to the left or right, and obviously, the data do not contain anomalies.

Subsequently, the analysis and graphical visualizations[1] for both the case studies is presented, containing reduced data (signal) from the input matrix as spectral density in Figs. 7(a), 7(b), 8(a), and 8(b), further detection of the highest peak in Figs 7(c), 7(d), 8(c), and 8(d), and other significant peaks in Figs. 7(e), 7(f), 8(e), 8(f). These visualizations are always depicted for both principal components PCA1 and PCA2. Finally, another step is to perform a PCA reduction for a data matrix obtained from an ensemble of six emergency classifiers (see Section 3 for details). Again two principal components are obtained (noted as PCA1f and PCA2f). Their visualization can be seen in 3D plots in Figs. 9(a) and 9(b). Finally, we provide a brief analysis of differences between both the case studies.

4.1. Case Study 1. Figure 7(a) shows the PCA1 component signal obtained from the data reduction input big data matrix of the H-630 machine tool. Within the time ranges from 0 [s] to 800 [s] and from 800 [s] to 820 [s] this signal contains a gaps and long straight lines. In Fig. 7(c), we can see that the system detected one large signal peak based on the maximum dominant signal deviation. Figure 7(e) shows other two significantly large deviation peaks marked with a star symbol. The third peak was not considered a significant anomaly, i.e., not marked. The next step was to analyze the PCA2 component of the condition monitoring tree node. Figures 7(b), 7(d) and 7(f) confirm the same situation as for the PCA1 component. The obtained final PCA1f and PCA2f components are depicted in Fig. 9(a).

As a partial conclusion of Case Study 1, we can claim that an emergency state (anomaly) has been detected on the data node of the condition monitoring tree structure of the H-630 machine tool based on the developed model. Numerous experiments lend weight to the argument that if a large gap can be seen on the PCA components (Figs. 7(a) and 7(b)) with a significant anomaly (peak), and these observations are similar to visualizations of final components, Fig. 9(a), this determines a high accuracy of the anomalies detection of the machine. In general, if the analyzed signal contains both significant gaps on the PCA components graphs and the maximum or minimum signal peaks and there are not many such signal peaks, this supports anomaly detection in an analyzed data node. However, not in all cases. The decision about anomaly detection is precisely determined by the final steps in the presented developed workflow, which are the processing of PCA components with emergency classifiers and subsequent secondary data reduction and final visualization in Fig. 9(a).

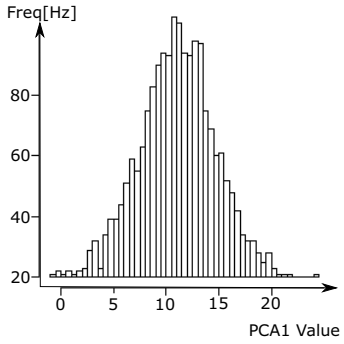

Fig. 5. PCA1 distributed plot.

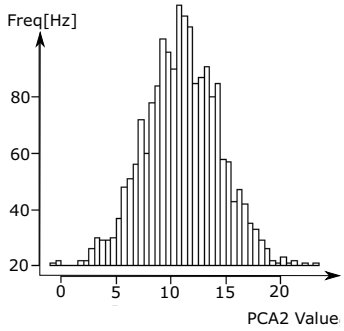

Fig. 6. PCA2 normally distribution plot.

4.2. Case Study 2. The second case study represents the completely different observations for the H-50 machine tool data node. Figure 8(a) shows the PCA1 component, which does not have any long gaps or long straight lines, but on the contrary, it has a bigger number of signal peaks (Fig. 8(c)). Further analysis in Fig. 8(e) reveals a smaller gap and a large number of signal peaks

[1]Full resolution figures are available at `https://ailab.fai.utb.cz/resources/`.

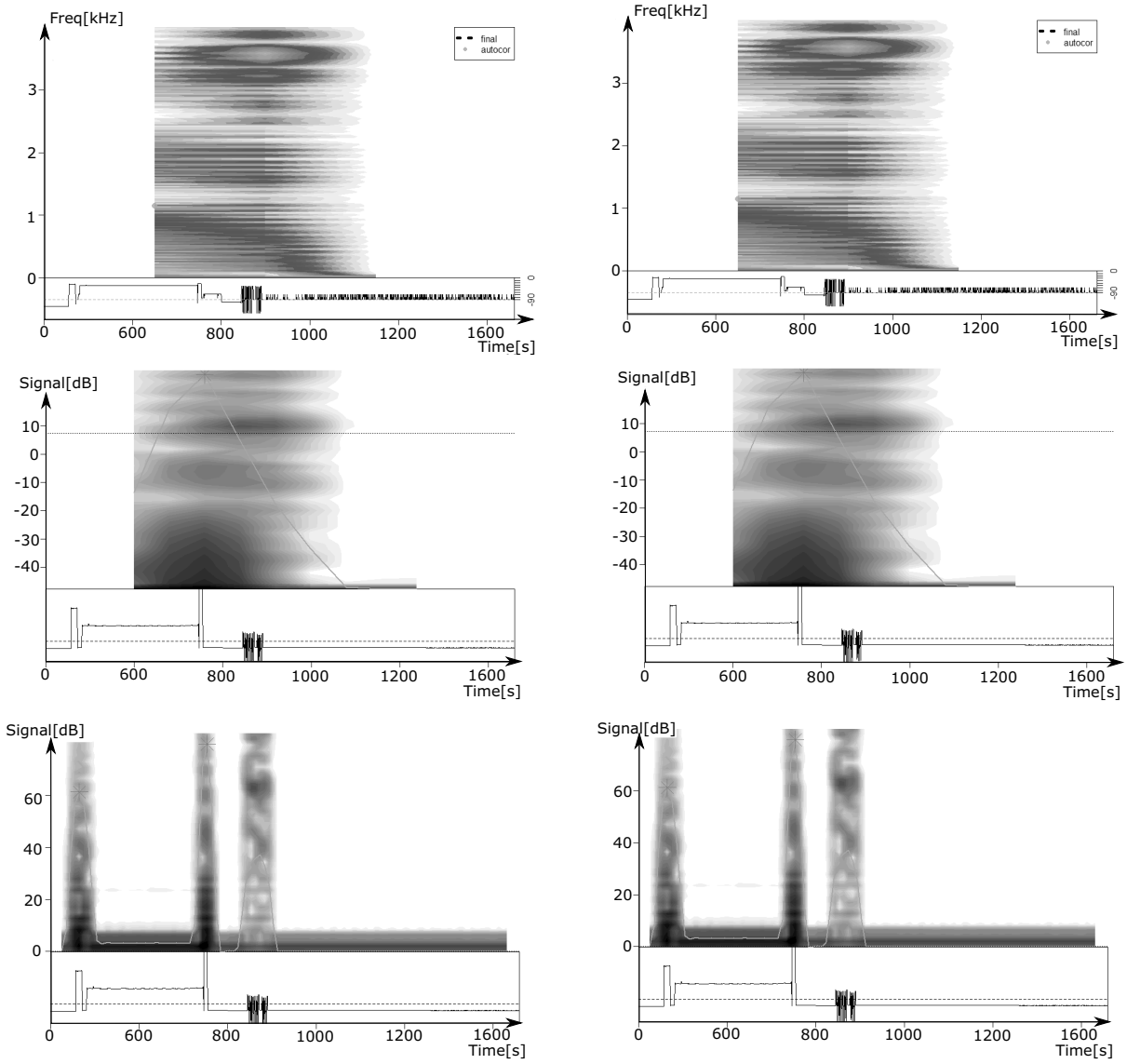

Fig. 7. Graphical outputs for Case Study 1: PCA1 (left) and PCA2 (right) signals. The first row: an emergency state PCA1 (a), PCA2 (b), the second row: emergency state, the highest peak detection PCA1 (c), PCA2 (d), the third row: an emergency state, the highest peaks PCA1 (e), PCA2 (f).

(anomalies).

A similar observable pattern occurs with the second PCA2 component (Figs. 8(b), 8(d) and 8(f)). However, compared with the PCA1, the deviation was only to a minimal extent, which in the end resulted only in a minimal possible collision and a further operation of the machine tool. This collision may not have a substantial effect.

The last step is to perform the final analysis using the developed model, which includes emergency signal classifiers and secondary data reduction to detect emergency states or anomalous collisions. The results are depicted in a diagram, Fig. 9(b), a brief analysis is given in the next section.

4.3. Comparison of case studies. The experiments encompass two different case studies confirming the necessity and accuracy of the proposed reduction and classification model.

In the first case (Fig. 7), the reduced signal from the data node contains one significant peak and, at the same time, a stable section (gap). The second case (Fig. 8) shows a complex signal with a number of peaks. If we focused on the number of peaks and the complexity of the signal, at the first glance, the second case would rather evoke the impression that there is

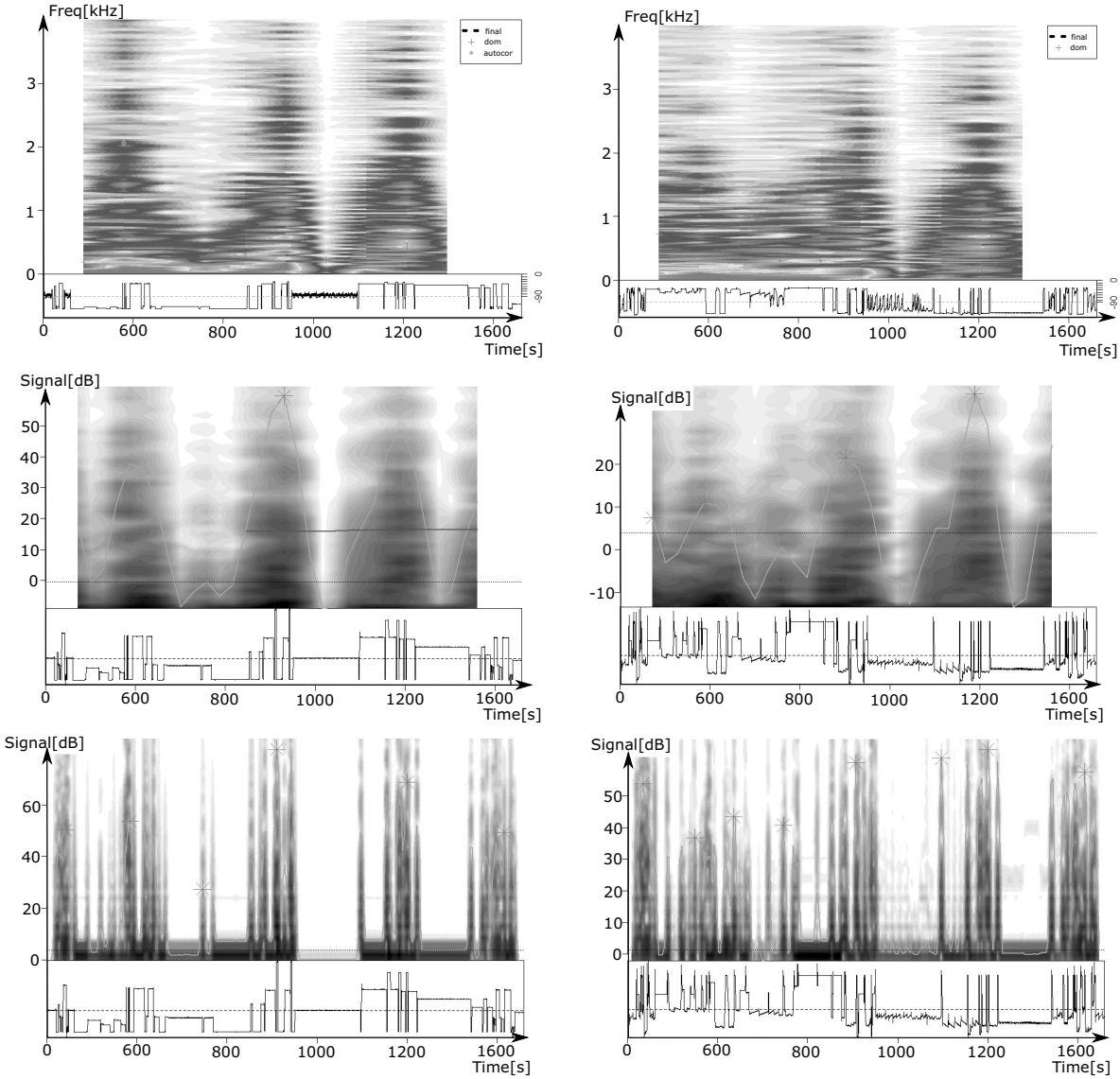

Fig. 8. Graphical outputs for Case Study 2: PCA1 (left) and PCA2 (right) signals. The first row: an emergency state PCA1 (a), PCA2 (b), the second row: an emergency state, the highest peak detection PCA1 (c), PCA2 (d), the third row: an emergency state, the highest peaks PCA1 (e), PCA2 (f).

an emergency state (anomaly). However, subsequent multi-level classification, secondary data reduction and visualization will confirm the opposite (Fig. 9).

Compared with the first case study (Fig. 9(a)), where we can see a similarity between PCA1, PCA2 observations and final reduced data after classification, the final analysis for the second case study (Fig. 9(b)) does not show a significant agreement of the classifiers and an explicit confirmation of the emergency state. For this reason, all graphical outputs of the second case study are referred to as "normal state."

Nevertheless, to be sure, it is necessary to examine the data for another time period and perform quality vibration diagnostics on monitored data (signal) nodes.

5. Conclusions

The developed model based on the PCA reduction with parallel processed core functions build on the Java platform has been inspired by the calculation of accident characteristics or detection of anomalous state principles in recent research (Zhang *et al.*, 2019b; Liang *et al.*, 2019; Żabiński *et al.*, 2019). Our work has presented several original approaches, mainly in the overall workflow, analyses, effectiveness and modular structure with parallel processing and selection of emergency classifiers. The main findings are listed as follows:

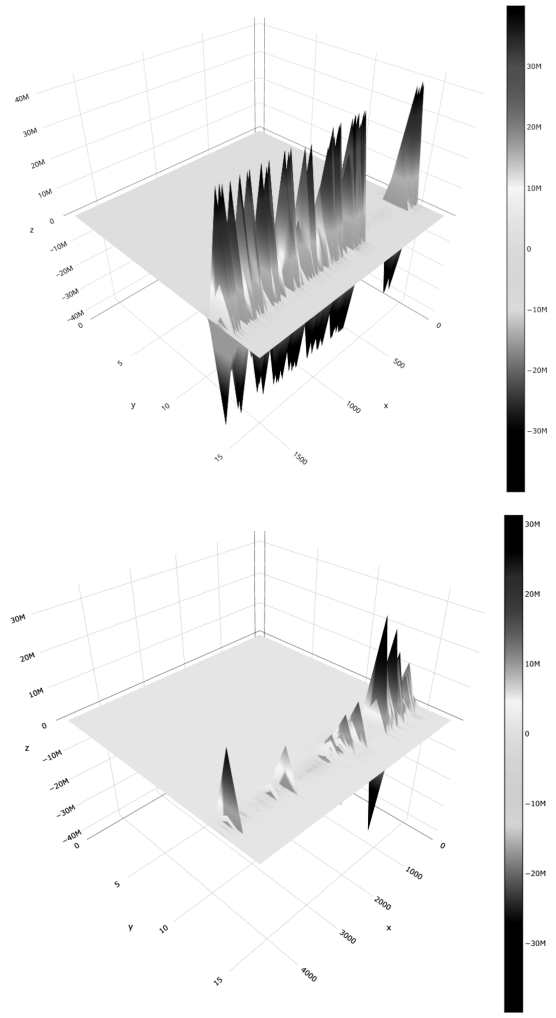

Fig. 9. 3D diagrams for PCA1f and PCA2f: top—emergency state detection, Case Study 1 (a), bottom—normal state detection, Case Study 2 (b).

- Supported by the numerous experiments, investigation, measurements, and research, we can claim that the created model is robust, effective, and able to analyze data and detect emergency states in the production process with a machine tool.
- The presented model and algorithms can process large data frames, which are collected in a dedicated big data node with a tree structure topology and stored in a database repository which supports hierarchical tree topology structures.
- The selected emergency classifiers were entropy, the maximum number of consecutive observations, the maximum level change, the maximum change in rolling means, autocorrelation (ACF1), the variance of residuals or the signal trend.
- Regarding the computational complexity, as an optimal variant for the detection of an emergency state of the machine, one of the classifiers can be chosen, which determines the probability of occurrence of a non-standard behavior of the machine tool. In such a case, the time complexity will be $O(n)$. In the worst case, it is possible to use all classifiers at once, which is presented in this paper. In such a case the accuracy of calculation or detection of an emergency or machine failure will be higher, the time complexity of the algorithm will be greater $O(n \cdot \log n)$.
- As discussed in the partial conclusions for investigated two-case studies, the necessity and accuracy of the whole proposed reduction and classification model including the secondary PCA reduction and an ensemble of emergency classifiers is confirmed by differences in diagrams depicted in Figs. 9(a) and 9(b). These diagrams show different situations, leading to different decisions for H-630 (Case Study 1) and H-50 (Case Study 2) machines tools connected to condition monitoring big data nodes.

Overall, the developed model is able to determine with great accuracy whether or not an anomaly has really occurred on the machine tool. Finally, it is necessary to mention that the presented reduction model and emergency state detection apply only when the machine is operating in normal mode. Because in the unique triggered load modes of the machine, situations arise that cause the machine's components to be overloaded, and the system will recognize this as an anomalous condition. In normal mode, the machine operates so that the machine load mode is not invoked, and therefore situations can be detected when a component fails.

Acknowledgment

This work was supported by the Internal Grant Agency of Tomas Bata University under the project no. IGA/CebiaTech/2021/001, and further by the resources of the A.I.Lab at the Faculty of Applied Informatics, Tomas Bata University in Zlín.

References

Abdi, H. and Williams, L.J. (2010). Principal component analysis, *Wiley Interdisciplinary Reviews: Computational Statistics* **2**(4): 433–459.

Bansal, K. and Bansal, M. (2016). Data clustering and visualization based various machine learning techniques, *International Journal of Advanced Research in Computer Science* **7**(6): 124–128.

Bartenhagen, C., Klein, H.-U., Ruckert, C., Jiang, X. and Dugas, M. (2010). Comparative study of unsupervised dimension

reduction techniques for the visualization of microarray gene expression data, *BMC Bioinformatics* **11**(1): 1–11.

Basora, L., Olive, X. and Dubot, T. (2019). Recent advances in anomaly detection methods applied to aviation, *Aerospace* **6**(11): 117.

Cooper, C., Wang, P., Zhang, J., Gao, R.X., Roney, T., Ragai, I. and Shaffer, D. (2020a). Convolutional neural network-based tool condition monitoring in vertical milling operations using acoustic signals, *Procedia Manufacturing* **49**: 105–111.

Cooper, C., Zhang, J., Gao, R.X., Wang, P. and Ragai, I. (2020b). Anomaly detection in milling tools using acoustic signals and generative adversarial networks, *Procedia Manufacturing* **48**: 372–378.

Ester, M., Kriegel, H.-P., Sander, J. and Xu, X. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise, *KDD'96 Proceedings, Portland, USA*, pp. 226–231.

Filter, L.V. and Filter, P. (2014). Seven techniques for dimensionality reduction, *Technical report*, KNIME, Zurich, `https://www.knime.com/sites/default/files/inline-images/knime_seventechniquesdatadimreduction.pdf`.

Goldin, D.Q. and Kanellakis, P.C. (1995). On similarity queries for time-series data: Constraint specification and implementation, *International Conference on Principles and Practice of Constraint Programming, Cassis, France*, pp. 137–153.

Hankerson, D., Hoffman, D., Leonard, D. and Linder, C. (2000). *Coding Theory and Cryptography: The Essentials*, CRC Press, Boca Raton.

Hyndman, R.J., Wang, E. and Laptev, N. (2015). Large-scale unusual time series detection, *2015 IEEE International Conference on Data Mining Workshop (ICDMW), Atlantic City, USA*, pp. 1616–1619.

Kamat, P. and Sugandhi, R. (2020). Anomaly detection for predictive maintenance in industry 4.0-a survey, *E3S Web of Conferences* **170**: 02007.

Liang, Y., Wang, S., Li, W. and Lu, X. (2019). Data-driven anomaly diagnosis for machining processes, *Engineering* **5**(4): 646–652.

Mühlbauer, M., Würschinger, H., Polzer, D., Ju, S. and Hanenkamp, N. (2020). Automated data labeling and anomaly detection using airborne sound analysis, *Procedia CIRP* **93**: 1247–1252.

Pathria, R. and Beale, P.D. (2011). *Statistical Mechanics*, Academic Press/Elsevier, Cambridge.

Schleinkofer, U., Klöpfer, K., Schneider, M. and Bauernhansl, T. (2019). Cyber-physical systems as part of frugal manufacturing systems, *Procedia CIRP* **81**: 264–269.

Siboni, S. and Cohen, A. (2020). Anomaly detection for individual sequences with applications in identifying malicious tools, *Entropy* **22**(6): 649.

ur Rehman, M.H., Liew, C.S., Abbas, A., Jayaraman, P.P., Wah, T.Y. and Khan, S.U. (2016). Big data reduction methods: A survey, *Data Science and Engineering* **1**(4): 265–284.

Yao, Y.-C., Chen, Y.-H., Liu, C.-H. and Shih, W.-P. (2019). Real-time chatter detection and automatic suppression for intelligent spindles based on wavelet packet energy entropy and local outlier factor algorithm, *International Journal of Advanced Manufacturing Technology* **103**(1): 297–309.

Żabiński, T., Mączka, T., Kluska, J., Madera, M. and Sęp, J. (2019). Condition monitoring in industry 4.0 production systems—The idea of computational intelligence methods application, *Procedia CIRP* **79**: 63–67.

Zhang, C., Song, D., Chen, Y., Feng, X., Lumezanu, C., Cheng, W., Ni, J., Zong, B., Chen, H. and Chawla, N.V. (2019a). A deep neural network for unsupervised anomaly detection and diagnosis in multivariate time series data, *Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, USA*, pp. 1409–1416.

Zhang, L., Elghazoly, S. and Tweedie, B. (2019b). Introducing AnomDB: An unsupervised anomaly detection method for CNC machine control data, *Annual Conference of the Prognostics and Health Management Society, Scottsdale, USA*.

Iaroslav Iaremko has been a PhD student at the Faculty of Applied Informatics, Department of Informatics and Artificial Intelligence, Tomas Bata University in Zlín, Czech Republic, since 2019. He graduated from Masaryk University in Brno in 2015 and obtained an MSc degree in computer systems, joined the Tajmac-ZPS company in 2018 and works there as an analyst programmer.

Roman Senkerik is an associate professor and the head of the A.I.Lab (`https://ailab.fai.utb.cz/`) since 2017 at the Department of Informatics and Artificial Intelligence, Tomas Bata University in Zlín. He is the author of more than 30 journal papers, 250 conference papers, several book chapters, and editorial notes. His research interests are the development of evolutionary algorithms, modifications and benchmarking of EAs, soft computing methods, as well as interdisciplinary applications in optimization and cyber-security, machine learning, neuroevolution, data science, the theory of chaos, and complex systems.

Roman Jasek is a full professor and the head of the Department of Informatics and Artificial Intelligence, Tomas Bata University in Zlín. His research interests include systems engineering, cryptography and cybersecurity in close connection to artificial intelligence applications, as well as biometric identification and control of systems using brain centre signals (BCI, BMI, EEG).

Petr Lukastiik is with the Tajmac-ZPS company in the Czech Republic.

Received: 11 June 2021
Revised: 19 August 2021
Re-revised: 14 October 2021
Accepted: 19 October 2021