Analysis of the RBF ANN-based classifier for the diagnostics of electronic circuit

Bartosz Połok, Piotr Bilski

Institute of Radioelectronics and Multimedia Technology, Warsaw University of Technology, ul. Nowowiejska 15/19 00-665 Warsaw, Poland

ABSTRACT
The paper presents the application of the Radial Basis Function (RBF) Artificial Neural Network (ANN) to the diagnostics of analog circuits. Such networks are in most cases useful in the approximation tasks as the alternative to multilayered perceptrons (MLP) or Support Vector Machines (SVM). In this work the analysis of various RBF ANN-based classifier configurations for the fault detection and identification module is conducted. The considered parameters included the optimal number of neurons in the hidden layer, coding schemes for the output layer and operation duration during the training and testing of the classifier. The efficiency of the diagnostic system was verified using the electronic circuit, i.e. the fifth order lowpass filter. It was analyzed in terms of testability, depending on the set of accessible nodes, confronted against the availability of the output node only. Experiments also covered accuracy comparison between the RBF, MLP and SVM classifiers. Results show advantages and drawbacks of the RBF ANN-based diagnostic module, compared to other available solutions.

1. INTRODUCTION

Artificial Neural Networks (ANN) are currently the most popular Artificial Intelligence (AI) tools used in the diagnostics of analog systems [1]-[3]. Their numerous advantages include the ability to extract generalized knowledge from the available measurement data, autonomous operation (making them useful in the automated online diagnostics) and the ability to accurately process data in uncertainty conditions. Limited memory usage and high processing efficiency make them useful in embedded applications (implemented in the digital signal processor or the FPGA array [4]). Their disadvantage is the obscure form of stored knowledge (illegible for the human operator), making the explanation of generated decisions difficult. However, this is not crucial as long as the diagnostic system accurately detects and identifies faults. Despite introducing hybrid approaches (such as hierarchical or

convolutional ANN [5]), traditional architectures are still in use, due to their simple operation principle and high efficiency proven in multiple cases.

The most popular ANN is the Multilayered Perceptron (MLP), successfully applied to solve biomedical, financial and technical problems [6], [7]. Their implementations in the diagnostics cover power lines [8] or electrical machines [9]. They are currently widely exploited and work as the standard diagnostic tool, or the self-diagnostic module [10]. Training the MLP classifier consists in selecting one of available algorithms (such as error backpropagation or Levenberg-Marquardt) and adjusting the available data to the predefined architecture. It is defined by the specific number of layers that contain the selected type of neurons (computational units). Optimizing these parameters is the main design challenge, as the optimal MLP architecture has to be determined individually for each

problem.

Similar issues are encountered during the application of Support Vector Machines (SVM) to the classification task. They are considered the optimal ANN-based classifiers in the uncertainty conditions [11]. Their disadvantage is the time consuming process of selecting the optimal kernel and its parameters [12]. Despite such challenges, SVM are also popular in diagnostics, used to identify faults in electrical machinery [13], electronic circuits [14] or power plants [15].

On the other hand, Radial Basis Function (RBF) networks are considered simpler in design. Their training is faster and structure simpler, as only the single hidden layer is present. Therefore, the following parameters are to be determined:

- the number of hidden computational units, depending on the size of training data;
- the number of output neurons, depending on the number of categories to distinguish and the selected coding scheme;
- the activation function width (identical for all hidden neurons).

In most cases the RBF network is used for the approximation task, as it has linear neurons in the output layer. Its application to classification may require substituting computational units with their sigmoidal counterparts. Such approaches are rare and not well described in the literature [16]. This calls for the thorough investigation of the RBF ANN classification characteristics during the fault detection and identification in analog systems.

The aim of this paper is to assess performance of various RBF ANN configurations during the fault detection and location in analog circuits. The presented work is an extension of the research published in [17]. All introduced parameters of the classifier were tested to find the optimal configuration maximizing the diagnostic accuracy, verified on the model of the 5th order electronic lowpass filter. The testability of the circuit was checked by changing the set of accessible nodes (at which system responses can be measured). The accuracy of the AI-based diagnostic module depends on the quality of the training data extracted from the system. Increasing the number of accessible nodes leads to the improvement of the diagnostic module. On the other hand, the number of pins in the circuit casing should be minimized. Therefore, a compromise between these two goals must be made.

Finally, the RBF ANN-based classifier is compared to its well-established counterparts, i.e. MLP and SVM to determine in which situations it outperforms its rivals and should be selected for the task.

The structure of the paper is as follows. Section 2 presents the applied diagnostic architecture. Details of the implemented network are described here. Training and testing data sets structure is presented in Section 3. The analysed circuit is introduced in Section 4, while experimental results are in Section 5. Conclusions about the conducted analysis are in Section 6.

2. ANN-BASED DIAGNOSTIC ARCHITECTURE

The typical diagnostic scheme using any type of the ANN-based classifier is presented in Figure 1. Here the System Under Test (SUT) is monitored by the hardware/software module, which performs data acquisition on the accessible nodes. In this way the extraction of the vector of features (symptoms) $e=\{s_1, \ldots, s_m\}$ (also called examples) from measured signals is
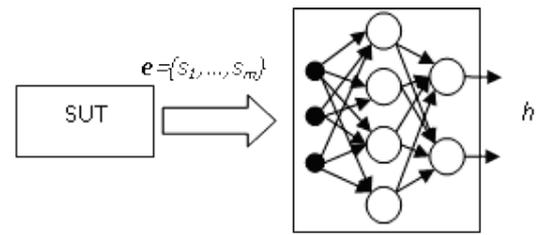

Figure 1. ANN-based diagnostic system architecture.

performed. They are subsequently processed to make the diagnostic decision (hypothesis) h. It is assumed that knowledge stored by the module allows for the automatic fault detection and identification. This requires implementation of the machine learning algorithms. During this operation the ANN-based classifier produces the binary output, in which every fault category is represented by the unique sequence of values $\{0,1\}$ or $\{-1,1\}$, depending on the used type of sigmoidal neurons.

Application of the RBF network for the fault classification task requires adjusting its structure to the specific problem. This involves selection of the number of neurons in the hidden layer and the coding scheme for binary units in the output layer. As opposed to MLP and similarly to SVM, the RBF network contains only one hidden layer with the Gaussian activation function $\varphi(s_i)$ units (1). The function width (defined as the standard deviation σ, where c is the centre, usually set to 0 for all neurons) determines the network performance. These parameters are optimized during the design stage.

$$\varphi(s_i) = \exp\left(-\frac{\|s_i - c\|^2}{2 \cdot \sigma^2}\right) \qquad (1)$$

2.1. Minimization of the number of hidden neurons

The number of neurons k in the hidden layer (usually greater for the RBF ANN than for MLP) is automatically adjusted in the result of the training. Although there are approaches to find it during one training sweep [18], in most cases the predetermined network structure is trained and validated, then the process is repeated for another structure. During this time consuming (though automated) process, the maximum acceptable value is determined. It is initially equal to the overall number of examples in the set L (2), which ensures the high classification accuracy, but makes the network large and slow. To avoid this, the number of hidden neurons should be minimized. The training examples are obtained after simulating the SUT, which allows for connecting its behaviour with the internal configuration of parameters.

In the presented work, the clustering of examples was applied to group the most similar ones. It is assumed that training data from the SUT simulation is redundant, caused by the following factors:

- low sensitivity of the SUT for changes of the specific parameter, which results in multiple examples describing the same category, as they are close to each other (in terms of distance). Such a group can easily be substituted by the single example representing all original members. It becomes their centroid.
- existence of ambiguity groups (AG) [19], which contain examples belonging to different categories that are close to each other. In this case examples cannot be replaced by the single representative and must all remain in the set.

Data preprocessing aims at discovering all groups of similar examples by the cluster analysis, which results in the set of groups G_l easily distinguishable from each other, based on the similarity calculations. The detailed scheme is presented in Figure 2. Redundancy in the set L may be exploited in two ways. The first one consists in providing the original data set for the RBF network training, but limiting the maximum number of neurons in the hidden layer to the number of clusters n_c. Alternatively, the reduced data set L' may be provided for training.

The distance-based similarity calculation was used in the presented case. Its only parameter is the threshold θ, below which two examples e_i and e_j are considered close to each other (2). The measure applied to create clusters exploits Euclidean d_E (3) and cosine d_c (4) distances, treating every example as the point in the m-dimensional space. This way groups located close to each other are easily identified. Thresholds θ_1 and θ_2 for both distances should be selected adaptively to minimize the number of clusters containing examples belonging to different classes (i.e. forming AG).

$$e_i, e_j \in G_l \Leftrightarrow d_E\left(e_i, e_j\right) < \theta_1 \wedge d_E\left(e_i, e_j\right) < \theta_2 \qquad (2)$$

$$d_E\left(e_i, e_j\right) = \sqrt{\sum_{k=1}^{m}\left(s_{ik} - s_{jk}\right)^2} \qquad (3)$$

$$d_c = \frac{\sum_{k=1}^{m} s_{ik} \cdot s_{jk}}{\sqrt{\sum_{k=1}^{m} s_{ik}^2} \cdot \sqrt{\sum_{k=1}^{m} s_{jk}^2}} \qquad (4)$$

The illustration of the clustering method for $m=2$ is presented in Figure 3.

2.2. Output layer coding schemes

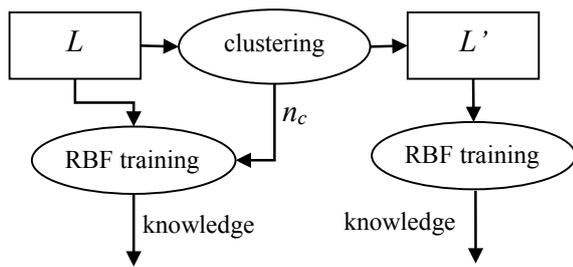The important issue during the design of the ANN-based classifier is the representation of multiple categories by binary-valued neurons in the output layer. Among multiple coding schemes, three were selected for experiments [20]. It is assumed that only single faults are considered (the most probable case). Schemes differ in the number of output neurons o:
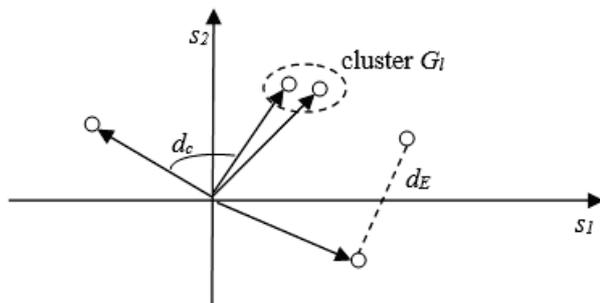
- One-vs-All (OvA) – here each output unit is responsible for a separate category. The number of fault codes l determines the number of neurons o in the output layer (see Table 1). During the fault identification, only one neuron should be active. If multiple neurons become active, they can be interpreted as fault candidates in the uncertainty conditions. This scheme makes the classifier suitable for the multiple fault diagnostics (i.e. when more than one parameter is out of tolerance margins).

- One-vs-One (OvO) – each neuron is responsible for distinguishing between the pair of categories. This time o classifiers with one output neuron are created and simultaneously trained. Their number is determined by the number of distinct fault code pairs: $l \cdot (l-1)/2$. In this scheme both active and inactive neuron determines the specific fault. The diagnostic decision is made based on the majority voting – the category pointed by the greatest number of networks supporting it. Note that each network in this scheme is trained on a different subset, extracted from L. In each, only examples belonging to two fault categories are present. For instance, three fault codes require the following units (trained on the corresponding subsets): $1: c_1$ vs c_2, $2: c_1$ vs c_3, $3: c_2$ vs c_3.

- Minimum Output Coding (MOC) – the minimal set of neurons, which represents each fault code by the unique combination of active units. The subsequent codes are represented by the binary representation of integer numbers (see Table 2).

The more sophisticated coding schemes, such as Error Correcting Output Coding (ECOC) [21] were excluded from experiments, as their implementation would complicate the output layer even more, not necessarily increasing diagnostic accuracy. ANN was implemented using the Matlab environment including the Neural Networks Toolbox.

3. DATA SETS DESCRIPTION

Learning L and testing T datasets are required to train the RBF network classifier and verify its accuracy. Both have the

Figure 2. Learning data set preprocessing for the RBF classifier training.

Figure 3. Illustration of the clustering method.

Table 1. Applied OvA coding scheme.

Category	Coding scheme
c_1	1,0,0, …,0
c_2	0,1,0, …,0
…	…
c_l	0, …,0,0,1

Table 2. Applied MOC coding scheme.

Category	Coding scheme
c_1	0,0,0, … ,0,1
c_2	0,0,0, … 1,0
c_3	0,0,0, … 1,1
…	…

same structure, containing *n* feature vectors **e** with *m* attributes (symptoms *s*), supplemented by the fault code, describing the state of the SUT. The generic form of the set (5) is the table, whose structure is identical for *L* and *T*.

$$L \equiv T = \begin{bmatrix} s_{11} & \cdots & s_{1m} & c_1 \\ \vdots & \ddots & \vdots & \vdots \\ s_{n1} & \cdots & s_{nm} & c_l \end{bmatrix} \qquad (5)$$

The single example is generated after simulating the SUT model with the fault introduced to its structure. This way the effect of the fault on the responses measured in accessible nodes can be observed. The key issue is the generation of the fault code for each example, based on the actual value of the faulty parameter. As before [22], it is the integer number, being the combination of the parameter identifier and its deviation from the nominal value. The code contains the identifier of the faulty parameter and the discrete degree of its change, both negative and positive (for values greater and smaller than the nominal one). The number of parameter states depends on the assumed thresholds for deviations. For instance, if the parameter value is above ten percent of nominal, the degree is labeled with "1" or "-1", indicating it being "larger than" or "smaller than", respectively (depending on the direction of deviation). Additional intervals may be introduced to increase the resolution of the diagnostic module. For instance, if the value of the parameter is above fifty percent of the nominal one, the degree "2" may be assigned. This way the second parameter with the value larger than the nominal one is represented by the code "21", while the first parameter with the value much smaller than the nominal one obtains the code "-12". The exception is the nominal state, encoded with the value "0". The number of simulated parameter states depends on the expected data set size and the accuracy of the SUT modeling. In the presented work, both sets are of the same size, containing mutually exclusive examples. This way the generalization of the ANN RBF-based diagnostic module may be verified.

Diagnostic accuracy *acc* of the classifier is measured using the set *T*, for which the number of incorrectly classified examples (i.e. the ones for which the hypothesis *h(**e**)* is not equal to the fault code *c(**e**)*) determines the sample error e_s:

$$acc = 1 - e_s = 1 - \frac{|e \in T : c(e) \neq h(e)|}{|T|} \qquad (6)$$

Each SUT simulation was performed after changing the single parameter value beyond the tolerance margins (here 10% of the nominal value) with all other parameters remaining nominal. A subset of fault-free examples was also prepared to check the resilience of the module to false alarms.

4. ANALYZED SUT

The fifth-order analog filter is a good example to implement the data processing methods. Similar active circuits are still popular in military or acoustic applications, therefore their analysis is justified [23]. It is complex (because of the large number of nodes and SUT elements) and difficult to diagnose based only on the analysis of the output node. Therefore, multiple symptoms are needed to identify the state of all elements (resistances and capacitances). The circuit in Figure 4 contains ten elements with the following nominal values: $R_1 = R_2 = R_3 = R_4 = R_5 = 1$ kΩ, $C_1 = 16$ nF, $C_2 = 19$ nF, $C_3 = 13$ nF, $C_4 = 51$ nF and $C_5 = 49$ nF. Subsequent resistances were labelled
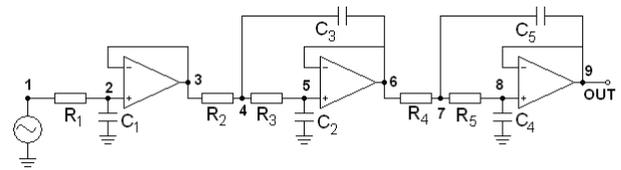

Figure 4. Scheme of the 5th order lowpass filter.

with numbers from 1 to 5 respectively, while capacitances were referred to as parameters No. 6 to 10. The cutoff frequency of the filter for such values is 10 kHz. The model of the circuit was implemented in the Simulink environment with the operational amplifiers realized as equivalent circuits with controlled voltage source. Simulations were performed to obtain examples of the SUT behaviour for different values of elements (changed up to 90 percent of the nominal value). The excitation signal provided at node No. 1 was a sinusoid with 9 kHz frequency (i.e. close to the cutoff frequency). The filter was analysed in the time domain, where at the accessible nodes sinusoidal responses were recorded. Measurements were taken at nodes 2, 3, 5, 6, 8 and 9. From each response the first three maximal and minimal values of the signal with their time instants and time instants of zero crossings are extracted (Figure 5). This gives the total number of 54 symptoms for each example (nine for each node), assuming all discussed nodes are accessible.

To evaluate the dependency between the size of the set and the accuracy of RBF-based classifier, various sizes of the sets were prepared, starting from 70 examples (7 simulations for each parameter, including the nominal state) for the set L_1 to 180 (18 simulations for each parameter) for the set L_2. To consider tolerances, additional sets were created with results of simulations affected by the random value added to the actual parameter. This allows for verifying if random deviations of parameter values influence the ability of the diagnostic module to distinguish faults. The number of different fault codes to distinguish was 41, as not all 5 were used for each parameter due to their smaller sensitivity.

5. EXPERIMENTAL RESULTS

Conducted experiments consisted of four stages. First, coding schemes were compared. Next, relation between the size
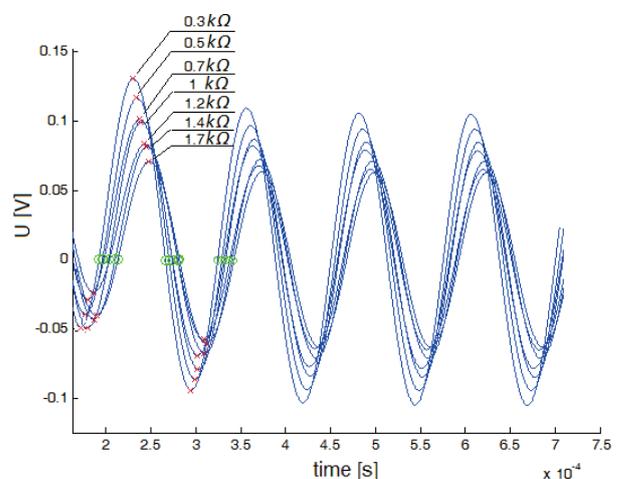

Figure 5. Examples of responses from the filter for various values of the resistor R_1.

of the data set and the diagnostic accuracy was determined. The testability of the circuit, depending on the number of analysed nodes, was also verified. Finally, comparison between RBF, SVM and MLP (also implemented in Matlab) was performed to determine efficiency of all classifiers and find their advantages and drawbacks. The ANN training is a random procedure, consisting in the adjustment of weights between neurons to the training data. Knowledge structure depends on the initial (randomly selected) values of weights, therefore every training sweep results in different content of weights matrices. Therefore, to gain the general information about the ANN efficiency, training and testing experiments were repeated ten times for each architecture and data set. Results presented in the following subsections refer to average accuracy values.

5.1. Coding schemes verification

Coding schemes presented in Section 2.2 were implemented to design the RBF network, train it on the set L and test using the set T. The first operation is done according to the Simulation Before Test (SBT) [24], [25] paradigm, i.e. when knowledge is extracted prior to the diagnostic module implementation in the industrial environment. Duration of the operation is of secondary importance here. The trained network is then used to process measurement data and make decisions. As the operation duration of ANN in the testing phase is negligible, there are virtually no restraints, even for their on-line application (where the diagnostic decision must be made within the specific time limitations). Optimal results for all configurations (regarding the width of the RBF σ, number of output neurons o, number of hidden neurons k, the maximum achieved training error e and obtained accuracy on the set T) for the set L_2 (180 examples) are in Table 3.

In most considered cases, the OvO coding is the best scheme, producing the minimum error for all network configurations. This method is the most complex, requiring multiple ANN, each responsible for only two faults. This leads to many networks, but with relatively simple structure. The average number of hidden neurons for the optimal OvO network structure was 8. The most complex is MOC, as multiple output neurons have to participate in producing the fault category. For each coding the best width of the Gaussian function was also determined. The full sweep optimization procedure was applied to obtain it (although more sophisticated approaches, such as simulated annealing [12] or genetic algorithm can be used). The optimal RBF width and the number of hidden neurons are in all cases in the middle of the verified range (for σ it is between 0.1 and 2.0). Figure 6 shows the relation between σ and the obtained accuracy (with all other parameters set as in Table 3). For all coding schemes, there is the best σ value, for which the minimum error is obtained. Results show that narrow Gaussian functions are preferred (like $\sigma = 0.5$ for OvA). The cost of the highest OvO accuracy is the
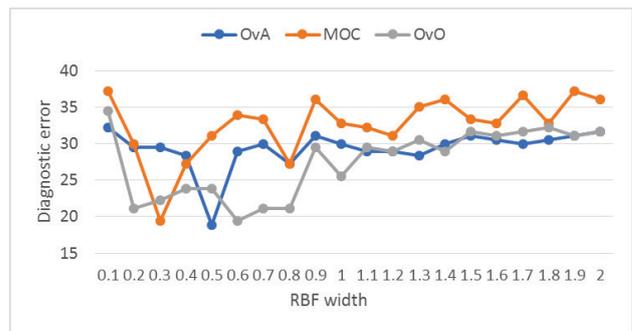

Figure 6. Influence of the RBF width on the diagnostic accuracy.

greater computation effort, caused by the large number of trained networks in comparison to the single network used for remaining coding schemes.

The experiments with the optimization of the number of hidden neurons show similar effects as during the selection of the Gaussian function width. The discrete sweep was implemented here to train and test the network while the number of hidden neurons was incremented (i.e. increased by one). At the specific point the maximum accuracy is reached. Adding more computational units does not improve the ANN efficiency but increases the training duration. This concludes the structure optimization stage, with results of the OvO coding presented in Figure 7. As each network distinguishes between two categories only, it is simpler than the ones implementing other coding schemes.

The detailed analysis of the network performance shows that most diagnostic errors are made because of mistaking the first element (R_1) with the sixth one (C_1). In such a case, the direction of changes is correct (for instance, the typical error is producing the code "-62" instead of "-12"). Changes in elements for which the SUT has low sensitivity (especially R_2, R_3 and C_3) are difficult to detect and the classifier regards the circuit as nominal. In such cases (when all symptoms are within tolerance margins) categories of such examples should be changed to "0". There are also errors of minor importance, i.e. the incorrect identification of the intensity of changes, although the element is correctly located (for instance, when the code "-52" instead of "-51" is generated). Such mistakes should be treated as less important and regarded with smaller weights during the accuracy (6) calculation.

5.2. Data set size

Experiments regarding the size of the training set L were divided into two steps. In the first one, various numbers of

Table 3. Comparison of performance of RBF network output coding schemes.

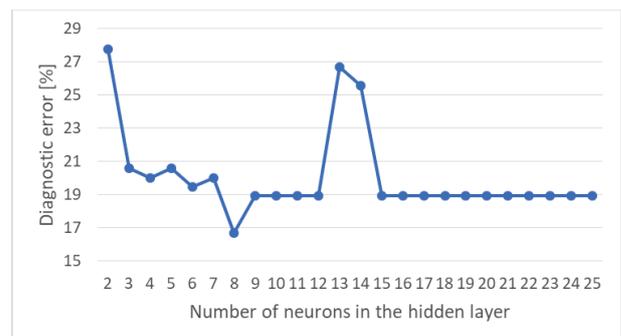| Coding scheme	σ	k	o	e	acc [%]
OvA	0.5	82	41	0.0	81.11
OvO	0.6	8	820	0.0	83.33
MOC	0.3	95	6	0.02	80.56

Figure 7. Influence of the number of hidden neurons on the diagnostic accuracy (OvO coding).

examples were inserted to the set and then used to train the RBF classifier (starting from the set L_1 with the minimal number of symptom vectors, through the medium set with added some new examples, and ending with the maximal set L_2). The initial diagnostic results for L_1 are significantly improved by introducing additional examples, provided they carry useful information. Table 4 compares the diagnostic accuracy for data sets of different sizes. As in other applications, the amount of training data influences the accuracy of the classifier, which may be a problem if they are difficult to collect (for example, when the model of the SUT does not exist or requires large amount of calculations).

During the second stage, the original set is processed to group the most similar examples (see Section 2.1) to compress their number. Additionally, dependency between the training duration and the size of processed data should be established. This is done using the clustering algorithm. In this way it is possible to eliminate redundant examples, which in turn would be represented by the single vector of symptoms. Results of the diagnostic procedure for the RBF ANN trained on the minimized version of the set are presented in Table 5. Increasing the number of examples improves the fault detection and identification performance, but to obtain the accuracy equal to the RBF trained on the L_2 set, the density of clustering must not be large. Comparable diagnostic results are obtained on data sets with size close to L_2. Again, the most effective is the OvO scheme.

Relation between the RBF network structure and training duration for the OvA and MOC coding is in Figure 8. All computations were performed on the computer equipped with the AMD FX-8320 processor (eight cores, 3.50 GHz) and 16 GB of RAM. The processing time for both mentioned coding schemes depends on the number of hidden neurons in the network structure. The relation between the size of the data set and the number of neurons is more complex, as even larger data sets may be optimally processed by smaller networks. The training duration of the RBF ANN-based classifier with OvO coding is significantly longer, requiring between 10 and 20 minutes to complete on the same hardware configuration. The time curves are close to linear, which is the positive effect. The OvO RBF classifier requires the greatest amount of time to
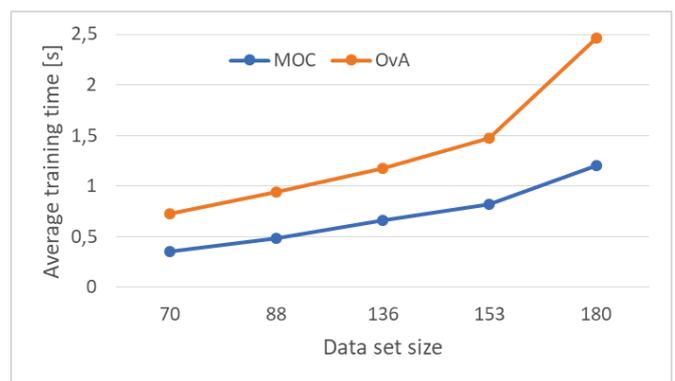

Figure 8. Training duration of the RBF network (OvA and MOC coding schemes).

train because of a large amount of networks to process. This calls for the parallel implementation of the training procedure [26], where each network would be processed by a separate processor, core or computer. As this procedure is implemented off-line, its duration is of secondary importance, although considering the thorough values optimization of σ and k for each network will complicate the process even more.

Another problem is the testing duration, as it determines the usability of the diagnostic module in the on-line mode. Average times required by the RBF ANN with subsequent coding schemes to process the single set of symptoms are in Figure 9 (on the same computer configuration as before). Although much faster for all schemes, the processing time for OvO coding is significantly longer, making it difficult to implement in the embedded system working under strict time limited conditions.

5.3. Testability verification

Experiments presented so far were conducted on the complete data sets, i.e. containing symptoms extracted from all circuit nodes, as discussed in Section 4. Because of practical reasons it is rarely possible to use all of them (especially in the integrated system). Therefore, the number of diagnostic pins available from the circuit casing should be minimal. The compromise between the accuracy and the set of accessible nodes must be made. Figure 10 presents comparison between different coding schemes of the optimal RBF network structure for different sets of accessible nodes. Besides the node 9 (always observable, therefore usable in the input-output analysis [24]), all other**s** can be theoretically made accessible. Because the filter has the sequential structure, it is reasonable to assume

Table 4. Influence of the data set size on the RBF classifier performance.

Data set size	acc_{OvA} [%]	acc_{MOC} [%]	acc_{OvO} [%]
70	68.57	67.14	65.71
120	75.78	73.86	75.54
180	81.12	80.56	83.33

Table 5. Influence of the data clustering on the RBF classifier performance.

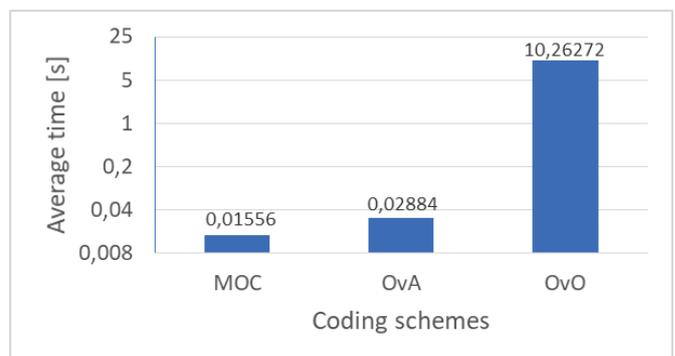| Data set size	acc_{OvA} [%]	acc_{MOC} [%]	acc_{OvO} [%]
70	68.57	67.14	65.71
88	70.67	69.56	68.23
103	72.23	68.34	71.12
109	70.00	70.56	71.12
136	75.89	73.45	75.78
153	76.67	72.23	77.42
162	81.12	72.78	81.89
180	81.12	80.56	83.33

Figure 9. Duration of processing the single vector of symptoms by the RBF ANN-based diagnostic module.
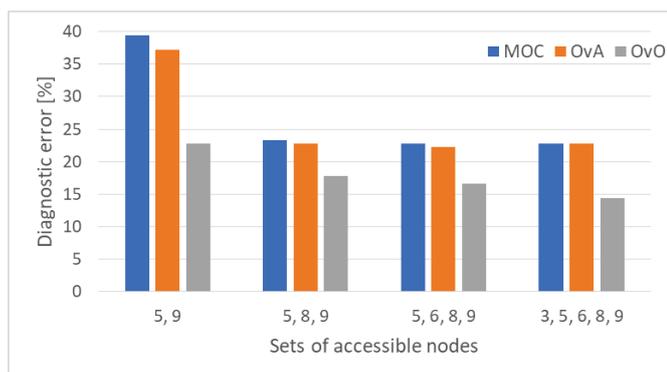
Figure 10. Testability analysis for various sets of ANN coding schemes.

that more information can be recovered from the nodes close to the output of the SUT. Therefore, the order of adding them to the analysis was as follows: 8, 6, 5, 3, and 2. The key step for improving the accuracy was the addition of the nodes 5 and 8. Further extensions led to only small change in diagnostic outcomes. Again, in all experiments the OvO coding proved to be the most effective. The procedure for other similar circuits (such as other filters) should be the same. If the system is more complex, more sophisticated node selection method should be used [27].

5.4. Comparison against other classifiers

Besides the RBF-based network classifier, also SVM and MLP can be used to process the same data sets. In Tab. 6 diagnostic outcomes of the optimal network configurations for the largest set T_2 are presented. Performance of classifiers is comparable, the main difference lies in the training time, which is the longest for the MLP and comparable for RBF and SVM. In the latter case, additional parameters must be determined (such as the kernel function and its parameters), which significantly increases the training duration. The RBF ANN-based classifier is the fastest one to design and has simpler training algorithm than competitors. Its optimization is also less complex than SVM. This classifier can then be used during the fault analysis of SUTs, which are moderately difficult to diagnose. In the case of wide element tolerances (for cheap circuits) or high level of noise, more sophisticated ANN (such as SVM) may be preferred.

6. CONCLUSIONS

The paper presented analysis and optimization of various RBF network-based classifier in the diagnostics of analog circuits. Multiple parameters, including the width of the activation function, number of neurons in the hidden layer and coding scheme in the output layer were verified. The testability of the circuit was also analysed.

The networks with OvO scheme were the best in most cases, but required the greatest amount of time to be trained. This is because this scheme requires creating a large number of

Table 6. Comparison of the ANN-based diagnostic modules (trained on the set L_2).

Data set size	acc_{OvA} [%]	acc_{MOC} [%]	acc_{OvO} [%]
RBF	81.12	80.56	83.33
MLP	80.75	80.56	79.80
SVM	81.12	81.12	79.80

simple networks, distinguishing between only two fault categories. The MOC, although requiring the smallest number of output neurons is more susceptible to random classification errors. The OvA scheme is the most popular one. Its accuracy is similar to OvO, but it is much faster trained. Selection of the scheme is a compromise between the expected accuracy and available time for training and processing the testing vector of symptoms (when the on-line diagnostics is considered).

The optimization of the RBF training process is similar to other ANN-based classifiers. The supervised learning is implemented here as the standard parameterized algorithm. The designer's task is to configure the network (by determining the number of hidden neurons or values of the activation function parameters) to ensure the highest diagnostic accuracy. This process is time consuming, but results in the improvement in the diagnostic accuracy. Multiple continuous or discrete optimization methods may be used for this purpose, regarding both network structure and the SUT testability.

Future research should cover analysis of other systems belonging to various technical domains. Successful implementation of the presented diagnostic scheme to their analysis would confirm usefulness of the RBF ANN in the fault detection and identification task. Also, more sophisticated approaches should be developed for the selection of the accessible nodes, especially in complex systems [10] with many parameters to determine. Finally, parallel implementations of the OvO coding scheme would eliminate its main disadvantage i.e. the long duration of both training and testing phases.

REFERENCES

[1] M.-Y. Chow, R.N. Sharpe, and J.C. Hung, "On the application and design of artificial neural networks for motor fault detection," IEEE Transactions on Industrial Electronics, Vol. 40, No. 2, 1993, pp. 181 - 188.

[2] Y. Cheng, R. Wang, and M. Xu, "A Combined Model-Based and Intelligent Method for Small Fault Detection and Isolation of Actuators," IEEE Transactions on Industrial Electronics, Vol. 63, No. 4, 2016, pp. 2403-2413.

[3] B. Aubert, J. Régnier, S. Caux, and D. Alejo, "Stator Winding Fault Diagnosis in Permanent Magnet Synchronous Generators Based on Short-Circuited Turns Identification Using Extended Kalman Filter", Acta IMEKO, Dec. 2014, Vol. 3, No. 4, pp. 4-9.

[4] S. Shreejith, B. Anshuman, and S.A. Fahmy, "Accelerated Artificial Neural Networks on FPGA for fault detection in automotive systems," Design, Automation & Test in Europe Conference & Exhibition (DATE), 14-18 March 2016, Dresden, Germany.

[5] L. Jing, M. Zhao, P. Li, and X. Xu, "A convolutional neural network based feature learning and fault diagnosis method for the condition monitoring of gearbox," Measurement, Vol. 111, 2017, pp. 1-10.

[6] A.S. Volosnikov and A.L. Shestakov, "Neural network approach to reduce dynamic measurement errors", Acta IMEKO, Nov. 2016, Vol. 5, No. 3, pp. 24-31.

[7] T.M. Kwon and E.H. Feroz, "A multilayered perceptron approach to prediction of the SEC's investigation targets," IEEE Trans. Neural Networks, Vol. 7, No. 5, Sep 1996, pp. 1286-1290.

[8] M.N. Mahmud, M.N. Ibrahim, M.K. Osman, and Z. Hussain, "Selection of suitable features for fault classification in transmission line," IEEE International Conference on Control System, Computing and Engineering (ICCSCE), 27-29 Nov. 2015, DOI: 10.1109/ICCSCE.2015.7482253.

[9] J. Penman, C.M. Yin, "Feasibility of using unsupervised learning, artificial neural networks for the condition monitoring of electrical machines," IEE Proceedings - Electric Power

Applications, Vol. 141, No. 6, Nov 1994, pp. 317-322.

[10] D. Zaleski, R. Zielonko, "Two-functional µBIST for Testing and Self-Diagnosis of Analog Circuits in Electronic Embedded Systems", Acta IMEKO, Dec. 2014, Vol. 3, No. 4, pp. 10-16.

[11] H. Ma, T.K. Saha, and C. Ekanayake, "Power transformer insulation diagnosis under measurement originated uncertainties," IEEE Power and Energy Society General Meeting, 25-29 July 2010, DOI: 10.1109/PES.2010.5589395.

[12] P. Bilski, "Automated selection of kernel parameters in diagnostics of analog systems," Przegląd Elektrotechniczny, No. 5, 2011, pp. 9-13.

[13] R. Fang and H. Ma, "Application of MCSA and SVM to Induction Machine Rotor Fault Diagnosis," Sixth World Congress on Intelligent Control and Automation, 21-23 June 2006, DOI: 10.1109/WCICA.2006.1714134.

[14] F. Ye, Z. Zhang, K. Chakrabarty, and X. Gu, "Board-Level Functional Fault Diagnosis Using Multikernel Support Vector Machines and Incremental Learning," IEEE Tran Computer-Aided Design of Int. Circuits and Systems, Vol. 33, No. 2, Feb. 2014, pp. 279-290.

[15] M. Berahman, A.A. Safavi and M.R. Shahrbabaki, "Fault detection in Kerman combined cycle power plant boilers by means of support vector machine classifier algorithms and PCA," 3rd International Conference on Control, Instrumentation, and Automation (ICCIA), 28-30 Dec. 2013, DOI: 10.1109/ICCIAutom.2013.6912851.

[16] I. Samy, I.-S. Fan and S. Perinpanayagam, "Fault diagnosis of rolling element bearings using an EMRAN RBF neural network-demonstrated using real experimental data," Sixth International Conference on Natural Computation (ICNC), 10-12 Aug. 2010, DOI: 10.1109/ICNC.2010.5583833.

[17] B. Polok, P. Bilski, "Optimization of the neural RBF classifier for the diagnostics of electronic circuit," 15th IMEKO TC10 Workshop, 6-7 July, 2017, Budapest, Hungary, pp. 121-126

[18] Y. LeCun, J. Denker, and S. Solla, "Optimal brain damage,"

[19] G.N. Stenbakken, T.M. Souders, and G.W. Stewart, 1989, "Ambiguity groups and testability," IEEE Trans. Instr. and Meas., Vol. 38, Issue 5, 1989, pp. 941-947.

[20] N. Garcia-Pedrajas and C. Fyfe, "Evolving Output Codes for Multiclass Problems," IEEE Trans. Evol. Comp., Vol. 12, No. 1, Feb. 2008, pp. 93-106.

[21] S. Escalera, O. Pujol and P. Radeva, "On the Decoding Process in Ternary Error-Correcting Output Codes," IEEE Trans. Pattern Anal. and Machine Intel., Vol. 32, No. 1, Jan. 2010, pp. 120-134.

[22] P. Bilski, J. Wojciechowski," Automated Diagnostics of Analog Systems Using Fuzzy Logic Approach", IEEE Trans. Instr. and Meas., Dec. 2007, Vol. 56, Issue 6, pp. 2175-2185.

[23] M. Catelani and S. Giraldi, "A measurement system for fault detection and fault isolation of analog circuits," Measurement, Vol. 25, No. 2, 1999, pp. 115-122.

[24] M. Catelani, A. Fort, V. Grande, M. Mugnaini, and I. Trotta, "Automatic Fault Diagnosis System for a Gas Turbine using a Simulation before Test Approach," Proc. IEEE Instrumentation and Measurement Technology Conference (IMTC), 24-27 April 2006, Sorrento, Italy.

[25] L. Chruszczyk, J. Rutkowski, and D. Grzechca, "Optimal Excitation in Fault Diagnosis of Analog Electronic Circuits," Proc. 14th International Conf. Mixed Design of Integrated Circuits and Systems (MIXDES), 21-23 June 2007, Ciechocinek, Poland, DOI: 10.1109/MIXDES.2007.4286218.

[26] J. Bilski and J. Smoląg, "Parallel Architectures for Learning the RTRN and Elman Dynamic Neural Networks," IEEE Trans. Parallel and Distributed Systems, 2015, Vol. 26, No. 9, pp. 2561-2570.

[27] A. Bilski and J. Wojciechowski, "Automatic parametric fault detection in complex analog systems based on a method of minimum node selection," Int. J. Appl. Math. Comput. Sci., 2016, Vol. 26, No. 3, pp. 655–668.

Advances in NIPS2, Morgan Kaufman, San Mateo, 1900, pp. 598-605.