WOJCIECH TOCZEK, MICHAŁ KOWALEWSKI

Gdansk University of Technology
Faculty of Electronics Telecommunication and Informatics
Poland, e-mail: Wojciech.Toczek@eti.pg.gda.pl

A NEURAL NETWORK BASED SYSTEM FOR SOFT FAULT DIAGNOSIS IN ELECTRONIC CIRCUITS

The paper considers the architecture and the main steps of development of a neural network based system for diagnosis of soft faults in analog electronic circuits. The definition of faults of interest, selection of an optimal set of measurements, feature extraction, the construction of the artificial neural network, training and testing the network, are considered. A fault dictionary method was implemented in the system. Experimental results are presented on an example of diagnosis of a 6-th order bandpass filter. The measuring part of the system performs input-output measurements in the frequency domain with the aid of a HP 4192 Transmitance Analyzer.

Keywords: analog fault diagnosis, fault dictionary, artificial neural networks, soft-faults

1. INTRODUCTION

Automated diagnosis of analog electronic circuits provides an extremely difficult challenge. This is due to the problem of modelling of faulty circuits, the tolerance of components, the problem of a limited number of test points and the nonlinear nature of the relationship between the circuit responses and the component characteristics, even if the circuit is linear. Among the various techniques suggested in the literature, for example: rule-based expert systems, model-based methods, identification methods, in practical engineering applications the taxonometric approach that employs a fault dictionary is most widely appreciated. This approach requires higher initial computational cost but provides fast diagnosis at production time [11]. The fault dictionary is built at the before-test stage by means of simulations of the circuit under test (CUT), under nominal and faulty conditions with a pre-defined input stimulus. The simulated behaviour of a faulty CUT is usually called its fault signature. In the diagnostic process, the measured circuit response is compared with the signatures corresponding to each potential fault condition stored in the dictionary. The state of the circuit is then reported. In the recently published works [1-21] the most popular for creating the fault dictionary, memorizing and verifying it, is an Artificial Neural Network (ANN).

The development of a diagnostic system consists of the following main steps:

- definition of faults of interest,
- selection of an optimal set of measurements,
- feature extraction (generation of CUT signatures),
- the choice of type and topology of the ANN,
- training and testing the ANN.

By a fault in analog electronic circuit we mean any change in the element value which can cause a failure of the circuit. Faults can be categorized according to their effect on the circuit function. A fault that prevents the circuit to perform its function is called a hard fault. A fault that does not prevent the CUT to perform its function, but causes it to operate out of its specification range is called a soft fault. The terms soft and hard fault give an idea about how difficult a fault is to detect. Diagnosis of soft faults is a more challenging task then hard ones.

This paper discusses the possibility of using the ANN and dictionary approach for soft-faults location in electronic circuits at the component level. In the following section we describe the architecture of the developed diagnostic system. The organization of the rest of this paper is as follows: In section 3 the method for selection of the test frequencies is presented. Details of the applied feature extraction technique are given in Section 4. In Section 5 we present a radial basis neural network classifier. Experimental results are reported in Section 6.

2. ARCHITECTURE OF THE DIAGNOSTIC SYSTEM

The architecture of the diagnostic system developed by the authors is shown in Fig. 1. The measuring part of the system provides diagnostic information from the CUT input/output measurements performed in the frequency domain. A test signal is applied to the CUT and the response is analyzed by the HP 4192A LF Transmitance Analyzer. The measurements on a real circuit are sent from the analyzer, via the IEEE 488 interface bus, to the computer PC. The system is besides of measuring equipment, composed of a feature extractor and a classifier. The former is used to reduce the measurement data and extract diagnostically relevant features, the latter is used to localise a faulty element. We extract significant features (signatures) from the responses and use them as inputs to the neural network. The ANN, which has been previously trained to classify single faults on the CUT, reports the diagnosis.

At the before-test stage all effort is focused on the problem of modelling of the circuit under test, the extraction of effective features and training the neural network. The training patterns are collected from the faulty circuit responses using the MATLAB model of the CUT, Monte Carlo simulation and the same feature extraction technique that will be used further in the testing stage.

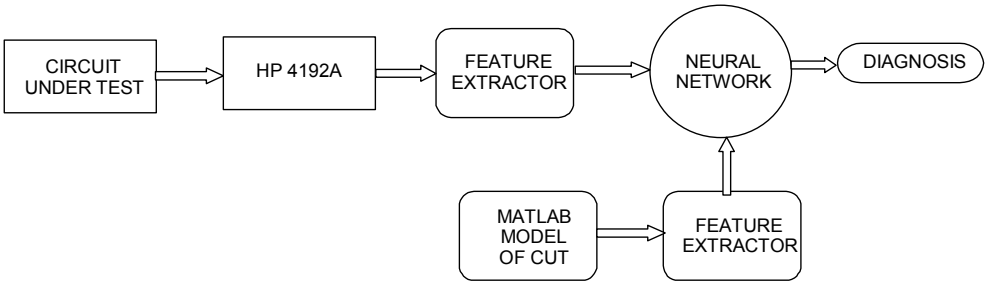


Fig. 1. The neural network-based diagnostic system architecture.

3. OPTIMIZING THE SET OF TEST FREQUENCIES

The 6-th order Deliyannis - Friend filter, shown in Fig. 2, was used as the circuit under test. The circuit is stagger - tuned bandpass filter with Butterworh response.

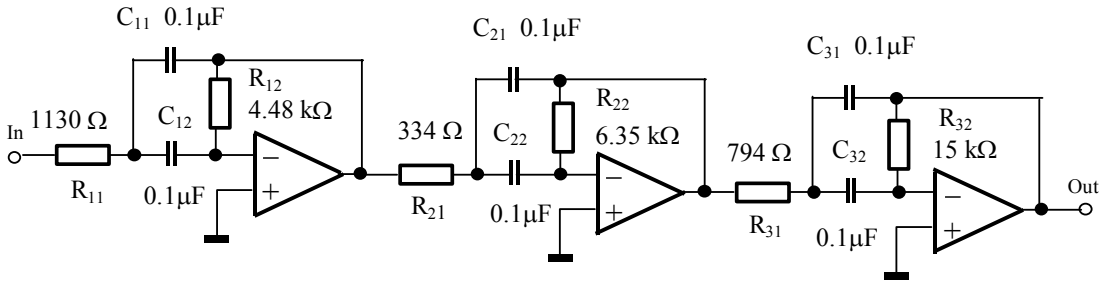


Fig. 2. The circuit under test.

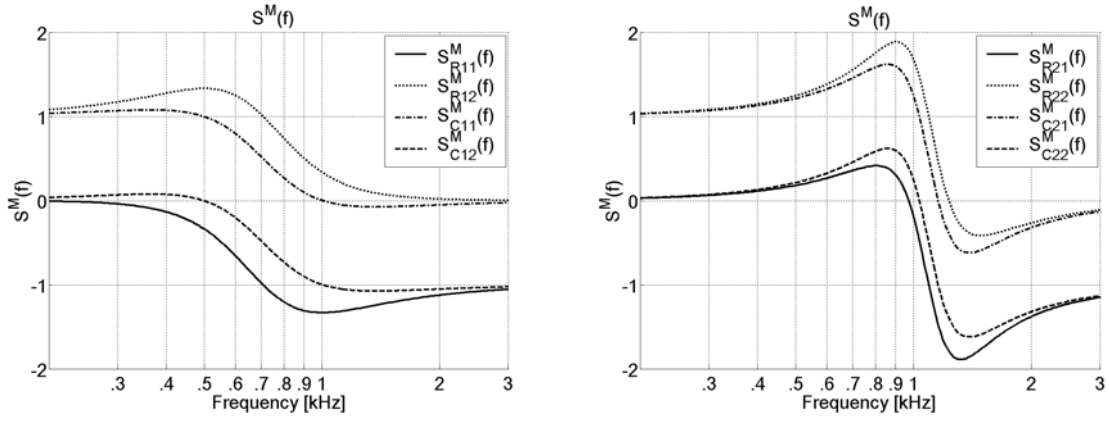
Signatures used to construction of a fault dictionary are obtained by applying a sinusoidal input signal with a variable frequency to the CUT and measuring the magnitude of the output signal. The number of measurement frequencies *m* should be kept to a minimum to reduce the cost of testing. The method of selecting test frequencies is based on differential sensitivity computation. Differential sensitivity reflects the change of the output parameter *P* due to an infinitesimal change of the element α. The following equation defines the differential sensitivity of parameter P_j with respect to component α_i

$$S_{\alpha_i}^{P_j} = \frac{\alpha_i}{P_j}\frac{\partial P_j}{\partial \alpha_i} = \left.\frac{\frac{\Delta P_j}{P_j}}{\frac{\Delta \alpha_i}{\alpha_i}}\right| \quad \Delta\alpha_i \to 0\,. \tag{1}$$

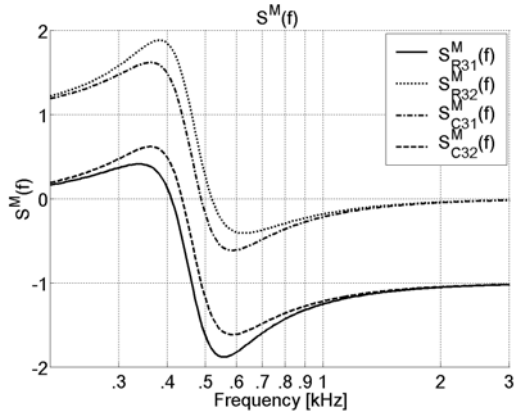
In our approach, parameter P_j corresponds to the *j*-th sample of the frequency response and element α_i is the *i*-th passive component of the CUT (R or C). Our goal is to find test frequencies for which a deviation of parameters gives an absolute maximum change in the magnitude of the transfer function. To this aim the differential sensitivity given in Eq. (1) must be expressed as a function of frequency

$$S_{\alpha_i}^{M}(f) = \frac{\alpha_i}{M(f)}\frac{\partial M(f)}{\partial \alpha_i}, \tag{2}$$

where *M* is the magnitude of the transfer function.



a) b)



c)

Fig. 3. Family of differential sensitivity curves for filter stages: a) I, b) II and c) III.

The filter under test has 3 stages connected in cascade and each stage is composed of 4 passive components described as follows: R_{k1}, R_{k2}, C_{k1}, C_{k2}, where k indicates the number of a stage. Hence for all circuit components there is a family of 12 differential sensitivity curves expressed as a function of frequency. These curves are shown, separately for each stage, in Fig. 3.

Selection of the optimum test set is based on determining frequencies for which sensitivity curves reach extreme values (maximum or minimum). The procedure is executed by solving the equation

$$\frac{\partial}{\partial f}\left(\frac{\alpha_i}{M(f)}\frac{\partial M(f)}{\partial \alpha_i}\right)=0 \tag{3}$$

for each passive component α_i of the CUT. The solutions of Eq. (3) for each component α_i give a vector $\mathbf{f}_{\alpha_i}$ of optimum test frequencies. Table 1 shows the rounded values of optimum frequencies obtained from Eq. (3).

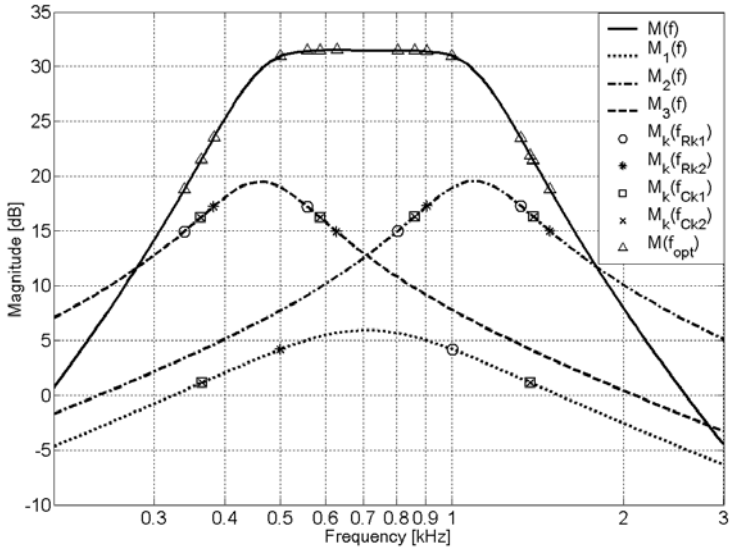


Fig. 4. Family of frequency responses for three stages: M1(f), M2(f), M3(f) and the overall filter response M(f). Marked points indicate magnitudes at frequencies obtained from Eq. (3).

Table 1. Optimum test frequencies

Stage 1		Stage 2		Stage 3	
Component α_i	Optimum frequencies $\mathbf{f}_{\alpha_i}$ [Hz]	Component α_i	Optimum frequencies $\mathbf{f}_{\alpha_i}$ [Hz]	Component α_i	Optimum frequencies $\mathbf{f}_{\alpha_i}$ [Hz]
R11	1001	R21	804, 1320	R31	339, 557
R12	500	R22	905, 1485	R32	382, 628
C11	364, 1374	C21	860, 1389	C31	362, 587
C12	364, 1374	C22	860, 1389	C32	362, 587

The results are shown graphically in Fig. 4 with the family of frequency responses for three stages: $M_1(f)$, $M_2(f)$, $M_3(f)$ and the overall filter response M(f). Optimum frequencies are marked with appropriate symbols on the frequency responses for each stage. As we see from Fig. 4 there are some similar and even recurrent frequency values. Hence a procedure for removing duplicated values and obtaining averaged of closely placed frequencies (difference ≤ 2Hz) was adopted. Finally a vector $\mathbf{f}_{opt}$ = [339, 363, 382, 500, 557, 587, 628, 804, 860, 905, 1001, 1320, 1374, 1389, 1485] Hz of 15 test frequencies was obtained and used further in the construction of a fault dictionary. These frequencies are marked by triangles in Fig. 4.

4. FEATURE EXTRACTION TECHNIQUE

The selection of the features that are going to be used in ANN training is an important step at the before-test stage. The importance of such a task stems from the fact that it significantly reduces the neural network's size and improves its performance.

Each circuit response is composed of samples of magnitude, taken at *m* test frequencies collected in vector $\mathbf{f}_{opt}$. Since the dimension of responses is large (15 elements in vector) we need to extract relevant diagnostic features in compressed form referred to as a fault signature. The goal of the feature extraction technique is to reduce the number of elements in the signature with acceptable loss of information. There are several feature extraction techniques reported in the literature: Principal Component Analysis (PCA), Fourier Transforms, Wavelets. The PCA technique, also known as the Karhuen-Loeve transform, was chosen. Its main advantage is effective data compression.

Let $\mathbf{x} = [x_1, ..., x_m]^T$ be an $m \times 1$ random vector of samples of a frequency response. The $m \times m$ covariance matrix of $\mathbf{x}$ is

$$\mathbf{C} = E\left[(\mathbf{x} - E(\mathbf{x}))(\mathbf{x} - E(\mathbf{x}))^T\right], \tag{4}$$

where E[·] is the expectation operator.

In practice, estimators of expected value $\hat{\mu}$ and covariance matrix $\hat{\mathbf{C}}$ are calculated on the basis of some data set. Let $\mathbf{X}$ be an $m \times n$ data matrix of n frequency responses sampled at *m* points. Points can be described by a vector containing the mean value of each row of the data matrix

$$\hat{\mu} = [\bar{\mathbf{x}}_1, \bar{\mathbf{x}}_2, ..., \bar{\mathbf{x}}_m]^T \ . \tag{5}$$

To estimate covariance matrix, the data matrix is first zero-centered. It means that each vector of data is shifted across the origin, so that the mean of the samples at each test frequency is zero. The $m \times m$ estimate of covariance matrix of $\mathbf{X}$ is

$$\hat{\mathbf{C}} = \frac{1}{\mathrm{n}-1}\left[\left(\mathbf{X}-\hat{\mathbf{X}}_{\mu}\right)\left(\mathbf{X}-\hat{\mathbf{X}}_{\mu}\right)^{\mathrm{T}}\right], \tag{6}$$

where matrix $\hat{\mathbf{X}}_{\mu}$ is composed of replicated and tiled vector $\hat{\mu}$.

From a geometrical point of view, any covariance matrix is associated to a hyper-ellipsoid in the *m* dimensional space. PCA corresponds to rotation of coordinates in the way that gives the associated hyper-ellipsoid in its canonical form. The novel coordinate basis is coincident with the hyper-ellipsoid principal axis. In general, the PCA technique transforms *n* vectors ($\mathbf{x}_1$, $\mathbf{x}_2$, ..., $\mathbf{x}_n$) from a *m*-dimensional space to *n* vectors ($\mathbf{y}_1$, $\mathbf{y}_2$, ..., $\mathbf{y}_n$) in a new, *r* - dimensional space of reduced dimensionality as

$$\mathbf{y}_i = \sum_{j=1}^{r} a_{k,i}\mathbf{e}_k, \tag{7}$$

where: $a_{j,i}$ are the projections of the original vectors $\mathbf{x}_i$ on the eigenvectors $\mathbf{e}_k$, corresponding to the *r* largest eigenvalues λ_i^2 of the covariance matrix for the original data set. These projections are called the principal components of the original data set. The value of $\lambda_i^2 / \sum \lambda_i^2$ gives the proportion of variation explained by the *i*-th principal component. To reduce the dimension of the new data set, $r<<n$ should be true. The choice of the number of dimensions required is usually based on the amount of variation accounted for by the leading principal components.

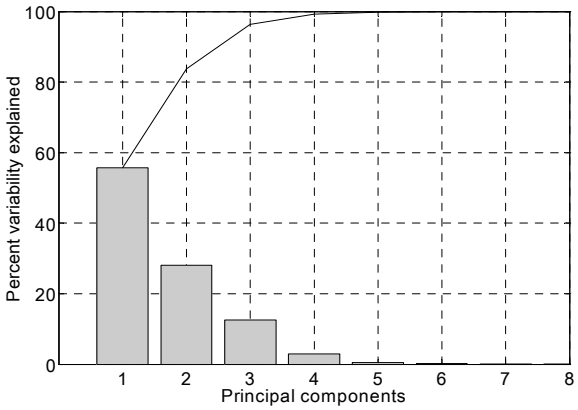


Fig. 5. A pareto plot of the percent variability explained by each principal component.

For example we can see from a pareto plot (Fig. 5) that the first four principal components explain roughly 99% of the total variability in the frequency responses. Those principal components which contribute only a small amount to the total variance in the data set can be eliminated.

The PCA has three effects: it orthogonalizes the components of original vectors so that they are uncorrelated with each other, it orders the components according to the magnitude of their variance and it eliminates those components that contribute the least to the variation in the data.

The MATLAB routine *prepca* uses singular value decomposition to compute the principal components. It is assumed that the input data set has already been autoscaled so that it has a zero mean and unitary variance. The function *prestd* can be used to autoscale the data.

To obtain a transformed data matrix the input matrix **X** is multiplied by a transformation matrix whose rows consist of the eigenvectors of the covariance matrix

$$\mathbf{Y} = \mathbf{\Phi X}\,, \tag{8}$$

where: **Y** is an $r \times n$ transformed data matrix, **Φ** is an $r \times m$ transformation matrix.

5. RADIAL BASIS FUNCTION NETWORK CLASSIFIER

A radial basis function network classifier was used to perform classification of signatures. In this paper the variant of radial basis function network called generalized regression neural network (GRNN) was chosen, because of its predestination to normalize the output vectors of the network.

The GRNN is a two layer radial basis function network with hidden layer neurons characterized by a radial basis function defined as

$$\mathrm{f}(\mathbf{y}) = \exp\left(-\frac{\|\mathbf{y}-\mathbf{w}\|^2}{2\sigma^2}\right), \tag{9}$$

where **y** is the input vector, **w** is the position in the input space (center) of the radial basis function, and σ is the corresponding scaling factor characterizing the area of the activation region. The second layer adds these contributions for each class of inputs and normalizes by the sum of the input vector elements. This normalization causes that the sum of all elements of the output vector for any input signature equals one.

The GRNN can be used for classification problems in the following manner. When an input signature is presented, the first layer computes distances from the input vector **y** to the training input vectors **w** (radial basis function centers) and produces a vector whose elements indicate how close the input is to a training input. A value of an *j*-th output layer neuron is a normalized sum of the radial basis function outputs which correspond to the *j*-th fault class. As the *j*-th value grows there is more probability that the input signature belongs to the *j*-th fault class. To detect a fault class, a *compete* transfer function is applied to pick the maximum of these probabilities and produce a '1' for that class and a '0' for the other classes.

Construction of GRNN for the classification problem is straightforward. In the first step, an unsupervised technique consisting of placing the hidden layer neuron centers on the centroids of the training input clusters is used. In this work the fuzzy C-Mean clustering algorithm was used to find cluster centroids. In the second step, the output layer weights are set to the matrix **T** of target vectors. **T** is the $cn \times hn$ matrix where *cn* is the number of classes and *hn* is the number of hidden layer neurons. Each column vector of matrix **T** is associated with one hidden layer neuron and has '1' only in the row associated with the class number to which that neuron belongs.

6. EXPERIMENTAL RESULTS

Some experiments were performed with GRNN in order to detect single parametric faults at the component level of the CUT. Non-faulty components are allowed to vary within their specified tolerance range. It was assumed that the parameter of the non-faulty component is a random value described by normal distribution $\mathrm{N}(\mu, \sigma)$ with the following parameters

$$\mu = \alpha_{i\,nom} \quad , \qquad \sigma = \frac{tol.\alpha_i}{3} \quad , \tag{10}$$

where $\alpha_{i\ \mathrm{nom}}$ is the nominal value of component parameter α_i, and $tol.\ \alpha_i$ is the tolerance of element α_i (1% for resistors and 2% for capacitors).

A parametric fault of any component is caused by exceeding its value outside the tolerance range. Moreover the value of the component may decrease or increase. Hence one class of faults (associated to one component) consists of two clusters. First a cluster is made up of Monte Carlo simulations for faulty component α_i varying within a specified range

$$0.5\alpha_{i\,nom} < \alpha_i < 0.95\alpha_{i\,nom} \,. \tag{11}$$

The second cluster is associated with the values greater than nominal and is specified with the following formula

$$1.05\alpha_{i\,nom} < \alpha_i < 1.5\alpha_{i\,nom} \,. \tag{12}$$

The 125 000 runs of the Monte Carlo simulation were performed to form 25 clusters (2 clusters for every component plus one describing a fault-free condition). Hence each cluster consists of 5000 points (Fig. 6). The fault-free cluster is placed in the middle of the figure. All signatures contained in that cluster are located nearby the origin of the principal component space. Clusters obtained on the basis of relation (11) are grouped above the fault-free cluster (the second principal component is greater than -0.3). Clusters which had been obtained on the basis of relation (12) are grouped below the fault-free cluster. For those clusters the second principal component is lower than -0.3.

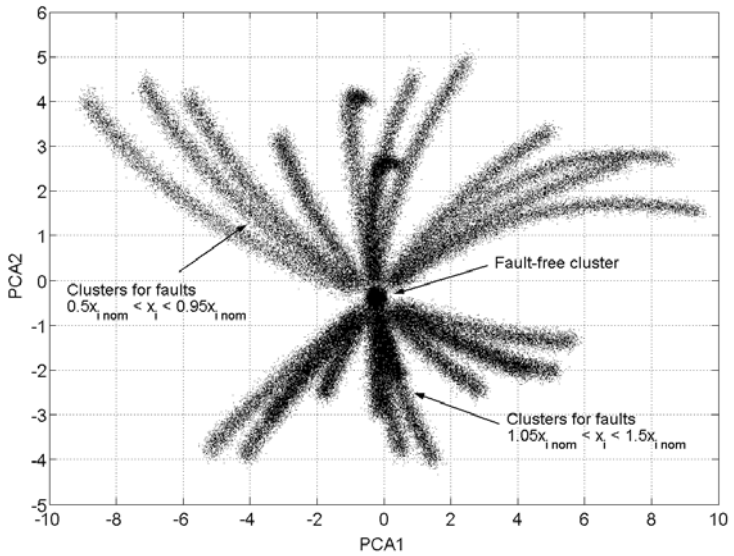


Fig. 6. The 25 clusters showed in the two first principal components space.

All signatures were then equally divided into two subsets: training and testing. For every cluster in the training subset the fuzzy C-Mean clustering algorithm was used to find 20 cluster centroids. Thus, 500 column vectors were obtained which define the coordinates of the radial basis function centers in the four-dimensional space of the first four principal components. Centers are shown in the three dimension principal component space in Fig. 7.

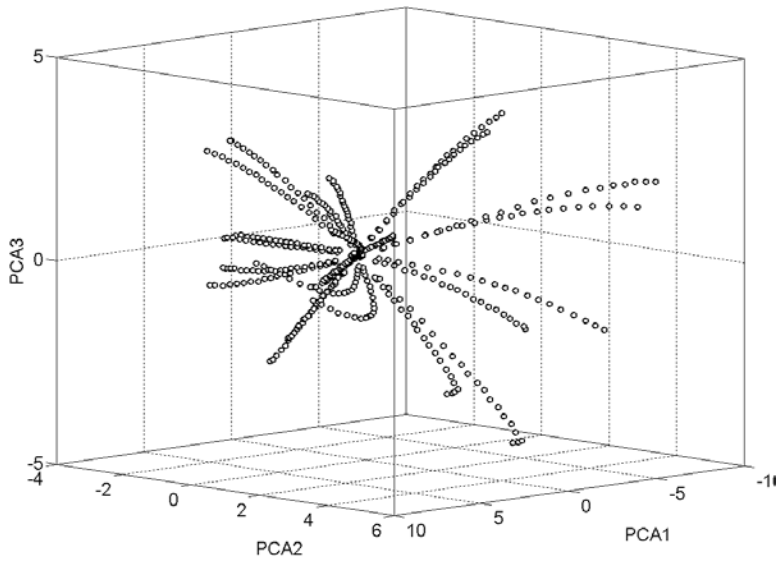


Fig. 7. Placement of the 500 radial basis function centers in the three dimension principal components space.

All clusters form 13 classes (one fault-free and 12 for faulty components). The assignment of radial basis function outputs to the GRNN output vector of probabilities is executed by the MATLAB routine *normprod*.

$$\mathbf{p} = \mathbf{T} \cdot \mathbf{h} \cdot \sum_{j=1}^{cn} \frac{1}{p_i}, \tag{13}$$

where **T** is the 13 × 500 second layer weights matrix, **h** is the 500 × 1 vector of hidden layer outputs and **p** is the 13 × 1 vector of probabilities. The number of class for each input signature is determined by the row number of the largest element p_1 in vector **p**.

The information about a number of classes indicated by the network classifier for a given signature is usually not sufficient to estimate the probability of appearance of a fault. There is a need to introduce a confidence or uncertainty measure on the output of the network which will indicate the level of confidence or uncertainty that the network associates with its answer [11]. We have applied an uncertainty measure *mu* described as follows

$$mu = 1 - (p_1 - p_2), \tag{14}$$

where p_1 and p_2 are the values of the largest and the next closest element of the output vector, respectively. This measure ranges from 0 to 1, where 0 means the smallest uncertainty and 1 means the highest uncertainty. As the uncertainty measure *mu* decreases, the input signature comes closer to one of the radial basis function centers and the output value of that neuron outstrips the output values of other hidden layer neurons.

The uncertainty level *ul* can be chosen arbitrarily from the range 0 to 1. This assumed level is compared with the uncertainty measure *mu* for a given output vector of the GRNN. If the relation

$$mu < ul \tag{15}$$

is fulfilled then the maximum value of the output vector indicates the fault class number. Otherwise the input signature is not classified. This informs that there are at least two classes to which the input signature could belong. Such a situation is usually caused by fault ambiguity when it is impossible to distinguish the source of the error from observation at the

output. Fault ambiguity can be resolved in two ways. One is to join two or more components into one class. Additional observation points could also be added, which will increase the fault detectability.Results obtained with GRNN on the testing subset for four different uncertainty levels are scheduled in Table 2. All input signatures for which uncertainty *mu* is smaller than the assumed uncertainty *ul*, form a yield subset. The term *yield level* is defined as the ratio of the number of signatures contained in the yield subset to the total number of 62500 signatures in the testing subset of the fault dictionary. The *ambiguous classification level* is defined as (100 – *yield level*) %. The term *classification error* refers to the percentage of signatures incorrectly classified by the network with respect to the entire yield subset.

Table 2. Signature classification results.

Uncertainty level *ul*	Number of signatures in the yield subset	Yield level [%]	Number of incorrectly classified signatures	Classification error [%]
1	62500	100	2839	4.54
0.1	55898	89.4	677	1.21
0.01	51829	82.9	199	0.38
0.001	48790	78	59	0.12

This experiment was repeated for the uncertainty level varying continuously from 0.001 to 1. Results including a plot of ambiguous classification level and classification errors versus uncertainty level are shown in Fig. 8. One can see that the assumption of a low uncertainty level significantly decreases the classification error. The greatest decrease of classification error occurs when the uncertainty level is close to one. Choosing an uncertainty level less than 0.01 does not reduce the classification error noticeably, but the ambiguous classification level still grows. Hence, the optimum interval for the uncertainty level should be chosen in the range of 0.01 to 0.1. This is a compromise between the number of misclassified signatures and the classification error.

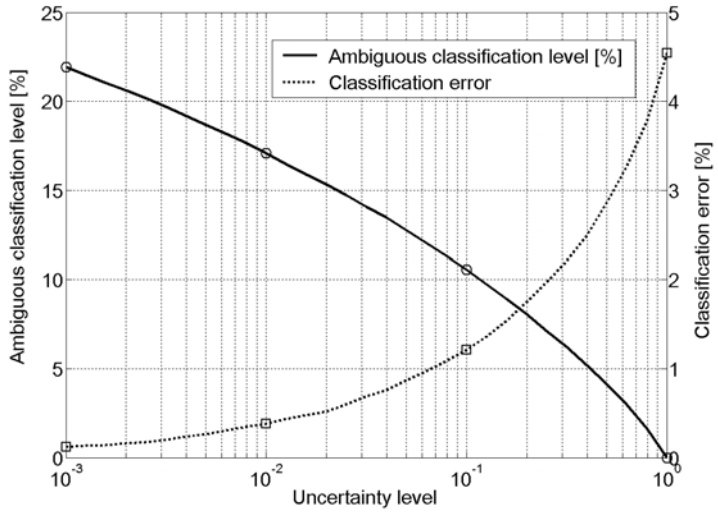


Fig. 8. The ambiguous classification level and classification error versus uncertainty level for GRNN.

7. CONCLUSIONS

The main steps of development fault dictionary system based on a neural network, for a diagnosis of soft faults in analog electronic circuits have been presented on the example of a 6-th order bandpass filter. It was shown that the diagnosis can be performed using a set of

systematically chosen 15 test frequencies. The problem of data compression from 15 to 4 dimensions was solved with the aid of Principal Component Analysis. A radial basis function network classifier was used to perform classification of soft fault signatures. The introduced measure of diagnosis uncertainty permits control of the classification error and assures a compromise between the number of misclassified signatures and the classification error. A dictionary-based diagnosis can provide a result very quickly, because the fault simulation has been done ahead of the testing time.

REFERENCES

1. Stopjakova V., Malosek P., Micusik D., Matej M., Margala M.: *Classification of Defective Analog Integrated Circuits Using Artificial Neural Networks*, Journal of Electronic Testing: Theory and Applications, vol. 20, 2004, pp. 25-37.
2. Malhi A., Gao R.X.: *PCA-Based Feature Selection Scheme for Machine Defect Classification*, IEEE Trans. Instrum. Meas., vol. 53, no. 6, 2004, pp. 1517-1525.
3. He Y.G., Tan Y.H., Sun Y.: *A Neural Network Approach For Fault Diagnosis Of Large-scale Analogue Circuits*, ISCAS'02, vol. I, 2002, pp. 153-156.
4. Alippi C., Catelani M., Fort A.: *Soft Fault Diagnosis In Analog Electronic Circuits: Sensitivity Analysis By Randomized Algorithms*, IEEE Instrum. and Meas Technology Conference, Budapest, 2001, pp. 1592-1595.
5. Catelani M., Fort A., Singuaroli R.: *Hard fault diagnosis in electronic analog circuits with radial basis function networks*, Proc. XVI IMEKO World Congress, Vienna, 2000, pp. 167-170.
6. Catelani M., Fort A., Bigi M., Singuaroli R.: *Soft fault diagnosis in analogue circuits: A comparison of fuzzy approach with radial basis function networks*, IEEE Instr. and Meas. Technology Conference, Baltimore, 2000, pp. 1493-1498.
7. Catelani M., Fort A.: *Soft Fault Detection and Isolation in Analog Circuits: Some Results and a Comparison Between a Fuzzy Approach and Radial Basis Function Networks*, IEEE Transactions on Instrumentation and Measurement, vol.51, no. 2, 2002, pp. 196-202.
8. Fanni A., Giua A., Marchesi M., Montisci A.: *A Neural Network Diagnosis Approach for Analog Circuits*, Applied Intelligence, vol. 2, 1999, pp. 169-186.
9. Rutkowski J.: *A two stage neural network DC fault dictionary*, ISCAS'94, 1994, pp. 299-302.
10. Rutkowski J., Machniewski J.: *Sensitivity Based Analog Fault Dictionary*, ECCTD'99, 1999, pp. 599-602.
11. Spina R., Upadhyaya S.: *Linear Circuit Fault Diagnosis Using Neromorphic Analyzers*, IEEE Transactions on Circuits and Systems - II: Analog and Digital Signal Processing, vol. 44, no. 3, 1997, pp. 188-196.
12. Materka A., Strzelecki M.: *Parametric Testing of Mixed-Signal Circuits by ANN Processing of Transient Responses*, Journal of Electronic Testing: Theory and Applications, 9, 1996, pp. 187-202.
13. Barschdorff D.: *Neural networks and fuzzy logic: new concepts for technical failure diagnosis?*, Proc. of the XIII IMEKO World Congress, Torino, 1994, pp. 2430-2437.
14. Osowski S., Herault J., Demartines P.: *Kohonen map applied to fault location in analogue circuits*, XVII-th National Conference Circuit Theory and Electronic Networks, Wrocław, 1994, pp. 585-590.
15. Kowalewski M., Toczek W.: *Diagnostic system for fault location with dictionary method*, Krajowa Konferencja Elektroniki, Koszalin, 2002, pp. 235-240. (in Polish)
16. Toczek W.: *Diagnosis of analog electronic circuits with the aid of histogram-based fault dictionary* (in Polish), Elektronizacja, 10, 2001, pp. 22-25.
17. Hora C., Segers R., Eichenberger S., Lousberg M.: *On a Statistical Fault Diagnosis Approach Enabling Fast Yield Ramp-Up*, Journal of Electronic Testing: Theory and Applications vol. 19, 2003, pp. 369-372.
18. Saab K., Hamida N. B., Kaminska B.: *Closing the Gap Between Analog and Digital Testing*, IEEE Transaction on Computer-Aided Design of Integrated Circuits and Systems, vol.20, no. 2/2001, pp.307-314.
19. Somayajula S., Sanchez-Sinencio E., Pineda de Gyvez J.: *Analog Fault Diagnosis Based on Ramping Power Supply Current Signature Clusters*, IEEE Transaction on Circuits and Systems-II: Analog and Digital Signal Processing, vol. 43, no. 10, 1996, pp. 703-712.
20. Bernieri A., Betta G., Liguori C.: *On-line Fault Detection and Diagnosis Obtained by Implementing Neural Algorithms on a Digital Signal Processor*, IEEE Transaction on Instrumentation and Measurement, vol. 45, no. 5, 1996, pp. 894-899.
21. Rutkowski J. : *Słownikowe metody diagnostyczne analogowych układów elektronicznych* Wydawnictwa Komunikacji i Łączności, Warszawa 2003.

SYSTEM DO DIAGNOSTYKI USZKODZEŃ PARAMETRYCZNYCH W UKŁADACH ELEKTRONICZNYCH Z WYKORZYSTANIEM SIECI NEURONOWEJ

Streszczenie

W artykule przedstawiono system do diagnostyki uszkodzeń parametrycznych w układach elektronicznych. W systemie zaimplementowano słownikową metodę lokalizacji uszkodzeń, bazującą na pomiarach w dziedzinie częstotliwości przeprowadzanych za pomocą analizatora transmitancji HP4192A. Rozważono główne etapy projektowania systemu: definiowanie modelu uszkodzeń, wybór optymalnych częstotliwości pomiarowych, ekstrakcję cech diagnostycznych, konstrukcję sieci neuronowej oraz trening i testowanie sieci.

Główną cechą prezentowanego podejścia jest zastosowanie słownikowej metody lokalizacji uszkodzeń do uszkodzeń parametrycznych. Rozpatrywane są pojedyncze uszkodzenia parametryczne elementów dyskretnych.

Przedstawiono metodę optymalizacji częstotliwości pomiarowych na podstawie analizy wrażliwościowej charakterystyki amplitudowej względem wartości parametrów elementów. Selekcja częstotliwości bazuje na ocenie ekstremalnych wartości charakterystyk wrażliwościowych.

Przestrzeń danych pomiarowych zredukowano do czterech wymiarów za pomocą analizy składowych głównych (PCA). Przekształcenie PCA ortogonalizuje elementy oryginalnych wektorów danych, porządkuje według wielkości ich wariancji i eliminuje składowe, które mają najmniejszy wkład w wyjaśnienie zmienności danych. Dane transformowane do przestrzeni o zredukowanej liczbie wymiarów służą jako dane wejściowe dla klasyfikatora.

Do klasyfikacji sygnatur zastosowano sieć neuronową typu GRNN z radialnymi funkcjami bazowymi w warstwie ukrytej. Klasteryzację danych przeprowadzono za pomocą algorytmu Fuzzy C-Mean. Centra radialnych funkcji bazowych warstwy ukrytej nałożono na centroidy poszczególnych klastrów. Zadaniem warstwy wyjściowej jest przyporządkowanie neuronów radialnych do poszczególnych klas uszkodzeń. Sieć dokonuje klasyfikacji sygnatury uszkodzenia otrzymywanej z pomiarów poprzez wskazanie najbardziej prawdopodobnego uszkodzenia. Zastosowany typ sieci wytwarza znormalizowane odpowiedzi umożliwiające wyznaczenie poziomu niepewności wyniku diagnozy.