The On-line Evolutionary Method for Soft Fault Diagnosis in Diode-transistor Circuits

Marek Korzybski and Marek Ossowski

Abstract—The paper is devoted to diagnostic method enabling us to perform all the three levels of fault investigations - detection, localization and identification. It is designed for analog diode-transistor circuits, in which the circuit's state is defined by the DC sources' values causing elements operating points and the harmonic components with small amplitudes being calculated in accordance with small-signal circuit analysis rules. Gene expression programming (GEP), differential evolution (DE) and genetic algorithms (GA) are a mathematical background of the proposed algorithms. Time consumed by diagnostic process rises rapidly with the increasing number of possible faulty circuit elements in case of using any of mentioned algorithms. The conncept of using two different circuit models with partly different elements allows us to decrease a number of possibly faulty elements in each circuit because some of possibly faulty elements are absent in one of two investigated circuits.

Keywords—electric circuit diagnosis, soft faults, multiple faults, evolutionary computation, gene expression programming, genetic algorithm, differential evolution

I. INTRODUCTION

FROM the beginning of the last two decades of the twentieth century a great progress of fault diagnosis methods for analog electronic circuits has been observed [1],[2],[3],[4],[5]. Nevertheless, no universal, effective and fast diagnostic method has been developed. The problem of detection, location and identification of soft and catastrophic faults is still present [6],[7],[8],[9],[10]. One of the fundamental multiple fault diagnosis problems is the choice of optimal measurement points sets ensuring proper diagnostic process. A measurement point means the set consisting of a measurement quantity, a measurement node or branch and all measurement conditions. Measurement conditions, which do not require supplementary excitations except for usual working, are very useful. Such measurements enable us to perform a successful fault diagnosis without breaking the usual work of a circuit. The method for diode-transistor circuits with DC and also small AC signals is developed in this paper. It is possible to measure two different values at each node accessible for measurements during the usual work of a circuit. These are DC components of node voltages, branch or source currents and also AC components. The method doubles the number of measurement points in a circuit under test. The idea of two independent circuit analyses, DC circuit and the circuit for small signals, is developed in [11],[12]. Presented method is a two-stage process. Each of these stages is performed for

M. Korzybski and M. Ossowski are with the Department of Electrical, Electronic, Computer and Control Engineering, Łódź University of Technology, Stefanowskiego 18/22, 90-924 Łódź, Poland (e-mail: marek.korzybski@p.lodz.pl).
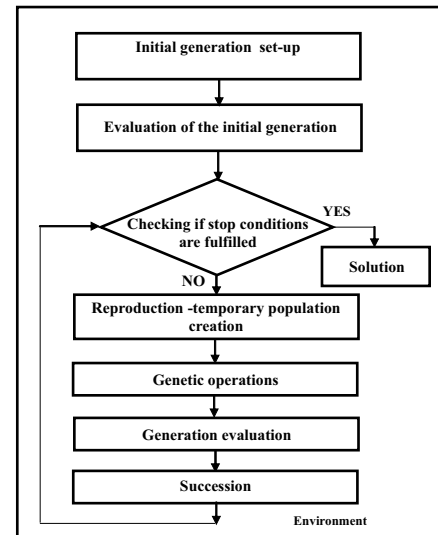
Fig. 1. Evolutionary algorithm scheme

a different measurement schema and uses different heuristic algorithm. These algorithms are evolutionary methods: gene expression programming, genetic algorithm and differential evolution. Unfortunately, the computation time of all of them increases rapidly along with the number of potentially defective elements. A method to compensate for that is to carry out the diagnostic process in two or more different systems, first in DC circuits, and then in the small-signal one, which allows a reduction of potentially faulty elements in each of them. The following parts of this paper consist of a short description of evolutionary algorithms in section II. Section III explains the method of performing both parts of the diagnostic process using GEP and GA with a sample circuit illustrating the effectiveness of the described algorithm. Section IV describes the two-stage diagnostic process using GEP and DE with an example illustrating its effectiveness. Section V contains comments and conclusions.

II. BASIS OF THE EVOLUTIONARY COMPUTING METHODS

Many algorithms elaborated during last few decades, which effectiveness is only experimentally but not mathematically confirmed, are defined as evolutionary methods (or techniques). Classified as heuristic, these algorithms are dedicated to solve optimization problems, where effectiveness is measured by time consumed by the process and size of the used computer memory. Both mentioned quantities should not rise

too rapidly when the problem being solved becomes larger. Heuristic methods commonly have multilevel structure. In that case, higher level algorithms are controlling algorithms of lower levels. Hence, these algorithms are usually defined as metaheuristics. Working principle of heuristic methods is defined as oriented penetration of solution space. In the proposed diagnostic algorithm we use the methods, in which the set of potential solutions, not only single one, is being processed. These techniques are defined as population heuristics. The prototype of these algorithms was the process of natural evolution leading to a better and better adaptation of subsequent generations to varying external requirements. The above considerations justify the use of following names for described methods: evolutionary algorithms (EA) and optionally evolutionary computing (EC). Evolutionary algorithm realizes the solution finding process by transforming populations (defined also as generations), meaning the sets of individuals being trial solutions (proposal of solutions) of the problem which specificity defines the environment (or working space) of the algorithm. Every member of the population (individual) is determined by the collection of features called phenotype. Its coded form is a genotype defined also as genome. Genotype's elements are chromosomes (the most common is monochronic genotype), which consist of elementary units called genes. Evaluation of individuals usefulness as a solution to the problem is the value of its fitness (adaptation) calculated with the use of genotype and fitness function defined by environment features. Evolutionary algorithm scheme is presented in Fig.1. After the initial population is created, the estimation of this generation adaptation is made and stop conditions are verified. If the process is not stopped, the reproduction, that is a preliminary selection, is performed. Its purpose is to create a transient generation, also called descendant population which shall be subjected to genetic operations. Next, the fitness function values for all individuals, being the evaluation of their adaptation, are calculated. Results of this evaluation are the base for making succession, that means the final selection. As a result, the new base (fundamental) generation is created. Next, checking stop conditions is performed again and the loop of algorithm is repeated if none of them is fulfilled.

A. Simple Genetic Algorithm

The most important evolutionary algorithm, genetic algorithm, appeared in the 60s of the last century [13],[14],[15],[16]. John Holland has applied the principle known from the theory of evolution to shape the population of individuals with binary coded properties. This gave rise to the use of methods known as genetic algorithms. These methods have been developed and widely distributed by the David Goldbergs later works. The most primary and the simplest version of this algorithm, referred to as a simple genetic algorithm (SGA) is briefly presented below. In SGA base population P, formed of N randomly generated individuals $x_i, (i = 1, ...N)$, where N is the population size, is subjected to the simulated evolution process. They form the initial population P^o. Since SGA uses binary encoding, the generation of a genotype of each of the N individuals of the initial population consists of performing the series of n draws (random selections). The main loop of the algorithm starts with an assessment of the population, which is the designation of values for the adaptation function for each individual. The adaptation (fitness) function reflects the usefulness of the individuals phenotype in solving the given problem. Then the stop criteria are performed, checking whether the test population contains a solution of the analysed problem. If the stop criteria are not met, the algorithms processing continues. A selection is carried out, involving a random choice of individuals suitable for reproduction. The SGA uses proportional selection, where the draw takes place with probability proportional to the calculated value of the fitness function of each individual. Draw with replacement is used here. Size of the group, referred to as the transient generation T, is the same as the SGA group, from which the drawing took place. Proportional selection means that individuals with the lowest adaptation values are unlikely to be found in the selected group, while those for which adaptation reaches the highest values may be subjected to multiple replication. The created transient population T is subjected to an action of genetic operators: crossing and mutation. Crossing is performed on pairs of randomly selected individuals with a probability of crossing predetermined for this process. Crossing is the exchange of parts of genotypes between individuals. The process is often called recombination, as the name clearly reflects the nature of the operation, which is mixing of genetic material of crossed individuals. After completing the crossing, a mutation is performed. The decision of its execution is performed randomly for each individual with a probability defined before starting the optimization. Gene, for which a positive decision has been taken, is subjected to the change of values for the opposite value, ie. from 1 to 0 and vice versa. Upon completion of crossing and mutation processes of individuals from transition generation T the creation of base population from all individuals of that generation is performed. In SGA the old generation is entirely replaced by the new one. Newly established population is evaluated and the process of creating the next generation is repeated if the stop conditions are not met.

B. Gene Expression Programming

The method, referred to as gene expression programming, invented at the end of the last century, is still a novel evolutionary genetic method which develops genetic programming. Fundamentals and practical implementation details of this technique were published by Candida Ferreira in 2001 [17],[18],[19],[20]. The main difference between GAs (genetic algorithms) and GEP (gene expression programming) is the nature of individuals forming generations. In both methods, individuals' representations (chromosomes) are sequences of symbols of defined length. However, in opposite to GAs, where phenotype of the population member is specific solution (some number, series of symbols defining a set etc.), chromosome in GEP defines a function of variables, which values set to the formula described by chromosome define function values in the interval, for which the function was created. Working principles of GEP do not deviate from general

scheme of evolutionary algorithm described at the beginning of the current chapter. GEP begins with the creation of initial population, which shall be evaluated. After verification by stopping procedures, the generation is subjected to a selection process and other genetic operators in numbers much greater than used for genetic algorithms discussed earlier. The gene construction in GEP should ensure that syntactically correct structures are created during the running of evolutionary operators. As the result of these operations, the new population is defined. Next, the algorithm loop, starting from every individual evaluation, is repeated the required number of times.

C. Differential Evolution

The first article presenting principles of the DE (differential evolution) was published by Rainer Storn and Kenneth Price in march 1995 [21]. Similarly to the other evolutionary algorithms, DE method is solving optimization problem by directional searching in the solution space, based on population processing. Differential evolution operates on vectors, which components define solution phenotype. The novel element, distinctive from other methods, is the way to create members of new generation. Candidate individuals for new population are created by adding weighted difference of two randomly selected vectors, to the third one, also drawn.

Scheme of the algorithm

1) initial population creation - random choice with uniform probability distribution of N vectors $\mathbf{x}_i$ $(i = 1, 2, ..., N)$ belonging to the solutions space, where:

$$\mathbf{x}_i = [x_{i,1}, x_{i,2}, ..., x_{i,m}] \tag{1}$$

2) setting-up mutated vector $\mathbf{v}_i$ for every individual $\mathbf{x}_i$ of the population, according to the formula:

$$\mathbf{v}_i = \mathbf{x}_{r_1} + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) \quad F > 0 \tag{2}$$

where

$$\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_2 \in [1, ..., m] \quad \mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_2 \neq i \tag{3}$$

are randomly chosen from defined above interval

3) creation of trial vector $\mathbf{u}_i$, (for every vector $\mathbf{x}_i$) which components are calculated according to the formula:

$$u_{i,j} = \begin{cases} v_{i,j} \ if \ R_{i,j}(0,1) < CR \ or \ j = j_R) \\ x_{i,j} \ elsewhere \end{cases} \tag{4}$$

where:

- $R_{i,j}(0,1)$ - is randomly chosen, for all the vector positions, constant value belonging to the interval $(0,1)$,
- CR - set-up earlier value defining probability of the acceptance gene of vector $\mathbf{v}_i$ inside vector $\mathbf{u}_i$,
- j_R - randomly chosen position in vector $\mathbf{u}_i$, where the component of vector $\mathbf{v}_i$ should be obligatory placed.

4) choice of new population vector - one of two: $\mathbf{x}_i$ or $\mathbf{u}_i$, with better fitness is selected.

It tends to be used another scheme of DE, differ from proposed before in the way of mutated vector $\mathbf{v}_i$ creation. Optional version of this process is as follows:

$$\mathbf{v}_i = \mathbf{x}_{r_1} + \lambda \cdot (\mathbf{x}_{best} - \mathbf{x}_{r_3}) + F \cdot (\mathbf{x}_{r_2} - \mathbf{x}_{r_3}) \tag{5}$$

where

$$\mathbf{r}_1, \mathbf{r}_2, \mathbf{r}_3 \in [1, ..., m] \quad \lambda, F > 0 \tag{6}$$

III. Diagnostic Use of GEP and AG Methods

The algorithm diagnoses single and multiple soft faults of diode-transistor circuits stimulated by signals having DC components and harmonic components with small amplitudes. It is possible to create for each of these circuits a diagram for the DC analysis and also the diagram for small harmonic signal analysis. In all probability a part of possible faulty elements belongs only to a small signal circuit, and it is absent from a DC circuit. A diagnostic process of a DC circuit can be quickly performed as for an original circuit because of a smaller number of possible faulty and in the same time faulty elements. The fault diagnosis of a small signal circuit is conducted after the end of DC circuit analysis. Its aim is to search for faults, but only in the set consisting of elements that are absent from the DC circuit. This analysis uses actual values of faulty elements that are results of fault identification in the DC circuit. In order to achieve a correct performance of a diagnostic process it must be assumed that:

- the circuit diagram and nominal values of all elements are known, so it is possible to calculate voltages of all nodes accessible for measurement using any values of differences $\delta_i(i = 1, 2...n)$ between actual x_i and nominal $x_{i,nom}$ parameters of possible faulty elements,
- differences $\delta_i = x_i - x_{i,nom}$ between faulty and nominal values of elements belong to a range assumed for calculating the formulas,
- the number k of simultaneously faulty elements does not exceed the one assumed for calculation.

A. DC Circuit Diagnosis

The diagnostic of a DC circuit is performed using the dictionary method presented in paper [22],[23]. The method's before test stage consists of creating set S of n possible faulty elements with parameters $x_i(i = 1, 2...n)$ and all its subsets $S_p(p = 1, 2...m)$ containing k elements, where k is the maximal number of simultaneously faulty elements. The number m of subsets S_p is equal to number of k-element combinations of n elements. The number m can be calculated using the Newtons formula. The next step consist of calculating for each subset S_p values $v_{p,j}(p = 1, 2...m, j = 1, 2...s)$ of accessible for measurement node voltages using any values of elements $x_i(i = 1, 2...k)$ selected from an in advance fixed range. A value s is a number of nodes accessible for measurement. When the number of values x_i selected from fixed range is equal to z, the result of this step is determination of m sets of $k \cdot z$ values of s node voltages. The formula (7) presents one of z_k subsets of s node voltages corresponding to one of m subsets S_p.

$$\left.\begin{array}{l} v_{p,1} = f_1(x_i, x_j, ..., x_l, \mathbf{b}) \\ v_{p,2} = f_2(x_i, x_j, ..., x_l, \mathbf{b}) \\ \vdots \\ v_{p,s} = f_s(x_i, x_j, ..., x_l, \mathbf{b}) \end{array}\right\} \underbrace{x_i, x_j, ..., x_l}_{k\ parameters} \quad (7)$$

In the formula (7) the vector $\mathbf{b}$ represents all independent sources in circuit under test and the expressions f_i ($i = 1, 2...s$) mean relationships of node voltages, parameters x_i and sources. The relationships f_i need not to be known, but there is a must to calculate node voltages $v_{p,j}$. Those are calculated using k values x_i (for each set S_p) selected from fixed range and nominal values of other elements. The node voltages obtained using formulas (7) are training sets for determining formulas (8) enabling to calculate parameters of faulty elements related to known node voltages for each k-element subset S_p.

$$\left.\begin{array}{l} x_{p,i} = f^*_{p,i}(v_{p,1}, v_{p,2}, ..., v_{p,s}) \\ x_{p,j} = f^*_{p,j}(v_{p,1}, v_{p,2}, ..., v_{p,s}) \\ \vdots \\ x_{p,l} = f^*_{p,l}(v_{p,1}, v_{p,2}, ..., v_{p,s}) \end{array}\right\} k\ equations \quad (8)$$

The formulas $f^*_{p,i}$ are calculated using evolutionary computation method - gene expression programming. The next step of the proposed diagnostic method consists of determining for each set S_p minimum values $(v_{p,i})_{min}$ ($i = 1, 2, ..., s$) and maximum ones $(v_{p,i})_{max}$ ($i = 1, 2, ..., s$) of node voltages accessible for measurement for the fixed ranges of parameters x_i from the considered set S_p. The result of the before test stage is a dictionary with entries associated with subsets S_p. Each dictionary signatures consists of s minimum values $(v_{p,i})_{min}$ of node voltages, s maximum ones $(v_{p,i})_{max}$ and k formulas , ($i = 1, 2, ..., k$) for calculation of faulty parameters. The after test stage consists of searching for all entries of dictionary (each entry is associated with the subset S_p), which fulfils conditions:

$$(v_{p,i})_{min} \leq v_i \leq (v_{p,i})_{max} \quad i = 1, 2, ..., s \quad (9)$$

In the formula (9) v_i means node voltages measured in a faulty circuit. Formulas (8) are associated with each entry and they are used for calculation of faulty parameter values x_i for each selected entry of the dictionary. The sets of k parameter values $x_{p,i}(i = 1, 2, ..., k)$ calculated according to formulas (8) are possible solutions to DC diagnostic problem. Each possible solution needs to be verified. This process consists of checking if all of k calculated parameter values $x_{p,i}$ for fixed p fulfil the conditions obtained from range of maximum fault values. The positive verified sets of k parameter values form the solution.

B. Fault Diagnosis of a Small Signal Circuit

The aim of a small-signal circuit analysis is to calculate all values of these possible faulty elements, which the diagnosed DC circuit does not contain. All results of the parameter identification presented in chapter II.A are taken now into consideration. An algorithm, which consists of an optimization of an encoded form of a solution, is used in order to perform diagnostic proces. The method is presented in paper [24]. It exploits the genetic algorithm. In the proposed method a quantity taken into account is differed by a nominal value difference between the actual value of a possible faulty element and its nominal one.

$$\delta_i = \frac{(x_i - x_{i,nom})}{x_{i,nom}} 100\% \quad (10)$$

Usually, this value is encoded as an 6-element binary set, named a chromosome. Its 5 elements create a binary form of δ_i and the other element determines the sign. If 1% accuracy is used for encoding, the range of δ_i is from -32% to +32%. The state of the circuit, determined by a vector $\boldsymbol{\delta} = [\delta_1, \cdots, \delta_n]^T$ consisting of n values of δ_i, is represented by a sequence of $6n$ binary elements (a genotype). Each 6-element set determines a value of δ_i. The fitness function used in genetic algorithm is as follows:

$$w(\boldsymbol{\delta_i}) = \sum_{k=1}^{n} (100 - E_k) \quad (11)$$

where

$$E_k = \begin{cases} 100 & for & \alpha > 1 \\ 100\,\alpha & for & c \leq \alpha \leq 1 \\ 0 & for & \alpha < c \end{cases} \quad (12)$$

and

$$\alpha = \frac{|v_{k\,cal}(\boldsymbol{\delta}) - v_k|}{v_k} \quad (13)$$

$w(\boldsymbol{\delta})$ - the value of the fitness function, v_k - a voltage value of k node measured in a faulty circuit, $v_{k\,cal}(\boldsymbol{\delta})$ - a voltage value of k node calculated with using parameter values determined by vector $\boldsymbol{\delta}$ associated with the evaluated individual, n - a number of nodes accessible for measurement, c - a constant determined by a measurement accuracy of node voltages. Each difference of node voltages in the formula (13) means the difference of real parts or imaginary ones of complex voltage values measured in a faulty circuit or calculated using vector $\boldsymbol{\delta}$. The number of bits associated with each value δ_i equal to 6 is determined by the measurement accuracy and it can be greater, appropriate to requirements. In order to achieve a correct convergence of the genetic algorithm the number of bits in a genotype does not need to be too great. It results directly from the number of possibly faulty elements and an assumed coding accuracy. Accordingly in the case of great number of possible faulty elements the decomposition of an analised circuit is indispensable. The fault diagnosis using decomposition is presented in the paper [25].

C. Numerical Example

Let us consider an audio amplifier shown in Fig.2. It is assumed that all resistances except the source resistance R_6 and all capacitances can be faulty. In accordance with the proposed fault diagnosis process presented in chapter II the DC circuit of the amplifier shown in the figure 3 needs to be analysed in the first place. The element R_8 is omitted in the creation of the DC circuit because of too small value of the resistance. Its changes cannot be observed as the nodes
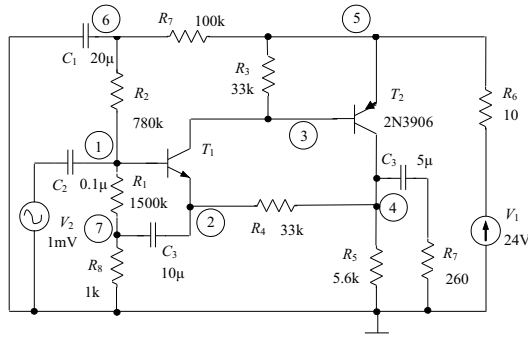
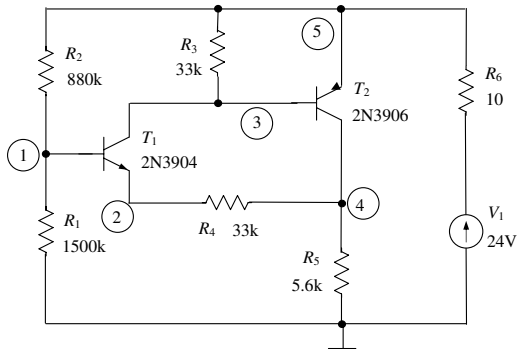Fig. 2. The analysed audio amplifier.

Fig. 3. The DC circuit of the analysed amplifier

accessible for measurement are 1, 2 and 4. The contents of the set of nodes accessible for measurement causes that resistances R_2 and R_7 are replaced with one resistance with the value equal to the series connection equivalent resistance denoted by R_2. So possible faulty elements of the DC circuit are the resistances $R_1 \div R_5$, maximum three at the same time. The maximum difference between the faulty element value and the nominal one is equal to 30% of the nominal value. It is possible to enlarge the range of investigated faults but it leads to a decrease in the accuracy of the calculations based on the formulas (8). In order to determine formulas (8) enabling us to define faulty element values the method named gene expression programming (GEP) with the random numerical constants presented in the paper [17] is used.

Case 1. Let us consider the case, when four elements are faulty: R_2, R_3, R_4 and C_3. The values of parameters are: $R_2 = 577.2k\Omega$, $R_3 = 41.25k\Omega$, $R_4 = 26.73k\Omega$ and $C_3 = 9.1\mu F$. The first step is diagnostic analysis of the DC circuit shown in Fig.3. Six sets consisting of three elements are found as a result of this process. These are:

$$R_1, R_2, R_3; \quad R_1, R_2, R_4; \quad R_1, R_2, R_5; \\ R_2, R_3, R_4; \quad R_2, R_3, R_5; \quad R_2, R_4, R_5. \qquad (14)$$

Four sets are negatively verified and only two sets are the results of this step:

$$R_2 = 575.64k\Omega, \quad R_3 = 39.369k\Omega, \quad R_4 = 26.334k\Omega \\ R_2 = 580.32k\Omega, \quad R_4 = 24.717k\Omega, \quad R_5 = 5.538k\Omega. \qquad (15)$$

The second step is the analysis of the small-signal circuit created by replacing all the transistors presented in Fig.2 by small-signal linear models. The models' parameters correspond with transistors operating points, obtained as the result of DC diagnostic process. The circuit is supplied by the source V_2 having amplitude equals to $1mV$ and zero initial phase. Optimised chromosome contains information about values of capacitors C_1, C_2, C_3 and resistors R_7, R_8. Taking into account real and imaginary parts of measured harmonic components of nodes voltages, optimisation performed with use of GA, leads in the first case to the value $C_3 = 9.2\mu F$ end nominal values of the remaining elements. In the second case the algorithm does not find the solution. Hence, the final values of elements are: $R_2 = 575.64k\Omega, R_3 = 39.369k\Omega, R_4 = 26.334k\Omega, C_3 = 9.2\mu F$. Case 2. Next, faulty elements R_1, R_3, R_4, C_1 are taken into account. For element values $R_1 = 1080k\Omega, R_3 = 23.76k\Omega, R_4 = 41.25k\Omega, C_1 = 14\mu F$, DC analysis leads initially to the five possible solutions. Final verification eliminates four of them and define the set: $R_1 = 1080k\Omega, R_3 = 23.133k\Omega, R_4 = 39.765k\Omega$ as the solution. The small-signal analysis identifies the only change elements values belonging to the optimized set. It leads to the conclusion that final solution of problem is: $R_1 = 1080k\Omega, R_3 = 23.133k\Omega, R_4 = 39.765k\Omega, C_1 = 14\mu F$.

IV. APPLICATION OF DIFFERENTIAL EVOLUTION

The method of implementation of the differential evolution algorithm is very similar to the one of the genetic algorithm described in section III.B. The processed quantity is a relative deviation of the actual value of the parameter x_i from its nominal value $x_{i,nom}$, expressed in percent, represented by formula (10). This quantity is coded in decimal form, and thus the vectors processed by DE have the length determined by the number of parameters whose actual values are sought. The applied objective function used is defined by the relation (11-13), wherein the differences occurring in the node voltages are differences in real and imaginary parts of voltages symbolic values respectively.

A. Numerical Example

Let us consider an preamplifier shown in Fig.4. It is assumed that all resistances $R_1 \div R_9$ and R_F and all capacitances $C_1 \div C_7$ except C_2 and C_7 can be faulty. In accordance with the proposed fault diagnosis process presented in chapter II the DC circuits of the amplifier shown in the Fig.5 needs to be analysed in the first place. In the DC left subcircuit there are 5 possible faulty resistances R_1, R_2, R_5, R_6 and R_8, maximum 2 at the same time. In the DC right subcircuit there are 4 possible faulty resistances R_3, R_4, R_7 and R_9, maximum 2 at the same time. The maximum difference between the faulty element value and the nominal one is equal to 30% of the nominal one. In order to determine formulas (8) enabling us to calculate faulty element values the method named gene

expression programming is used. Let us consider the case, when five elements are faulty: R_1, R_6, R_4, R_7 and C_5. The actual values of parameters are: $R_1 = 132.5k\Omega$, $R_6 = 13k\Omega$, $R_4 = 7.82k\Omega$, $R_7 = 1.2k\Omega$ and $C_5 = 8\mu F$. The first step is diagnostic analysis of the left DC subcircuit shown in Fig.5. Three sets consisting of two elements are found as a result of this process. These are:

$$R_1, R_2; \quad R_1, R_6; \quad R_5, R_6. \tag{16}$$

Two sets are negatively verified and only one set is the result of this step:

$$R_1 = 133.2k\Omega; R_6 = 12.9k\Omega. \tag{17}$$

In the right DC subcircuit three sets consisting of two elements are found as a result of diagnostic process. These are:

$$R_3, R_7; \quad R_9, R_7; \quad R_4, R_7. \tag{18}$$

The set R_9, R_7 is negatively verified and two sets are the results of this step:

$$\begin{aligned} R_3 = 124k\Omega; R_7 = 1.05k\Omega \\ R_4 = 7.97k\Omega; R_7 = 1.29k\Omega \end{aligned} \tag{19}$$

The results of DC diagnostic process are two sets:

$$\begin{aligned} R_1 = 133.2k\Omega; R_6 = 12.9k\Omega; R_3 = 124k\Omega; R_7 = 1.05k\Omega \\ R_1 = 133.2k\Omega; R_6 = 12.9k\Omega; R_4 = 7.97k\Omega; R_7 = 1.29k\Omega \end{aligned} \tag{20}$$

The next step is the analysis of the small-signal circuit created by replacing all the transistors presented in Fig.4 by small-signal linear models. The values of model parameters correspond to transistors' operating points. This take into account the obtained values of faulty resistors. The circuit is supplied by the sinusoidal source having amplitude equal to 1mV and zero initial phase. Chromosome that contains information about values of capacitors C_1, C_3, C_4, C_5, C_6 and resistor R_F has six genes. It is optimised with the use of DE. Fitness function takes into account real and imaginary parts of measured harmonic components of node voltages: 1, 3, 4 and 5. The DE optimisation leads in the first case to the value $C_5 = 9\mu F$ and nominal values of the remaining elements. In the second case the value $C_5 = 7.6\mu F$ is determined. Hence, the final solution to the problem are two sets:

$$\begin{aligned} a) R_1 = 133.2k\Omega; R_6 = 12.9k\Omega; \\ R_3 = 124k\Omega; R_7 = 1.05k\Omega; C_5 = 9\mu F; \\ b) R_1 = 133.2k\Omega; R_6 = 12.9k\Omega; \\ R_4 = 7.97k\Omega; R_7 = 1.29k\Omega; C_5 = 7.6\mu F; \end{aligned} \tag{21}$$

Both above sets form ambiguity groups. Exclusion of one of them requires additional measurements.

V. CONCLUSION

The presented method consists of two stages. First stage, uses GEP algorithm, belongs to the SBT (simulation before test) methods and second stage, uses GA or DE, to SAT (simulation after test) methods. At the first stage, the time indispensable for calculating formulas (7) and (8) increases with the number of possible faulty elements in a circuit.
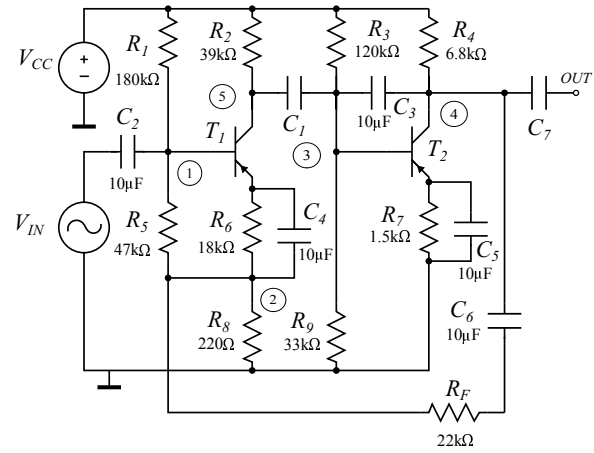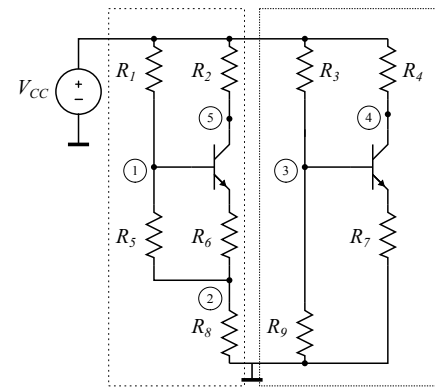

Fig. 4. The analysed audio preamplifier.

Fig. 5. The DC circuit of the analysed preamplifier consisting of two separate subcircuits

The method enable us to split all possible faulty elements in investigated circuits into two sets. One of them is associated with the DC circuit, the other only with the small-signal circuit. In result, the number of possible faulty elements in DC circuit is smaller as in the case of only one investigated circuit and the time consuming for calculating formulas (7) and (8) decreases. The number of possible faulty elements in the small signal circuit, at the second stage of the method, is also smaller than the number of all possible faulty elements because actual values of all faulty elements from first set (associated with the DC circuit) are calculated at the first stage. The fact that the small-signal models parameters are calculated on the basis of faulty elements values found during DC analysis (first stage of the method), almost always vitiated by errors, seems to be the certain disadvantage of the proposed diagnostic scheme. Measurement of harmonic components of node voltages values (real and imaginary parts or amplitude and phase) having relatively small values is also the weak point of the method. The time consumed for calculating formulas (8) increased and their accuracy decreases with the number of concurrently faulty elements. In result the precision of the presented method is pure for 4 or more simultaneously faulty elements of DC circuit. The disadvantage of the proposed method is the fact that it does not take into account the tolerances of healthy elements values. The authors have developed the variant of

the method for handling tolerances. Unfortunately, it requires many time consuming calculations during before test stage of the diagnostic procedure.

REFERENCES

[1] N.Sen and R.Saeks, "Fault diagnosis for linear systems via multi-frequency measurement," *IEEE Transaction on Circuits and Systems*, vol. 26, pp. 457–465, 1979.

[2] H.Chen and R.Saeks, "A search algorithm for the solution of multifrequency fault diagnosis equations," *IEEE Transaction on Circuits and Systems*, vol. 26, pp. 589–594, 1979.

[3] J.W.Bandler and A.E.Salama, "Fault diagnosis of analog circuits," *Proceedings IEEE*, vol. 73, pp. 1279–1325, 1981.

[4] P.M.Lin and Y.S.Elcherif, "Analoque circuits fault dictionary - new approaches and implementation," *International Journal of Circuit Theory and Applications*, vol. 13, pp. 149–172, 1985.

[5] T.Ozawa, *Analog Methods for Computer Aided Analysis and Diagnosis*. New York, USA: Marcel Dekker, 1988.

[6] D.Liu and J.A.Starzyk, "A generalized fault diagnosis method in dynamic analoque circuits," *International Journal of Circuit Theory and Applications*, vol. 30, pp. 487–510, 2002.

[7] M.Tadeusiewicz, S.Hałgas, and M.Korzybski, "An algorithm for soft-fault diagnosis of linear and nonlinear circuits," *IEEE Transactions on Circuits and Systems-I: Fundamental Theory and Applications*, vol. 49, pp. 1648–1653, 2002.

[8] J.Rutkowski, *Słownikowe metody diagnostyczne analogowych układów elektronicznych*. Wydawnictwa Komunikacji i Łączności, Warszawa, first ed., 2003.

[9] F.Aminian, M.Aminian, and H.W.Collin, "Analog fault diagnosis of actual circuits using neural networks," *IEEE Transactions on Instrumentation and Measurements*, vol. 51, pp. 544–550, 2002.

[10] T.Golonek, D.Grzechca, and J.Rutkowski, "Evolutionary metod for test frequences selection based on entropy index and ambiguity sets," in *Proceedingsof International Conference on Signals and Electronic Systems ICSES 06*, pp. 511–514, June 2006.

[11] M.Tadeusiewicz and S.Hałgas, "An algorithm for multiple fault diagnosis in analog circuits," *International Journal of Circuit Theory and Applications, J.Wiley and Sons. Ltd.*, vol. 34, pp. 607–615, 2006.

[12] M.Korzybski, "Diagnozowanie wielokrotnych uszkodzeń parametrycznych w obwodach tranzystorowo-diodowych," *Proceedings of VII KKE*, 2008.

[13] D.E.Goldberg, *Algorytmy genetyczne i ich zastosowania*. WNT Warszawa, 1998.

[14] Z.Michałkiewicz, *Algorytmy Genetyczne + Struktury Danych = Programy Ewolucyjne*. WNT Warszawa, 1999.

[15] D.Rutkowska, M.Piliński, and L.Rutkowski, *Sieci neuronowe, algorytmy genetyczne i systemy rozmyte*. Wydawnictwo Naukowe PWN, 1999.

[16] J.Arabas, *Wykady z algorytmów ewolucyjnych*. WNT Warszawa, 2000.

[17] C.Ferreira, "Gene expression programming: a new adaptive algorithm for solving problem," *Complex Systems*, vol. 13 (2), pp. 87–129, 2001.

[18] C.Ferreira, "Gene expression programming in problem solving," in *Proceedings of the 6th Online World Conference on Soft Computing in Industrial Applications*, 2001.

[19] C.Ferreira, "Genetic representation and genetic neutrality in gene expression programming," *Advances in Complex Systems*, vol. 5(4), pp. 389–408, 2002.

[20] C.Ferreira, *Gene Expression Programming: Mathematical Modeling by an Artifical Intelligence*. Springer, 2006.

[21] R.Storn and K.Price, "Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization, Kluwer Academic Publishers*, vol. 11, pp. 341–359, 1997.

[22] M.Korzybski, "Zastosowanie metody ewolucyjnej do lokalizacji i identyfikacji uszkodzeń," *Materiały VI Krajowej Konferencji Elektroniki, Darłowo*, vol. 1, pp. 169–174, 2007.

[23] M.Korzybski, "Dictionary method for multiple soft and catastrophic fault diagnosis based on evolutionary computation," *Proceedings of International Conference on Signals and Electronic Systems*, pp. 553–556, 2008.

[24] M.Korzybski, "Zastosowanie algorytmów genetycznych do lokalizacji i identyfikacji uszkodzeń," *Materiały XXV IC-SPETO*, vol. 1, pp. 443–446, 2002.

[25] M.Tadeusiewicz and M.Korzybski, "A method for fault diagnosis in linear electronic circuits," *International Journal of Circuit Theory and Applications, J.Wiley and Sons. Ltd.,vol.28*, vol. 28, pp. 245–262, 2000.