

We are IntechOpen, the world's leading publisher of Open Access books Built by scientists, for scientists

6,900

Open access books available

186,000

International authors and editors

200M

Downloads

Our authors are among the

154

Countries delivered to

TOP 1%

most cited scientists

12.2%

Contributors from top 500 universities



WEB OF SCIENCE™

Selection of our books indexed in the Book Citation Index
in Web of Science™ Core Collection (BKCI)

Interested in publishing with us?
Contact book.department@intechopen.com

Numbers displayed above are based on latest data collected.
For more information visit www.intechopen.com



Memetic Method for Passive Filters Design

Tomasz Golonek and Jantos Piotr

Additional information is available at the end of the chapter

<http://dx.doi.org/10.5772/53716>

1. Introduction

The design of analog passive filters with specialized (not typical) frequency responses is not a trivial problem. The presence of finite load impedances for filter sections and limited quality factors of coils are just two of many concerns which a design engineer has to take into account. Additionally, classical techniques of filters synthesis require assuming of the approximation type (e.g. Butterworth, Chebyshev) before calculating the filter transfer function's poles and zeroes. This choice is frequently a challenge itself.

One of the methods allowing for elimination of the mentioned problems is the use of evolutionary computations (EC). Evolutionary techniques are a well known and frequently used tool of global optimization [1-3]. This kind of the optimization imitates natural processes of individuals' competition as candidates for reproduction. Better fitted individuals have higher survival probability and their genetic material is preferred. During the recombination process some parts of parents' genotypes are exchanged and offspring individuals are created. A new generation collected after the succession procedure conserves the features consisted in the previous genotypes. Besides, to assure a system resistance for a stagnation effect, mutation operations are applied to EC. The most popular sorts of EC approaches are: genetic algorithm (GA), genetic programming (GP), evolutionary strategies (ES), differential evolution (DE) and gene expression programming (GEP).

The main drawback of evolutionary approaches is an ineffective and insufficient local optimization. This property and significant computational efforts necessary for a huge generation processing predispose the EC applicability especially to the NP hard global searching problems [4-10]. This chapter describes the passive filters synthesis method by means of EC. The process of circuits' automated designing is a very complex issue. A wide area of solutions should be probed during the early stage of computations and its local parameters should be finally optimized. In contrary to the alternative systems [4-6], the method present-

ed in this chapter is based on an application of a hybrid system - a synergy of genetic programming (GP) (used for the purpose of determining an optimal network of a passive filter circuit) and a deterministic local search by the means of Hooke and Jeeves method (HJM), which enables the system to find accurate values of the filter's elements. The proposed design system allows for obtaining the desired frequency response and, optionally, production yield optimization.

Section 2 explains the general algorithm of the proposed system, Sections from 3 to 5 present the descriptions of the important details of the algorithm. Next, in Section 6, the exemplary results of an automated circuit design are placed. Finally, in Section 7, some considerations of the method future development and final conclusions are presented.

2. Optimization process overview

The process is initialized with the desired filter specifications. Additional algorithms' parameters, e.g. population size, number of Monte Carlo analyses, and the like, are assumed. The use of GP and HJM is briefly presented in following paragraphs.

In the presented research both filters topology and circuit parameters values are being optimized. As far as GP has been proven an effective tool of circuits networks determination, it is not as efficient in adjusting resistors, coils, or capacitors values. The latter has been solved by the use of a deterministic, non-gradient local search algorithm - Hooke and Jeeves direct search method [11]. A synergy of evolutionary global optimization and local optimization algorithm is called a memetic algorithm [12-16]. In the presented research the proposition of the memetic genetic programming (MGP) introduction for the purpose of the analog filters design is described.

The block diagram of the optimization process has been presented in Fig. 1. After system running, the first, primary generation of the N_{mx} individuals is created randomly. It is very important to assure the possible wide range of dispersion for the starting solutions, so the diversity of this population is extremely desired. In the proposed system, the uniform probability of the primary individuals' randomization with the maximal allowed size limitation is applied. To evaluate the actual, random solutions, the fitness function is determined for each individual from population. The distances between the parameters of the evaluated phenotypes and the target specifications are checked during this process. Next, the GP system is executed. During reproduction, the mating pool is collected with the reproduction method that prefers more fitted individuals. An adequate strength of selection pressure has to be kept during this process, and it has crucial impact for a system convergence. During recombination process, pairs selected from an intermediate pool are crossed with assumed probability. Besides, offspring genotypes can be mutated and it assures adequate veracity of the population and allows for achieving the new regions of the searching space. Detailed information about the genetic operations and the fitness function of the GP part of the proposed memetic system are included in Section 4.

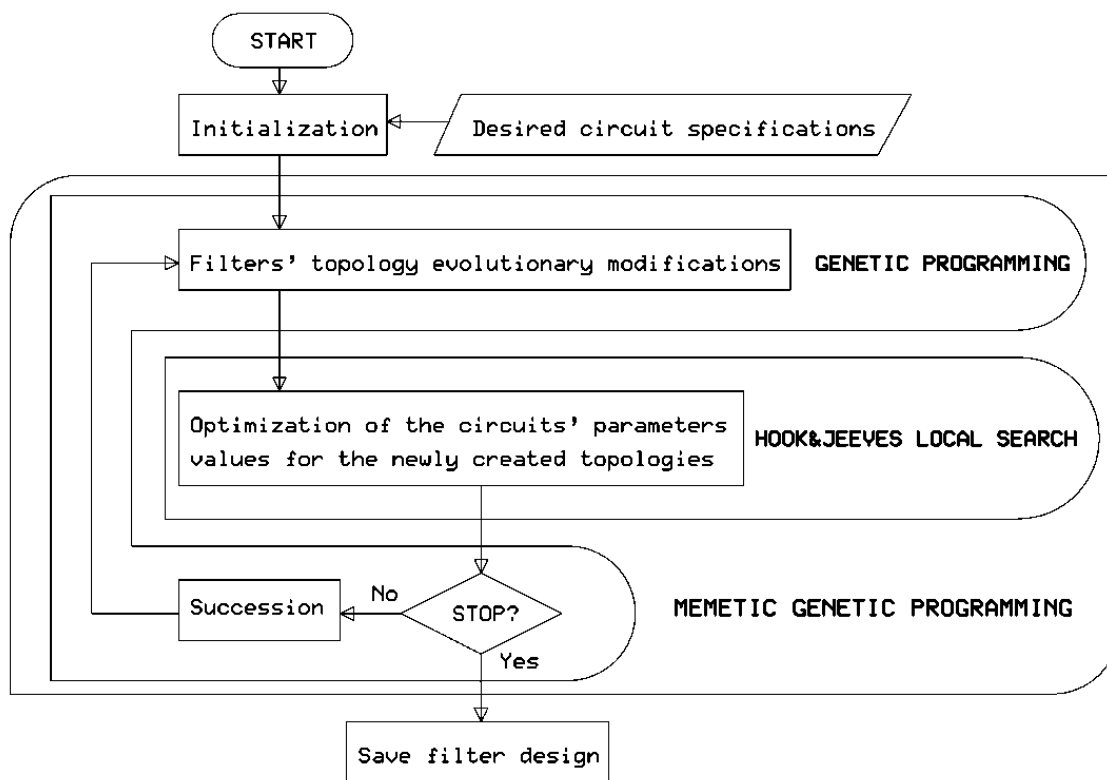


Figure 1. The process overview

This heuristic stage impacts on the coded circuits topology especially (i.e. the values of the filter resistances, inductances and capacitances are optimized not effectively), however the values of its elements are adjusted on the next stage during local deterministic optimization.

The newly created population of solutions represents a sort of circuits with non optimal values of its elements and now they are determined by means of HJM (pattern search) algorithm.

The differential of the fitness function is unknown. Hence, the local search algorithm could not have required its gradient computation. Among various optimization methods that meet this requirement, the pattern search method is characterized by its simplicity. The HJM method consists of two moves, i.e. exploratory and pattern, repeated sequentially until stop conditions are fulfilled.

Evolutionary algorithms are a trade-off between a global and local optimization. As far as they provide a good solution it never is but a sub-optimal one. The aim of the presented research was to achieve the best possible solution. Hence, it has been decided to use local search algorithms. The application of a local search algorithm after the evolutionary optimization is completed would improve the performance of the last chosen individual. Though, it would not affect the optimization process itself. Memetic solutions aim to improve the overall optimization process. The local search algorithm is applied within the evolutionary algorithm's main loop. It allows for faster convergence of the process. Applying a full local search process within each of the evolutionary iterations would influence the required computation time significant-

ly and negatively. Therefore, only a few iterations of the HJM were allowed. For the purpose of improving the final result a full HJM cycle is applied after the evolutionary cycle is over.

In the presented approach the Hooke and Jeeves method had to be modified so the search is carried out in the assumed search space (E24 sequence of resistances, capacitances and inductances). The detailed description of the algorithm is presented in Section 5.

Next, after the topology and the values of elements determination, if the production yield optimization is required, the Monte Carlo analysis can be applied. The algorithm is terminated if all specification (and yield) requirements are met or if the last allowed generation G_{mx} was reached.

3. Circuit structure coding

The filter circuit structure is coded with the use of a binary tree structure. An exemplary tree is illustrated in Fig. 2. Its inner nodes contain functions, however terminals keep their arguments. The length from a root node to the most remote leaf is defined by the maximal depth D_{mx} and it reaches the value $D_{mx}=4$ for the illustrated case. Finally, the tree presented in Fig.2 can be decoded as a general expression given below:

$$y = f_1(f_2(t_1, f_3(t_2, t_3)), f_4(t_4, t_5)). \quad (1)$$

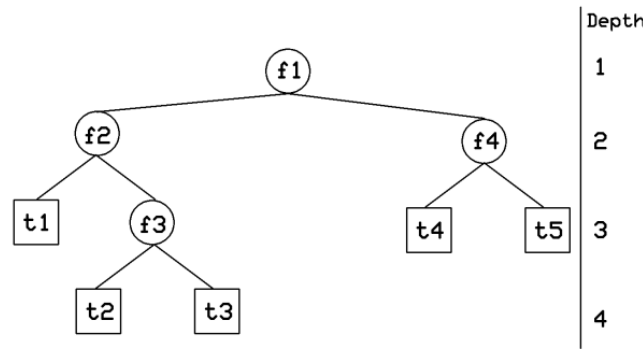


Figure 2. The example of tree structure

This kind of structure allows for defining the circuit topology and values of its elements in the flexible way with the assumed type of connection functions defined in F and the set of basic terminal blocks defined in T . Besides, it enables an easy way to the genetic operations implementation in the GP system. The set F contains unique symbols for connection types and for the described system it was defined as follows:

$$F = \{—, >>, | \}, \quad (2)$$

where contained symbols denote a serial, a cascade and a parallel configuration respectively.

Generally, passive filters circuits are constructed by the use of resistors, capacitors and magnetic coils (inductors), so the T-typed three-terminal (two-port) blocks which contain these elements (one in each branch) were chosen as basic ones:

$$\mathbf{T} = \{ R v_e o_u L v_e o_u C v_e o_u, L v_e o_u C v_e o_u R v_e o_u, C v_e o_u R v_e o_u L v_e o_u \}, \quad (3)$$

where integer numbers v_e for $e=(1,..,E)$ and o_u for $u=(1,..,U)$ determine the value and its order for the element identified by the antecedent descriptor (i.e. R for a resistor, L for a inductor and C for a capacitor). In the proposed system, discrete values v_i are selected from a set \mathbf{V} that contains elements from a widely known practical series E24 ($E=24$) prepared for the components manufactured with tolerance equal to $\delta_{tol}=5\%$:

$$\mathbf{V} = \left\{ \begin{array}{l} 10, 11, 12, 13, 15, 16, 18, 20, 22, 24, 27, 30, \\ 33, 36, 39, 43, 47, 51, 56, 62, 68, 75, 82, 91 \end{array} \right\}. \quad (4)$$

Order values are included in the set \mathbf{O} and they should enable the desired range of coding, and consist of U integer ciphers ($0 \leq o_u \leq 9$) which determine exponents of decimal multipliers:

$$\mathbf{O} = \{ o_0, \dots, o_{(U-1)} \}. \quad (5)$$

Finally, values of resistance R , inductance L and capacitance C are calculated from the equations (6) ÷ (8) respectively:

$$R = v_e \cdot 10^{o_u} [\Omega], \quad (6)$$

$$L = v_e \cdot 10^{o_u} [\mu\text{H}], \quad (7)$$

$$C = v_e \cdot 10^{o_u} [\text{pF}]. \quad (8)$$

The idea of the terminals' T (basic circuit blocks) coding is illustrated in Fig.3, where the three exemplary two-ports and adequate coding strings are presented. Impedances of elements from two-ports branches of these blocks are calculated from:

$$Z_R = R, \quad (9)$$

$$Z_L = 2\pi fL, \quad (10)$$

$$Z_C = \frac{1}{j2\pi fC}, \quad (11)$$

where f is a signal frequency and j denotes imaginary unit. Impedances Z_1, Z_2, Z_3 for terminal illustrated in Fig.3a can be determined according to equations (9), (10), (11) and its element sequence (i.e. $Z_1=Z_R, Z_2=Z_L, Z_3=Z_C$ for RLC-typed leaf, $Z_1=Z_L, Z_2=Z_C, Z_3=Z_R$ for LCR-typed leaf and $Z_1=Z_C, Z_2=Z_R, Z_3=Z_L$ for CRL-typed leaf).

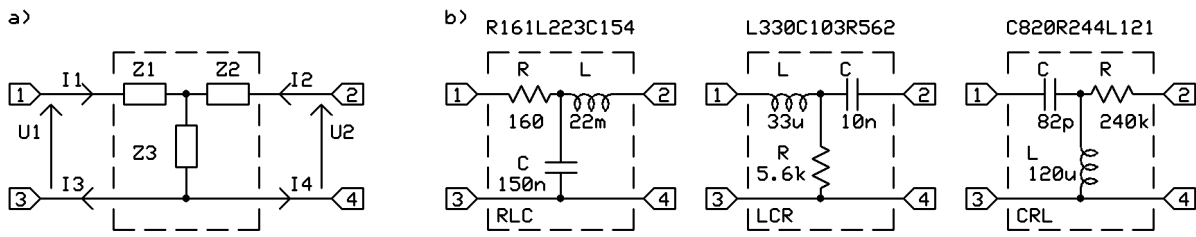


Figure 3. Terminal block structures: a) general, b) exemplary two-ports coded by the respective above strings

Finally, basing on elements from (2) and (3), the trees which code passive filters circuits can be constructed and some example is presented in Fig.4.

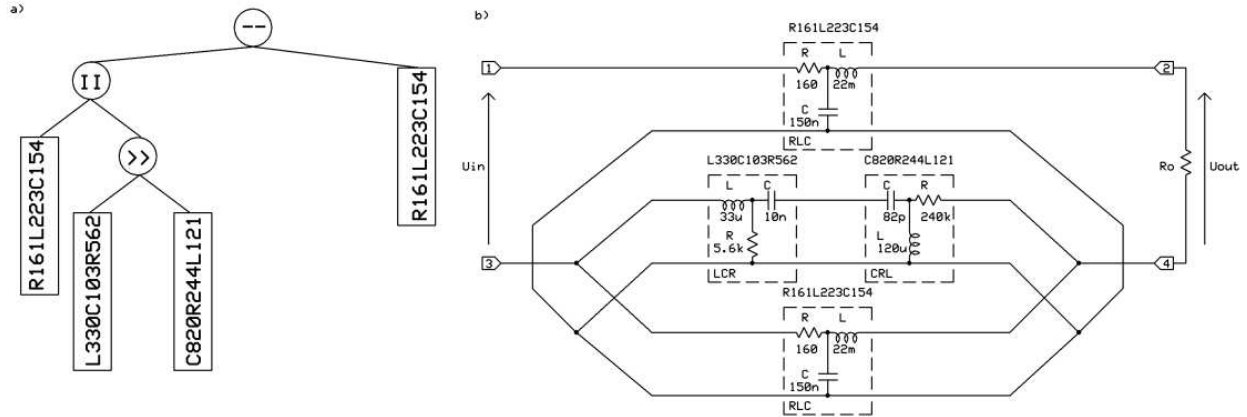


Figure 4. Exemplary structures of: a) genotype, b) phenotype

3.1. Analysis of the blocks connections

An important capability of the technique described is that it assures galvanic (direct) connection of the ground line between input and output ports and it is very desired for practically implemented filter circuits. Besides, as can be seen in the analysis below, each type of

the basic blocks configurations allows for defining an equivalent T-typed circuit and it unifies the interpretation of genotypes of any shapes and sizes.

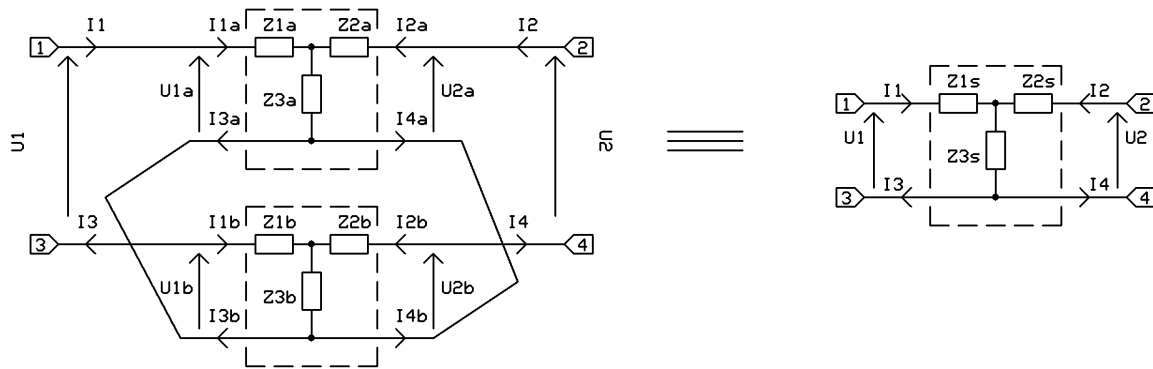


Figure 5. Basic blocks connected in series and its equivalent circuit

Two blocks configured in a series (genotype node denoted as '--') are illustrated in Fig.5. A simple analysis of this circuit leads to equations defining the impedances for the equivalent circuits:

$$Z_{1s} = Z_{1a} + Z_{1b}, \quad (12)$$

$$Z_{2s} = Z_{2a} + Z_{2b}, \quad (13)$$

$$Z_{3s} = Z_{3a} + Z_{3b}, \quad (14)$$

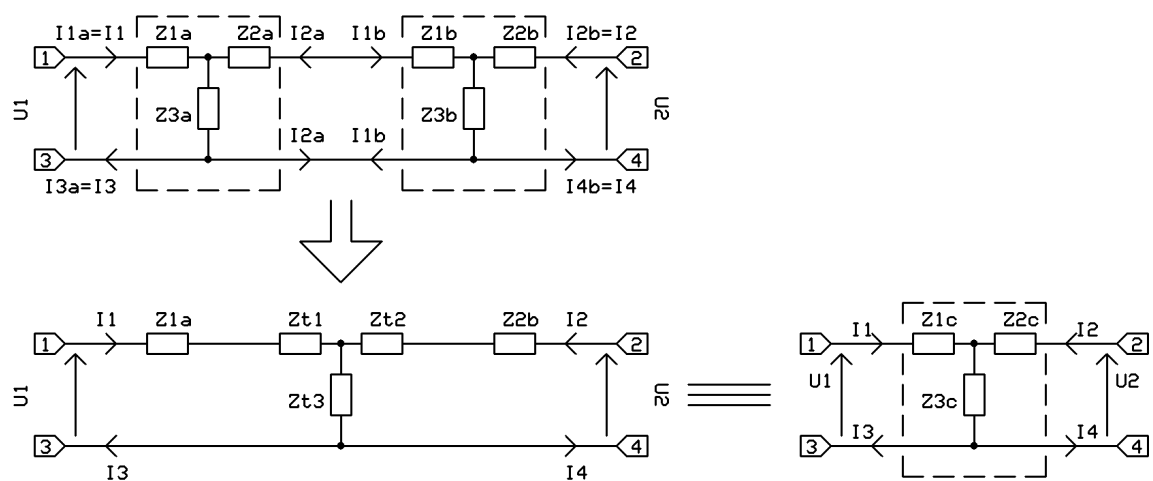


Figure 6. Basic blocks connected in cascade and its equivalent circuit

As can be seen in Fig.6, for a cascade connection (genotype node denoted as '>>') a conversion from a Π to a T-typed structure can be used for an equivalent circuit determination:

$$Z_{1c} = Z_{1a} + Z_{t1} = Z_{1a} + \frac{(Z_{2a} + Z_{1b})Z_{3a}}{Z_{\Pi ab}}, \quad (15)$$

$$Z_{2c} = Z_{2b} + Z_{t2} = Z_{2b} + \frac{(Z_{2a} + Z_{1b})Z_{3b}}{Z_{\Pi ab}}, \quad (16)$$

$$Z_{3c} = \frac{Z_{3a}Z_{3b}}{Z_{\Pi ab}}, \quad (17)$$

where:

$$Z_{\Pi ab} = Z_{2a} + Z_{3a} + Z_{1b} + Z_{3b}. \quad (18)$$

The parallel configuration (genotype node denoted as '| |') is the last kind of connection assumed in (2) and it can be easily modeled by an equivalent circuit after a few transformations explained in Fig.7. Finally, impedances for an equivalent circuit can be calculated from relations:

$$Z_{1p} = \frac{Z_{1xy}Z_{3xy}}{Z_{\Pi xy}}, \quad (19)$$

$$Z_{2p} = \frac{Z_{2xy}Z_{3xy}}{Z_{\Pi xy}}, \quad (20)$$

$$Z_{3p} = \frac{Z_{1xy}Z_{2xy}}{Z_{\Pi xy}}, \quad (21)$$

where:

$$\frac{1}{Z_{1xy}} = \frac{1}{Z_{1x}} + \frac{1}{Z_{1y}} = \frac{1}{Z_{1a} + Z_{3a} + \frac{Z_{1a}Z_{3a}}{Z_{Ta}}} + \frac{1}{Z_{1b} + Z_{3b} + \frac{Z_{1b}Z_{3b}}{Z_{Tb}}}, \quad (22)$$

$$\frac{1}{Z_{2xy}} = \frac{1}{Z_{2x}} + \frac{1}{Z_{2y}} = \frac{1}{Z_{2a} + Z_{3a} + \frac{Z_{2a}Z_{3a}}{Z_{Ta}}} + \frac{1}{Z_{2b} + Z_{3b} + \frac{Z_{2b}Z_{3b}}{Z_{Tb}}}, \quad (23)$$

$$\frac{1}{Z_{3xy}} = \frac{1}{Z_{3x}} + \frac{1}{Z_{3y}} = \frac{1}{Z_{1a} + Z_{2a} + \frac{Z_{1a}Z_{2a}}{Z_{Ta}}} + \frac{1}{Z_{1b} + Z_{2b} + \frac{Z_{1b}Z_{2b}}{Z_{Tb}}}, \quad (24)$$

$$Z_{\Pi xy} = Z_{1xy} + Z_{1xy} + Z_{1xy}. \quad (25)$$

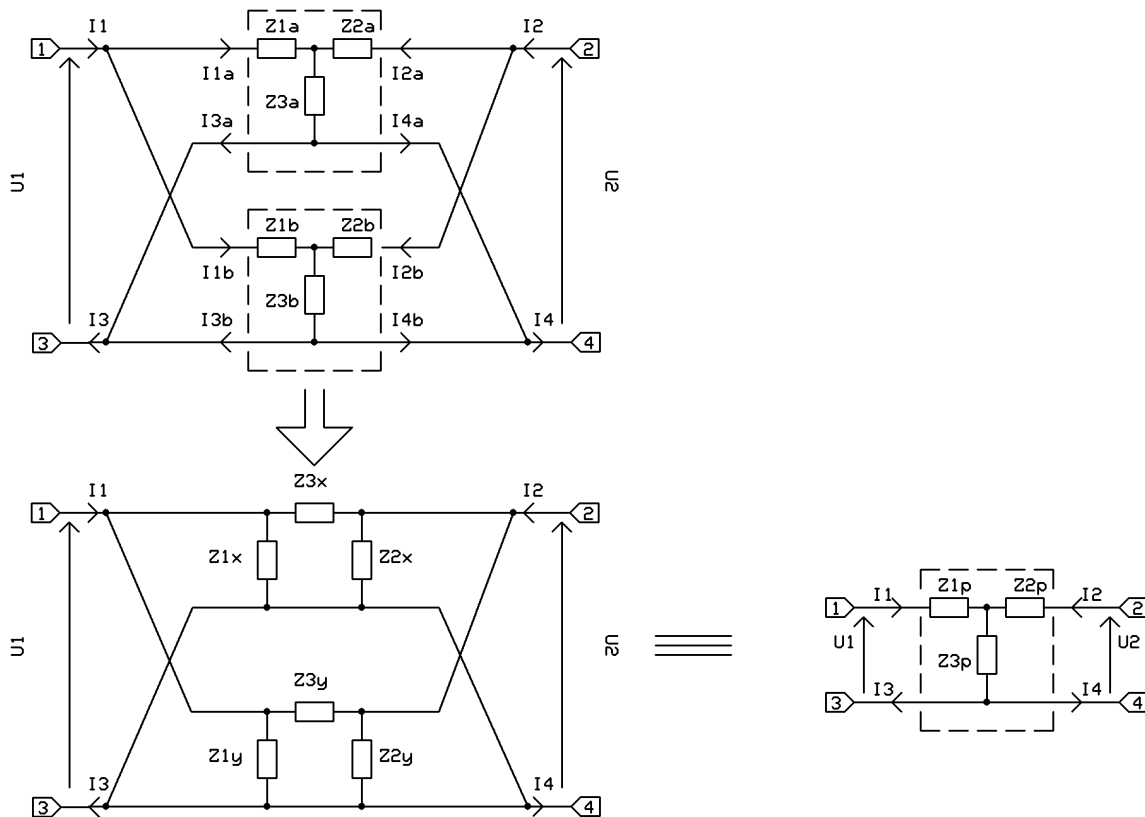


Figure 7. Basic blocks connected in parallel and the equivalent circuit

The transformations described above allow to obtain the resultant impedances Z_{1r} , Z_{2r} and Z_{3r} of the filter circuit (Fig.8) coded by a genotype tree. Finally, a frequency response of the filter loaded by the impedance Z_o can be calculated from:

$$K = \frac{U_{out}}{U_{in}} = \frac{Z_o}{(Z_o + Z_{2r}) \left(1 + \frac{Z_{1r}}{Z_{3r}} \right) + Z_{1r}}. \quad (26)$$

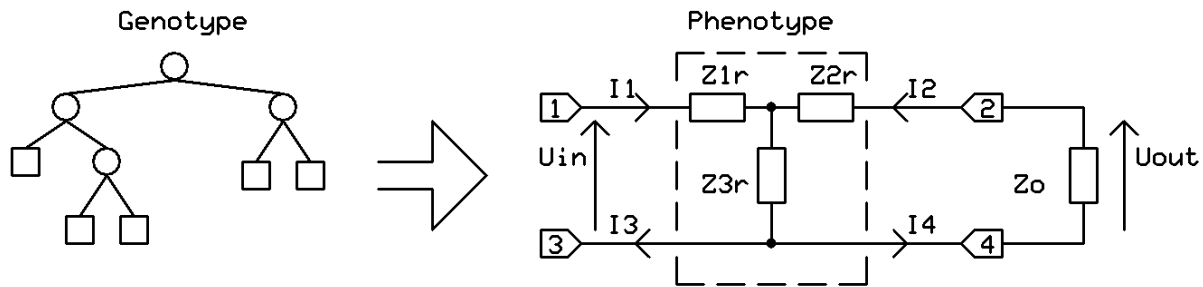


Figure 8. The resultant equivalent filter circuit

4. Genetic operations and fitness calculation

4.1. Reproduction and crossover

The first genetic operation executed after primary generation evaluation is reproduction and it completes a mating pool. A rang method of reproduction is applied in the proposed solution. This kind of operation assures the minimization of the evolutionary system's tendency to promote the average fitted phenotypes (selection pressure regulation). The probability P_{sel} of an individual selection for an intermediate pool of candidates for recombination depends on rang r value:

$$P_{sel} = P_{mn} + (1 - P_{mn}) \frac{r}{N_{mx}}, \quad (27)$$

where P_{mn} denotes an assumed minimal value of the resultant probability (to avoid the currently most fitted genotypes domination), and N_{mx} is a total number of individuals in the population. The rang r is equal to the position of an individual in population ordered by a fitness value (i.e. for the worst $r=1$ and for the best evaluated $r=N_{mx}$).

The crossover (recombination of the genetic material) process is carried out with probability P_{cr} for two parental genotypes selected randomly (with uniform probability) from a mating pool and its idea is illustrated in Fig.9. After two crossover points CP1 and CP2 random designation for genotypes (individually for each one), adequate sub-trees are exchanged between them. This process allows for preserving and propagating the genetic information of the well fitted individuals inside the generation and this operation promises to obtain better evaluated offspring genotypes. Additionally, for the proposed system, the protection against too much tree growing was applied to crossover process. Recombination is accepted only if for each offspring individual the maximal depth does not exceed the assumed D_{mx} value.

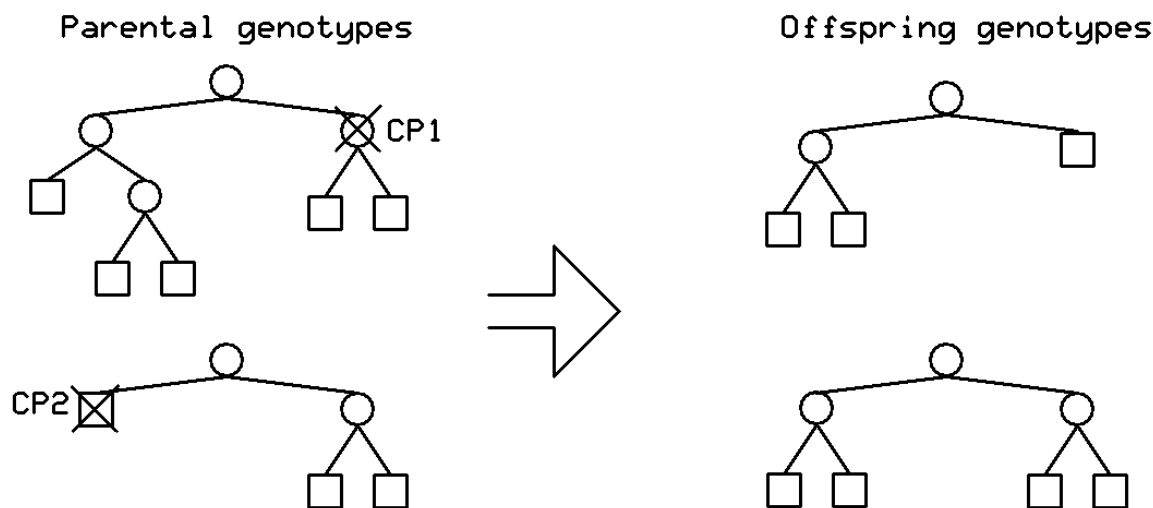


Figure 9. The idea of the crossover process

4.2. Genotypes mutations

The main goal of the mutation operations is the protection against the optimization process stagnation in the local area of the searching space. Four types of mutation procedure are used in the system described:

- a function node mutation with probability P_{1mu}
- a terminal node mutation with probability P_{2mu}
- a sub-tree deletion with probability P_{3mu}
- a sub-tree mutation with probability P_{4mu} .

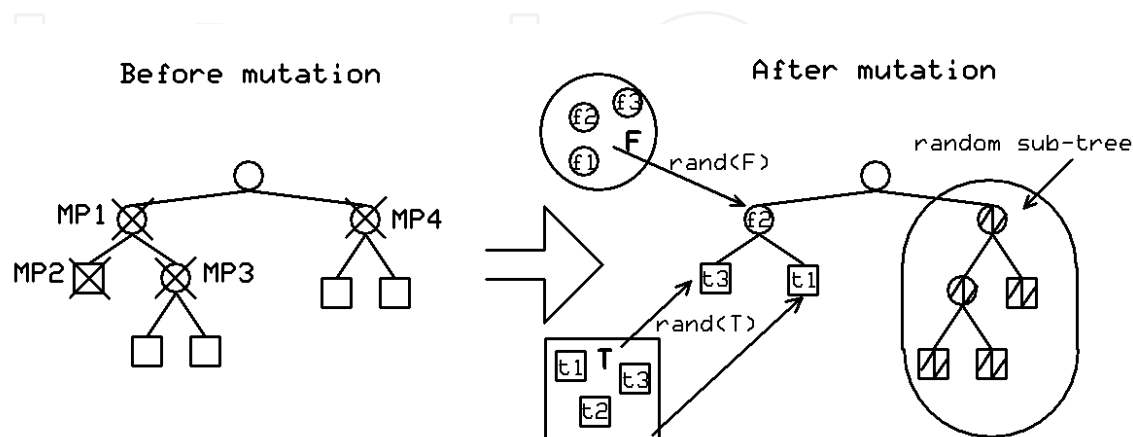


Figure 10. Genotype mutations illustration

The ideas of all these genotype modifications are illustrated in Fig.10. In case of node mutation, it is replaced by a random one (function or terminal adequately) selected from the initially assumed set (2) or (3). During the sub-tree mutation process it is deleted or replaced by a randomly created one. Sub-tree cutting allows for simplifying the phenotype, so it is desired to the filter circuit size minimization. On the other hand, the new sub-tree adding can lead to a genotype growing, so this process is controlled and only results with the maximal depth up to D_{mx} are accepted.

4.3. Fitness value calculation

The quality of the phenotype is evaluated by means of fitness value calculation:

$$Q = \sqrt{\frac{\sum_{m=1}^M (|K_m| - |K_{0m}|)^2}{w_a|_{a=1}^A}} + \max \left(\frac{(|K_m| - |K_{0m}|)_{|m=1}^M}{w_a|_{a=1}^A} \right). \quad (28)$$

Equation (28) is a sum of two components. The first one is a mean square distance of the $|K_m|$ amplitude response of the tested phenotype from the desired region of the filter design specifications $|K_{0m}|$ and it is calculated in M points totally of frequency response for the assumed range of analysis. Besides, to assure the same impact on optimization for each frequency band a defined in filter specifications, its distance is averaged by division by the number w_a of frequency points included in the actually analyzed band. The second one is the average value of the maximal deviation of the frequency point for bands and it additionally prevents the system from stagnation at average solutions. The memetic optimization system minimizes Q during evolutionary cycles and for the best phenotype this distance reaches zero.

After offspring individuals creation its terminal nodes (i.e. filter elements values) are searched with the use of HJM in the way described in the next section and after collecting the new generation it replaces the previous one during the succession process. Finally, the best fitted genotype codes the structure of the filter circuit.

5. Local search

5.1. Hooke and Jeeves local search algorithm

The pattern search optimization algorithm consists of two types of moves, i.e. exploratory move and pattern move.

The purpose of the first of them is to find and utilize information about the optimized function values around the current base point $\mathbf{b}_{(k=0)}$ (individual). In increasing order of indexes, each of the following circuit parameters' values is given a small increment (firstly in positive and then, if required, in the negative direction). The value of the fitness

function is checked and if there is a progress noticed, the value of this variable will be kept as a new $\mathbf{b}_{(k+1)}$ vector. This step is being repeated with reduced range of the parameters values change. When no fitness function value improvement is possible, a pattern move, starting from the current point, is made.

The aim of the pattern move is to speed up the search by information gained about the optimized function and to find the best search direction. A move from the current $\mathbf{b}_{(k+1)}$ in the direction given with $\mathbf{b}_{(k+1)} - \mathbf{b}_k$ is made. The fitness function value is calculated in the point given by

$$\mathbf{p}_k = \mathbf{b}_{(k+1)} - 2\mathbf{b}_k. \quad (29)$$

The procedure is continued with a new sequence of exploratory moves starting from the point \mathbf{p}_k . If the achieved fitness function's value is lower than the one around point \mathbf{b}_k , then a new base point $\mathbf{b}_{(k+2)}$ has been found. In this case a new pattern move (29) is carried out. Otherwise, the pattern move from $\mathbf{b}_{(k+1)}$ is dropped and the new exploratory sequence starts from $\mathbf{b}_{(k+1)}$. The procedure is continued until the length of the step for each of the variables has been reduced to the value assumed previously.

5.2. The implementation of the Hooke-Jeeves method

The search space in the presented research is not only discrete but also limited by the E24 sequence. Considering it several correcting procedures needed to be implemented.

5.2.1. Increasing and decreasing values of the circuit parameters

The first problem to be addressed was to properly increase the values of the circuit parameters. It has been achieved by translating the \mathbf{V} and \mathbf{O} sets into two vectors of integers:

$$\mathbf{V}_v = \{1, \dots, 24\}, \quad (30)$$

$$\mathbf{V}_o = \{1, \dots, 7\}, \quad (31)$$

where: $\mathbf{V}_v(1)$ denotes $v_1=10$, $\mathbf{V}_v(2)$ denotes $v_2=11$, etc. $\mathbf{V}_o(1)$ denotes o_1 , $\mathbf{V}_o(2)$ denotes o_2 , etc.

Let step_k be the current size of the variable var_i change. The i -th variable is given with:

$$\text{var}_i = (v_v^i, v_o^i), \quad (32)$$

where v_v^i is the translated value of v_e and v_o^i is the translated value of o_u (see (3)-(5)).

The increment of the var_i is carried out according to the presented algorithm:

- $v_v^i = v_v^i + step_k$
- if $v_v^i > 24$ then:
- $v_v^i = v_v^i - 24$
- $v_o^i = v_o^i + 1$
- if $v_o^i > 7$ then $v_v^i = 24 \wedge v_o^i = 7$

The decrement procedure is carried in a similar way, i.e.:

- $v_v^i = v_v^i - step_k$
- if $v_v^i < 1$ then:
- $v_v^i = v_v^i + 24$
- $v_o^i = v_o^i - 1$

if $v_o^i < 1$ then $v_v^i = 11 \wedge v_o^i = 0$

5.2.2. The „in-loop“ local search

The local search procedure implemented within the evolutionary loop has been implemented in a way that the computation time was not affected significantly.

First of all, there has been a strict limit of pattern searches number set. Secondly, the initial step size has been small. It allows for exploring only a small area around the base point given with following individuals.

6. Exemplary Results

For the efficiency of the technique presentation, an automated design of the specialized filter was realized. The assumed, desired amplitude response specifications for a filter are as follows ($A=7$ bands):

$$\left\{ \begin{array}{ll} |K_0| \in \langle -\infty, -20 \rangle \text{dB} & \wedge \quad f = \langle 1e-7, 0.1 \rangle \text{MHz} \\ |K_0| \in \langle -\infty, 0 \rangle \text{dB} & \wedge \quad f = \langle 0.1, 0.325 \rangle \text{MHz} \\ |K_0| \in \langle -3, 0 \rangle \text{dB} & \wedge \quad f = \langle 0.325, 0.375 \rangle \text{MHz} \\ |K_0| \in \langle -\infty, 0 \rangle \text{dB} & \wedge \quad f = \langle 0.375, 0.6 \rangle \text{MHz} \\ |K_0| \in \langle -30, -20 \rangle \text{dB} & \wedge \quad f = \langle 0.6, 0.7 \rangle \text{MHz} \\ |K_0| \in \langle -\infty, 0 \rangle \text{dB} & \wedge \quad f = \langle 0.7, 0.9 \rangle \text{MHz} \\ |K_0| \in \langle -\infty, -40 \rangle \text{dB} & \wedge \quad f = \langle 0.9, 1 \rangle \text{MHz} \end{array} \right. , \quad (33)$$

for an assumed load resistance $Z_0=R_0=150$. The MGP system was executed with the initial parameters:

- the number of generations $G_{mx}=50$,
- the maximal allowed depth for genotypes $D_{mx}=4$,
- the population size $N_{mx}=10$,
- the crossover probability $P_{cr}=0.9$,
- the mutations probabilities $P_{mu1}=0.2$, $P_{mu2}=0.3$, $P_{mu3}=0.1$, $P_{mu4}=0.2$,
- the minimal probability of reproduction $P_{mn}=0.3$,
- the number of frequency points $M=100$,

and for the orders (5) of values for searched elements:

$$\mathbf{O} = \{0, 1, \dots, 6\} . \quad (34)$$

The best found genotype evaluated for the specifications (33) is illustrated in Fig.11 and it codes the temporary version of the filter circuit placed in Fig.12a. Next, due to radically small or high values for some elements and due to the obtained kinds of connections (e.g. shortened or opened branches, the same types of elements connected in series or in parallel), unnecessary elements from this circuit can be removed or simplified. This stage of the circuit synthesis can be supported by a simulation software tool. In Fig.12b the final filter circuit obtained for the considered example is presented. In comparison to the temporary one, its size was radically reduced without amplitude response degradation.

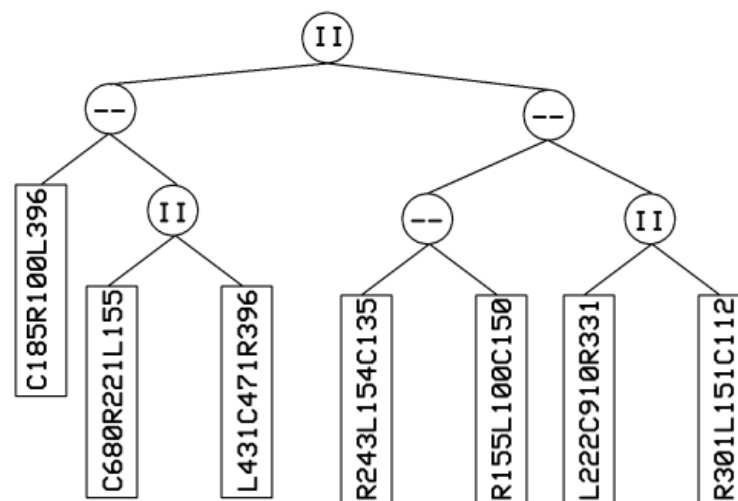


Figure 11. The resultant genotype

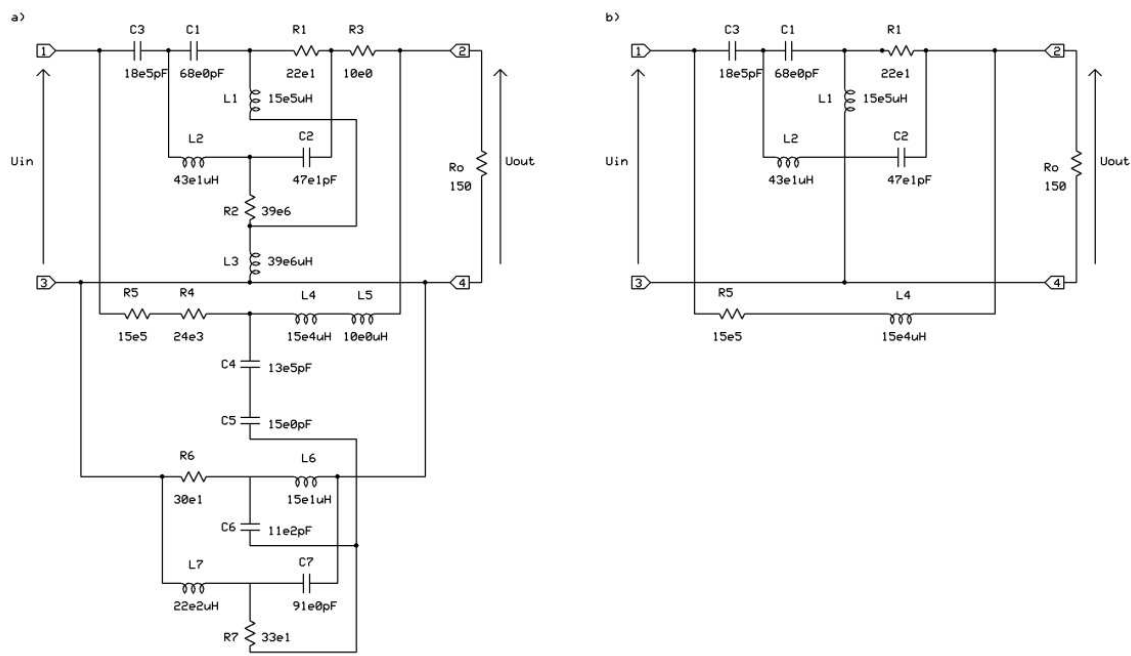


Figure 12. The structures of the designed filters: a) temporary version, b) final version

The quality of the specifications (33) keeping can be seen in Fig.13. The rectangles defining the allowed region for the filter amplitude response and simulation results obtained for the resultant circuit from Fig.12b are placed there. Only one restriction from (33) (the last frequency band) was a little violated, but the other ones are fully fulfilled. It should be emphasized, that due to discretization of the available values of components to a practical E24 series (4) not all theoretical shapes of frequency responses assumed on filter specifications defining stage will be practically reachable and it is necessary to conciliate with some inaccuracies. Besides, too high tolerance decreasing leads to the undesirable growing of production costs.

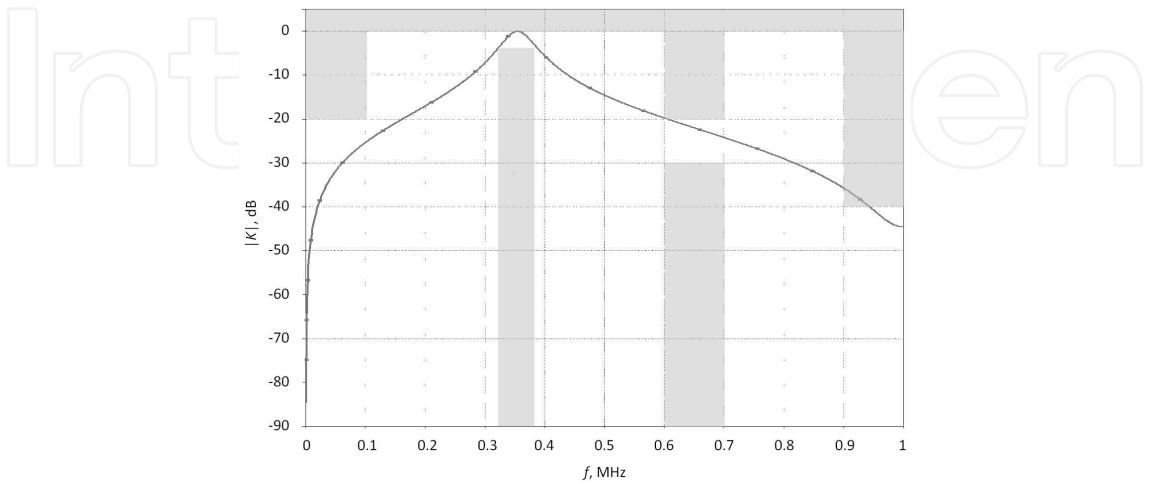


Figure 13. The amplitude response of the designed filter

7. Conclusions

The automated system for a passive filter circuits design was presented in this chapter. Any initial information about the filter circuit structure is not necessary for filter synthesis, only desired specifications should be defined. The circuit's topology as well as its elements values are optimized together in the MGP system. Thanks to the deterministic algorithm of the local searching engaging (HJM), the speed of convergence to the well evaluated solutions during the evolutionary computations grows significantly and the values of the filter's elements are adjusted to the most fitted ones for an actual circuit topology. Finally, after redundant elements elimination, the filter circuit is obtained with components selected from a practical production series (i.e. for practically reachable nominal values), and it makes the circuit realization easier. Besides, the genotypes sub-trees deletion applied in the system enables the complexity of the circuit minimization.

For the future system improving, the additional criteria for fitness calculation (28) can be added. For example, the group delay value and the kind of circuit elements (the quantity of inductors minimization) may be optimized. Besides, the applying of the models of real elements should be considered (to the parasitic parameters taken into account). Additionally, the proposed system can be adapted to the active circuits automated synthesis.

Author details

Tomasz Golonek* and Jantos Piotr

*Address all correspondence to: tgolonek@polsl.pl

Silesian University of Technology, Poland

References

- [1] Koza J. R., Genetic Programming: on the programming of computers by means of natural selection, MIT Press, 1992.
- [2] Banzhaf W., Nordin P., Keller R. E., Francone, F. D. Genetic Programming - An Introduction, Morgan Kaufmann Publishers Inc., San Francisco, California, 1998.
- [3] Goldberg D. E., Genetic Algorithms in Search, Optimization, and Machine Learning, Kluwer Academic Publishers, Boston, 1989.
- [4] Koza J. R., Bennett F. H., Andre D., Keane M. A., Dunlap F., "Automated Synthesis of Analog Electrical Circuits by Means of Genetic Programming", IEEE Transactions on Evolutionary Computation, Vol. 1, No. 2, 1997, 109-128.

- [5] Tathagato Rai Dastidar, P. P. Chakrabarti, Partha Ray, "A Synthesis System for Analog Circuits Based on Evolutionary Search and Topological Reuse", *IEEE Transactions on Evolutionary Computation*, Vol. 9, No. 2, 2005, 211-224.
- [6] Budzisz H., "Evolutional searching for circuit structures", *Electronic Letters*, Vol. 34, No. 16, 1998, 1543-1545.
- [7] Golonek T., Jantos P., Rutkowski J., Stimulus with limited band optimization for analogue circuit testing, *Metrology and Measurement Systems*, Vol. XIX, No. 1, pp. 73-84, 2012.
- [8] Jantos P., Rutkowski J., Evolutionary methods to analogue electronic circuits yield optimisation, *Bulletin of the Polish Academy of Sciences - Technical Sciences*, Vol. 56, Issue 1, 2008, pp. 9-16.
- [9] Golonek T., Rutkowski J., Genetic-Algorithm-Based Method for Optimal Analog Test Points Selection, *IEEE Trans. on Cir. and Syst.-II*, Vol.54, No.2, 2007, pp. 117-121.
- [10] Golonek T., Grzechca D., Rutkowski J., Application of Genetic Programming to Edge Decoder Design, *Proc. of the Inter. Symposium on Cir. and Sys.*, ISCAS 2006, Greece, 4683-4686.
- [11] Hooke R. , Jeeves T. A., Direct search solution of numerical and statistical problems, *Assoc. Computing Machinery J.*, Vol.8 , No.2, 1960, 212-229.
- [12] Moscato P., "On Evolution, Search, Optimization, Genetic Algorithms and Martial Arts: Towards Memetic Algorithms", *Caltech Concurrent Computation Program (report 826)*, 1989.
- [13] Land M. W. S., *Evolutionary Algorithms with Local Search for Combinatorial Optimization*, 1998.
- [14] Ozcan E., "Memes, Self-generation and Nurse Rostering". *Lecture Notes in Computer Science. Lecture Notes in Computer Science*, Springer-Verlag, 3867, 2007, 85-104.
- [15] Chen X., Ong Y. S., Lim M. H., Tan K. C., "A Multi-Facet Survey on Memetic Computation", *Evolutionary Computation*, *IEEE Transactions on* , vol.15, no.5, 2011, pp. 591-607.
- [16] Ong Y. S., Lim M. H., Zhu N., Wong K. W., "Classification of adaptive memetic algorithms: a comparative study," *Systems, Man, and Cybernetics, Part B: Cybernetics*, *IEEE Transactions on* , vol.36, no.1, 2006, pp. 141-152.