JJEE
Volume 10 | Number 1 | March 2024 | Pages 1-26

Jordan Journal of Electrical Engineering

ISSN (print): 2409-9600, ISSN (online): 2409-9619

Homepage: jjee.ttu.edu.jo

Analysis of the Circuit Optimization Process Based on a Generalized Approach and a Genetic Algorithm

Alexander Zemliak[1*], Andrei Osadchuk[2]

[1] Physics and Mathematics Department, Autonomous University of Puebla, Puebla, Mexico
E-mail: azemliak@yahoo.com
[2] The Arctic University Museum of Norway, UiT – The Arctic University of Norway, Tromsø, Norway

Received: March 18, 2023 *Revised: June 6, 2023* *Accepted: June 13, 2023*

***Abstract*—** Recently - based on generalized optimization - we developed an approach and successfully applied it to the problem of designing electronic circuits using deterministic optimization methods. In this paper, a similar approach is extended to the problem of optimizing electronic circuits using a genetic algorithm (GA) as the main optimization method. The fundamental element of the generalized optimization is an artificially introduced control vector that generates different strategies within the optimization process and determines the number of independent variables of the optimization problem, as well as the length and structure of chromosomes in the GA. In this case, the GA forms a set of populations defined by a fitness function specified in different ways depending on the strategy chosen within the framework of the idea of generalized optimization. The control vector allows generating different strategies, as well as building composite strategies that significantly increase the accuracy of the resulting solution. This, in turn, makes it possible to reduce both the number of generations - required during the operation of the GA - and the processor time by 3–5 orders of magnitude when solving the circuit optimization problem compared to the traditional GA. The performed analysis of the optimization procedure for some electronic circuits shows the effectiveness of this approach. The obtained results prove that the applied modification of the GA makes it possible to overcome premature convergence and increase the minimization accuracy by 3-4 orders of magnitude.

***Keywords*—** Circuit optimization; Genetic algorithm; Generalized optimization; Control vector.

1. INTRODUCTION

Designing a large system requires significant computational time to achieve an optimal solution to the design problem. The task of obtaining the optimal solution in the shortest possible time is especially important when designing VLSI circuits. Typically, the design process consists of circuit analysis for an initial approximation and continues until the system parameters are tuned to achieve the required performance as defined in the specification. The optimization procedure in this case acts as the main methodological technique for adjusting the parameters. Thus, instead of the difficult task of synthesizing a complex system, the design process can be implemented through analysis and optimization. Optimization procedures are a set of iterative algorithms that allow obtaining the required characteristics of the system being designed by minimizing some specially chosen objective function. In this case, the design procedure includes two main blocks: a block for analyzing the mathematical model of the circuit and an optimization procedure that minimizes the objective function. The minimum of this function makes it possible to obtain the required characteristics of the circuit. The interaction of these two blocks determines the circuit optimization process. In terms of this

* Corresponding author
Article's DOI: 10.5455/jjee.204-1679101785

approach, various circuit analysis methods and various optimization procedures can be used. Possible systems optimization methods can be divided into two main groups: deterministic optimization methods and stochastic search methods.

Deterministic algorithms developed both at the theoretical level and in the applied numerical aspect have found the widest application in optimizing electronic systems. Traditional numerical methods such as the gradient method, Newton's method, DFP method, etc. have been widely used in finding the minimum of functions, despite some convergence problems. These methods required the selection of a good initial approximation to perform adequately. In addition, they were most often used to search for a local minimum, and when searching for a global minimum, it was necessary to run the algorithm many times with different initial approximations. To overcome these problems, non-standard ideas have been developed. For example, to choose the initial approximation in works [1, 2], the idea of centering was used. In [3, 4], the geometric programming method was used, which to some extent ensured convergence to the global minimum. In [5, 6], the method of space mapping was used, which makes it possible to achieve a satisfactory solution. This technology was successfully used for optimization of microwave systems but there is no experience for solution of other problems. Another way to find the best solution was proposed in [7]. At the same time, based on the control theory and the idea of generalized optimization of electronic systems, a method was developed that improves the convergence of the optimization procedure and minimizes the processor time. However, in general, the problem of "falling" into local minima has not been overcome.

Stochastic heuristic methods also do not give an exact solution but can give a good approximation in a reasonable time, depending on the specific problem. Analysis of various stochastic optimization algorithms revealed several groups of methods: simulation annealing method [8-10], evolutionary computation methods producing different approaches as evolutionary algorithms [11-14], particle swarm optimization (PSO), GA, differential evolution, and genetic programming.

The PSO method is one of the evolutionary algorithms competing with genetic algorithms. The issue of using swarm algorithms has been worked out in an interesting review article [15]. Swarm intelligence algorithms [16-21] have significantly influenced the development of optimization procedures. In [16], a hybrid optimization algorithm based on particle swarm and annealing simulation is discussed. In a later work [17], swarm intelligence algorithms for solution of the task of layout of blocks of electronic equipment and VLSI planning are investigated and analyzed. As part of this work, experimental studies have shown the effectiveness of swarm algorithms for solving optimization problems in comparison with standard iterative, heuristic, and genetic algorithms. The high degree of stochasticity in the swarm search algorithms allows faster exit from local optima compared to the GA and the annealing algorithm.

Separately, we single out the GA, which is used to solve problems of nonlinear programming both for optimizing systems of various nature [22-29], and, in particular, for optimizing and designing electronic systems [30-34]. The genetic algorithm has been used as the main optimization procedure for analog circuits because of its potential to find an acceptable solution. The development of modifications of the genetic algorithm also leads to the improvement of optimization procedures. In [23] an algorithm is proposed that promotes the transfer of the best genes to chromosomes without losing the balance between selection and population diversity. This uses a new set of genetic operators that prevents premature convergence of the algorithm. In [29], the nonlinear robust control was reformulated as a constrained nonlinear optimization problem solved using a genetic algorithm. In the work [32], a modified GA for automated structural-parametric synthesis of a stepped directional coupler on coupled lines was developed. In [27], a modified Goldberg model is analyzed. The tournament selection method is used and possible options for obtaining descendants are investigated using the single-point crossover and mutation operators and some chromosomes obtained after selection. It is shown that the algorithm that compares the descendants after two main genetic operators and the parent chromosome after selection has the best characteristics. The best chromosome of these three is passed on to the next generation. The article [29] considers the possibility of reformulating the nonlinear robust control as a nonlinear conditional optimization solved by a genetic algorithm to analyze the optimal operation of a gas turbine. Special non-linear robust controllers are proposed for the studied gas turbine model, which allow obtaining a solution in an acceptable time with good convergence based on a GA. The paper [33] describes an automated method of structural-parametric synthesis of a microwave transistor amplifier based on a genetic algorithm. The problem described in this paper is as follows: to optimize the time of structural-parametric synthesis by applying a genetic algorithm.

One of the significant disadvantages of GA is the premature convergence of the method to local minima and, for this reason, an increase in processor time when a high accuracy of obtaining a solution is specified. To overcome this shortcoming, one can use the approach developed in the optimization of electronic circuits in the case of using deterministic optimization methods [7]. In this case a detailed analysis of the behaviour of various optimization strategies showed the possibility of a significant improvement in the characteristics of the process of optimizing electronic circuits, both in terms of accuracy and processor time. It would like to find out the legitimacy of this approach when solving problems of optimizing electronic circuits by GA. This assumption is the main driving motive of the present study. The work [34] presents an attempt to use the idea of a generalized approach in optimizing electronic circuits using a genetic algorithm. In contrast to this work, in the proposed article, the principle of choosing the best chromosome is similar to [27], but a four-point crossover is used. And most importantly, the idea of generalized optimization implemented in GA allows you to define a set of different optimization strategies and search for the best strategies among this set.

The rest of the paper is organised as follows. Section 2 discusses the main provisions of the generalized approach to solving nonlinear programming problems and its adaptation when using the genetic algorithm as the principal optimization method. Section 3 considers the solution of both abstract problems of nonlinear programming, which are test problems, and optimization problems of electronic circuits. The results obtained are analyzed and discussed in order to generalize and develop the main recommendations.

2. GENERALIZED APPROACH FOR THE GENETIC ALGORITHM

We define the optimization process of any analog system as the problem of minimizing the objective function $C(X)$ in the presence of a system of constraints, i.e. as a non-linear programming problem. In this case, the system of constraints is a mathematical model of the analog system given by the following equation:

$$g_j(X) = 0, \mathrm{j} = 1,2,...,M \quad (1)$$

where $X \epsilon R^N$.

We can divide the vector X in two parts: X' and X'' where the vector X' can be named as the vector of independent variables ($X' \epsilon R^K$), and the vector X'' as the vector of dependent variables ($X'' \epsilon R^M$), where K and M are the number of independent and dependent variables respectively and N=K+M.

Minimization of the objective function can be performed using a two-step optimization procedure defined by the vector equation:

$$X'^{s+1} = \Lambda(X'^{s}), \mathrm{s} = 1,2,... \quad (2)$$

where s is the number of iterations, Λ is the transition operator from step s to step s+1, and this operator depends on the objective function $C(X)$.

This is a typical statement of the problem of conditional optimization, which can be called the traditional optimization strategy (TOS). In this case, it is necessary to solve system (1) for each step of the optimization process. Besides, taking into account the specifics of the optimization process, it can be seen that there is no need to solve system (1) at each step of the optimization procedure. The fulfillment of these conditions is quite sufficient for the end point of the optimization process. In this case, Eqs. (1) and (2) are easily redefined in such a way that the difference between independent and dependent variables disappears. This is the main idea behind the application of the penalty function method. We can exclude the problem of solving a nonlinear system, since our task is not to analyze an electronic system, but to design it through optimization. This idea was put forward in the last century [35, 36]. This means that it is not necessary to satisfy Kirchhoff's laws at every step of the optimization procedure, but, of course, they must be fulfilled at the end point of the process, otherwise it is not clear which system was analyzed. The generalized objective function, involved in a certain way, just allows, on the one hand, to optimize the main objective function $C(X)$ and, in addition, ensures the fulfillment of Kirchhoff's laws at the end point of the optimization procedure, i.e. at the point of problem solving. This idea gives rise to another way of optimizing the system and can be called a modified traditional optimization strategy (MTOS). Later, this idea was generalized in the sense that it was proposed to exclude from the system of Eq. (1) not all equations, but some of them. Constraints (1) must be satisfied at the end point of the optimization procedure (2) for all strategies. At the same end point of the optimization process, the objective function should reach its minimum.

We applied the idea [7], which leads to a generalization of the optimization process. Let's declare all variables of the vector of dependent variables X'' to be independent. In this case, the constraint Eq. (1) can be excluded, but at the same time, to ensure the physical meaning of the problem and fulfill the required constraints at the end of the optimization procedure, it is necessary to involve some other objective function, namely the generalized objective function $F(X)$, defined by the following expression:

$$F(X) = C(X) + \varphi(X) \quad (3)$$

where $\varphi(X)$ is a special penalty function, which should equal zero at the end point of the optimization procedure. This ensures the fulfillment of conditions (1) at this point. The penalty function can be given by the following equation:

$$\varphi(X) = \sum_{j=1}^{M} g_j^2(X) \tag{4}$$

In this case we define MTOS. However, we can generalize this approach by declaring as independent variables a certain part of dependent variables, for instance Z arbitrary variables, where Z ϵ [0, M]. In this case, Eq. (4) consists of Z terms, and in this case it is required to exclude Z equations from system (1). This approach allows us to generalize the optimization problem by introducing an additional control vector U, which changes the structure of the main equations and thereby redistributes the processor costs between the circuit analysis block and the optimization block. It is this redistribution that creates the prerequisites for reducing the cost of CPU time when solving the problem of optimizing the system. The control vector $U=(u_1, u_2,\dots, u_M)$ is the mechanism that allows you to change the structure of the equations of the optimization process. System (1), in this case, can be given in the following form:

$$(1-u_j)g_j(X) = 0, j = 1,2,\dots, M \tag{5}$$

where u_j is the jth component of the control vector $U=(u_1, u_2,\dots, u_M)$, $u_j \in \Omega$, $\Omega=\{0;1\}$. Eqs. (3) and (4) are converted into the following Eqs. (6) and (7):

$$F(X,U) = C(X) + \varphi(X,U) \tag{6}$$

$$\varphi(X,U) = \frac{1}{\sigma}\sum_{j=1}^{M} u_j g_j^2(X) \tag{7}$$

where σ is an additional parameter and in our case is equal to 1.

The optimization process (2) includes the control vector too

$$X^{s+1} = \Lambda\left(X^s, U\right), \qquad s = 1, 2,\dots \tag{8}$$

Because it depends on the new objective function $F(X,U)$. The meaning of the control function u_j is defined as follows: in the case of $u_j = 0$, the equation with the number j remains in system (5), and the corresponding term $g_j^2(X)$ from the right side of Eq. (7) is removed, and vice versa, when $u_j = 1$, the equation j is removed from system (5), and the corresponding term $g_j^2(X)$ remains on the right side of Eq. (7).

In this case, the control vector U, as the main mechanism for modifying the GA, allows you to change the main system of constraint Eq. (5) and the structure of the function $F(X,U)$. In addition, if all components of the vector U have zero values, then TOS is determined. System (5) in this case coincides with system (1) and therefore must be analyzed at all steps of the optimization process. The function $F(X,U)$ coincides with the objective function $C(X)$, since the penalty function $\varphi(X,U)$ in this case is equal to zero.

Some of the equations of system (5) disappears when the corresponding component of the vector U is equal to 1 ($u_j = 1$). In this case, the information corresponding to this equation passes into the penalty function $\varphi(X,U)$ and into the generalized objective function $F(X,U)$. If all components of the vector U are equal to 1, then the optimization determines the MTOS. In this case the system (5) disappears, and the penalty function includes complete information about system (5). It is also necessary to make appropriate changes to the optimization procedure (2).

The optimization procedure based on deterministic methods can be given by differential Eq. (9) or difference Eq. (10):

$$\frac{dx_i}{dt} = f_i(X,U), i = 1,2,...,N \quad (9)$$

$$X^{s+1} = X^s + t_s H^s \quad (10)$$

where $f_i(X,U)$ or H are determined by a specific optimization method (gradient, Newton, etc.). Changing the value of the control vector component u_j from 0 to 1 leads to an increase in the number of independent variables and the number of equations in system (9), (10). In this case, the number of equations in the constraint system (5) decreases. The structure of the control vector U specifies various strategies and various trajectories in the parameter space within the framework of generalized optimization. In this case, the number of possible, alternative strategies becomes equal to 2^M, where M is the number of dependent variables, for example, the number of nodal voltages. An increase in the number of dependent variables M leads to an exponential increase in the number of basic strategies. In this case, the total number of possible strategies is not limited to the strategies of the structural basis, since a change in the current strategy is possible at any step of the optimization process. That is, it is possible to define compound strategies consisting of several different structural basis strategies. The switching point from one strategy to another will be denoted by Sp.

The system of Eqs. (5) to (10) includes a set of strategies determined by the components of the control vector. In work [7], it was shown that the control vector produces a set of strategies, among which there are strategies that implement the optimization process in significantly less processor time than TOS.

A generalized optimization approach can also be implemented if the optimization process is based on a stochastic method such as GA. In this case, Eq. (9) or (10) must be replaced by an optimization procedure based on the GA.

As mentioned, a modified Goldberg model is used. Chromosomes are selected by the tournament method. The two main genetic operators, the crossover operator and the mutation operator, are performed with a probability of 0.95 and 0.05. The operation of the algorithm with a four-point crossover operator is provided. We define NN as the number of chromosomes in a generation, and $\boldsymbol{X}$ is some matrix that includes N rows and NN columns, provided that each column corresponds to a certain value of the X vector. The fitness function in this case is given by the next formula.

$$P(\boldsymbol{X},U) = 1/F(\boldsymbol{X},U) \quad (11)$$

Given the concept of generalized optimization, it can be said that various modifications of the GA with a different structure of the fitness function can be implemented by changing the control vector U. In this case, the structure of the control vector determines and changes the structure of chromosomes both in the initial generation and during the operation of the algorithm.

For this stochastic algorithm, as in the case of the deterministic approach, it is possible to define a vector X, the components of which are determined by the average values over the generation:

$$x_i = \frac{1}{NN}\sum_{j=1}^{NN} x_{ij} \quad (12)$$

where x_{ij} is an element of the matrix $\boldsymbol{X}$.

3. RESULTS AND DISCUSSION

3.1. Some details of the algorithm

For the examples under consideration, the limiting values for changing the parameters of the GA were determined. Chromosome length (*L*) ranged from 20 to 60 for each task variable. The number of chromosomes (*NN*) in each generation ranged from 100 to 400.

In classical GA, in addition to an arbitrary assignment of the number of chromosomes in a generation, the choice of various methods for selecting chromosomes, and various methods for setting the probabilities for the implementation of the main genetic operators of crossover and mutation, the probabilistic characteristics are set in software implementation using random number generators (RNG). In our case, the RNG of the C++ language rand() was used, as well as the srand() operator to generate new sequences of random numbers. In this case, the initial sequences of chromosomes were randomly formed, as well as the point of crossover of the chromosomes of the parent pair and the point of a possible mutation. The number of trials for each studied strategy ranged from 40 to 60.

The first two examples are purely test ones, the solution for which is known. These examples serve to demonstrate the approach formulated in the previous section.

3.2. Example 1

Minimize $C(X)$

$$C(X) = 2x_1^2 - 3x_2^2 - 2x_1 \tag{13}$$

subject to:

$$(x_1 - 3)^2 + (x_2 - 2)^2 = 0 \tag{14}$$

In this example, there is only one independent variable x_1 (K=1), and parameter M=1 because there is only one constraint Eq. (14). Let's define variable x_2 as dependent that can be calculated from Eq. (14).

In fact, there is an analytical solution to this problem. Indeed, the fulfillment of the necessary constraint (14) is ensured by the solution of this equation and is achieved at the point x_1 = 3, x_2 = 2. In this point, the goal function $C(X)$ takes the minimum zero value. These values are the solution to the problem. Let us find, however, this solution in accordance with the developed approach.

Based on the generalized approach, Eq. (14) is transformed into the following equation:

$$(1-u)\left((x_1 - 3)^2 + (x_2 - 2)^2\right) = 0 \tag{15}$$

where u is the component of the control vector U, in this case the only one.

Consider two main strategies: TOS which has a control vector U=(0) and MTOS which has a control vector U=(1).

Here, we analyse the results of optimization by means of a GA for these strategies. However, it was shown that in the case of a deterministic optimization process, a combination of several strategies can reduce both the number of steps of the optimization process and the computation time.

Table 1 shows the dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ of minimizing the objective function *F* for three strategies: TOS, MTOS and composite strategy (0)(1) with an optimal switching point Sp from strategies (0) to strategy (1). The optimal switch point Sp improves the characteristics of the composite strategy, but in this paper, it was obtained manually.

The optimal value of the switching point Sp was obtained by additional analysis. This value, as can be seen from the table, depends on the required precision δ. It is clear that when using the TOS, the number of generations and CPU time is less than for MTOS up to a certain level of precision (10^{-4}).

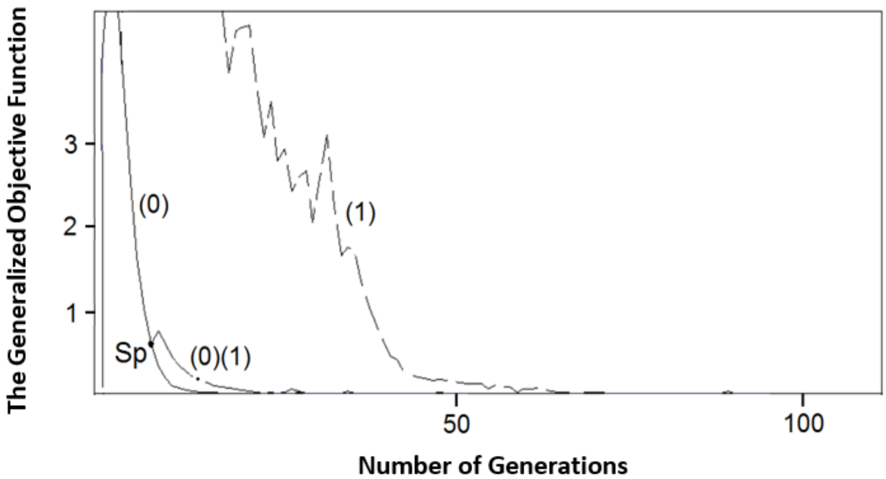
Table 1. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ of function *F* for three strategies: TOS, MTOS and composite strategy (0)(1) with the optimal switching point Sp.

Precision δ	Number of generations (Processor time (s))		
	Control vector (0)	Control vector (1)	Control vector (0)(1)
10^{-1}	11 (0.048)	56 (0.125)	16, Sp =10 (0.044)
10^{-2}	18 (0.091)	69 (0.143)	24, Sp = 9 (0.059)
10^{-3}	23 (0.073)	79 (0.162)	27, Sp = 10 (0.066)
10^{-4}	37 (0.113)	91 (0.18)	42, Sp = 13 (0.095)
10^{-5}	32883 (100.217)	96 (0.19)	53, Sp = 9 (0.113)
10^{-6}	-	104 (0.206)	66, Sp = 14 (0.146)
10^{-7}	-	114 (0.261)	75, Sp = 6 (0.176)
10^{-8}	-	121 (0.271)	81, Sp = 14 (0.179)
10^{-9}	-	129 (0.275)	104, Sp = 15 (0.227)
10^{-10}	-	134 (0.278)	104, Sp = 15 (0.226)
10^{-11}	-	1368 (2.906)	114, Sp = 15 (0.247)
5×10^{-12}	-	1368 (2.906)	117, Sp = 10 (0.254)
4×10^{-12}	-	-	120, Sp = 10 (0.256)
10^{-12}	-	-	121, Sp = 10 (0.266)
5×10^{-13}	-	-	300, Sp = 31 (0.603)
10^{-13}	-	-	989, Sp = 5 (1.928)
2.65×10^{-14}	-	-	989, Sp = 5 (1.928)

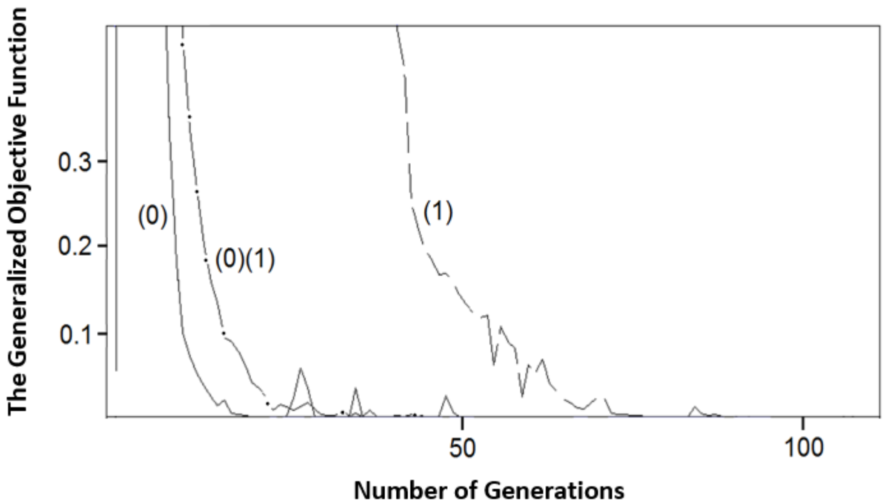
The TOS allows finding a solution up to the error level of 10^{-5}, but the number of generations increases dramatically. At the same time, this strategy cannot find a solution with higher accuracy. The MTOS with control vector (1) finds a solution with a much higher accuracy up to 5×10^{-12}.

At the same time a composite strategy consisting of two, (0) and (1) with an optimal switching point between them, gives a solution with an accuracy of 2.65×10^{-14} and, importantly, with a smaller number of generations.

Fig. 1 shows the dependence of the generalized objective function *F* under successive generational change for strategies (0), (1) and composite strategy (0), (1) for two scales: scale 1 and scale 2.



(a)



(b)

Fig. 1. Dependence of the generalized objective function *F* under successive generational change for strategies (0), (1) and composite strategy (0)(1) for: a) scale 1; b) scale 2.

It is clear that the best strategy for minimizing the fitness function is the composite strategy (0)(1), which, in the case of the optimal switching point Sp, solves the problem in the best way compared to other strategies.

Variables x_1 and x_2 take the values 3 and 2, respectively, but with different degrees of accuracy for different strategies.

3.3. Example 2

Minimize $C(X)$

$$C(X) = (x_3 - 0.15)^2 \tag{16}$$

subject to:

$$x_1 - x_2 + 2x_3 - 6 = 0$$
$$x_1 - 2x_2 - 8 = 0 \tag{17}$$

In this case, M=2, that is, system (17) is determined by two dependent variables, and the third is an independent parameter. We define x_1 as an independent parameter. In this case, x_2 and x_3 are dependent.

This test problem also has an analytical solution. It can be seen that the objective function, being non-negatively defined, reaches the minimum, zero value at the point x_3 = 0.15. In this case, to fulfill the restrictions (17), the variables x_1 and x_2 take the following values: x_1=3.4, x_2= - 2.3. Let us find a solution to the problem in accordance with the developed approach. Using the generalized optimization approach, system (17) is transformed into the following system:

$$(1 - u_1)(x_1 - x_2 + 2x_3 - 6) = 0$$
$$(1 - u_2)(x_1 - 2x_2 - 8) = 0 \tag{18}$$

The control vector for this example has two components: $U = (u_1, u_2)$. Table 2 shows the dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ of minimizing the objective function F for three strategies: TOS, MTOS and composite strategy (00)(11) with optimal switching point Sp between strategies (00) and (11).

The traditional strategy requires much more generations than modified or composite strategies while obtaining the same precision.

Analyzing the results in the table, one can see that TOS can find a solution with a precision of 10^{-3} and no higher. At the same time, the MTOS with the control vector (11) and the composite strategy with the control vector (00)(11) make it possible to find a solution with a precision of 5×10^{-5} and 3×10^{-8}, respectively. It can be seen that MTOS with U = (11) and a combined strategy with U = (00) (11) find a solution for a smaller number of generations and a smaller processor time than TOS with U=(00). We see that MTOS and the composite strategy solve an optimization problem with two orders of magnitude fewer generations than TOS for 10^{-2} precision and four orders of magnitude less for 5×10^{-3} precision.

TOS solves the optimization problem in 5.959 s with a precision of 10^{-2} and 240.53 s with a precision of 5×10^{-3}. The composite strategy solves this problem in 0.047 s with an accuracy of 10^{-2} and 0.05 s with an accuracy of 5×10^{-3}. In this case the CPU time gain is 126 times for 10^{-2} precision and 4810 times for 5×10^{-3} precision.

The dependence of the generalized objective function F is shown in Fig. 2 under successive change of generations for strategies (00), (11) and composite strategy (00)(11).

It can be seen that for the three presented strategies, different behavior of the function F is observed. MTOS and the composite strategy provide a large gain in generation's number and CPU time to ensure the desired precision.

Table 2. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ of function *F* for three strategies: TOS, MTOS and composite strategy (00)(11) with the optimal switching point Sp.

Precision δ	Number of generations (Processor time (s))		
	Control vector (00)	Control vector (11)	Control vector (00)(11)
4×10^{-2}	29 (0.043)	22 (0.06)	14, Sp =2 (0.038)
2×10^{-2}	1017 (1.472)	25 (0.067)	16, Sp = 2 (0.043)
10^{-2}	4118 (5.959)	25 (0.067)	18, Sp = 2 (0.047)
5×10^{-3}	165741 (240.53)	27 (0.07)	19, Sp = 2 (0.05)
10^{-3}	-	32 (0.085)	21, Sp = 2 (0.063)
10^{-4}	-	39 (0.107)	37, Sp = 2 (0.105)
5×10^{-5}	-	69 (0.186)	45, Sp = 5 (0.124)
10^{-5}	-	-	51, Sp = 16 (0.142)
5×10^{-6}	-	-	52, Sp = 16 (0.147)
10^{-6}	-	-	75, Sp = 27 (0.207)
10^{-7}	-	-	82, Sp = 27 (0.226)
3×10^{-8}	-	-	149, Sp = 27 (0.416)

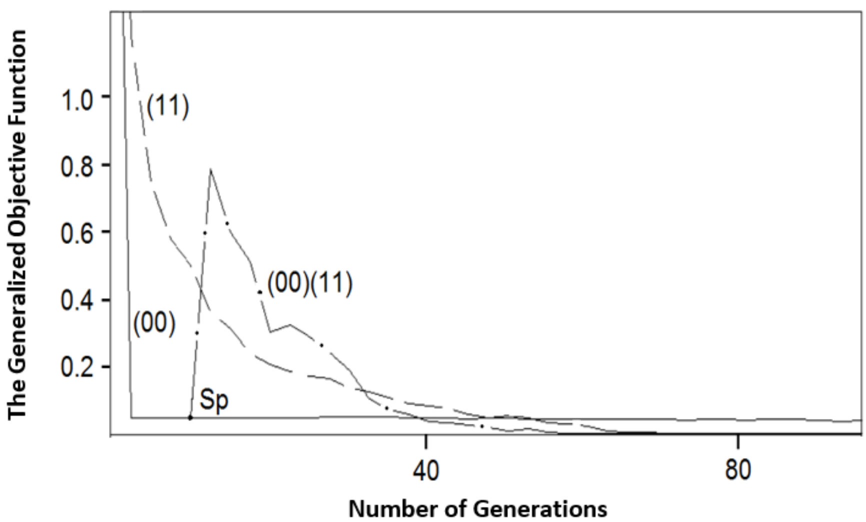


Fig. 2. Dependence of the function *F* under successive generational change for strategies (00), (11) and composite strategy (00)(11).

Variables x_1, x_2 and x_3 take values of 3.4, -2.3 and 0.15, respectively, but with different degrees of accuracy for the three studied strategies.

3.4. Example 3

This example analyzes the process of optimizing a function for one of the reference problems - finding the global minimum of the modified Shekel function. This function is given by the following equation:

$$C(X) = -\sum_{i=1}^{m}\left(\sum_{j=1}^{N}\left(x_j - a_{ij}\right)^2 + c_i\right)^{-1} + c_0 \tag{19}$$

where m is the number of possible minima of the function, N is the total number of variables, a_{ij} are the coordinates of possible minima, c_i are the coefficients that determine the values of possible minima. There is no coefficient c_0 in the standard definition of the Shekel function. Such assignment of the Shekel function is typical for the problem of unconstrained optimization. Possible minima of function (19) are located in the negative area and the global minimum corresponds to the deepest dip. Let us define the following coefficients in Eq. (19): $N = 2$, $m = 5$. For this example, the Shekel function depends on two variables x_1 and x_2, and is defined by five possible minima given by the following coordinates: $a_{11} = 1.10$, $a_{12} = 0.0316$, $a_{21} = 2.0$, $a_{22} = 1.0$, $a_{31} = 3.0$, $a_{32} = 2.828427$, $a_{41} = 3.5$, $a_{42} = 3.952847$, $a_{51} = 4.0$, $a_{52} = 5.196$. Each pair of coefficients determines the coordinates of the minima. The values of the minima correspond to the coefficients c_1, c_2, c_3, c_4 and c_5, which are defined below. Since the optimization problem is being solved in the presence of constrains, we set constrains in the following form:

$$\left(x_1 - 1\right)^3 - x_2^2 = 0 \tag{20}$$

$$x_1 \geq 0,\ \ x_2 \geq 0 \tag{21}$$

Eq. (20) is a relationship equation between variables, being a model of some system and when an independent variable x_1 is specified, the dependent variable x_2 is uniquely determined.

A feature of optimizing an electronic circuit and applying a generalized approach is that the objective function can be set to be non-negative and its global minimum, therefore, has a value of 0. In this case, some modification of the Shekel function is required, which consists in adding the coefficient c_0 in Eq. (19), which is equal to the absolute value of the global minimum. In this case, the entire function "rises" by the value of the global minimum and is non-negative.

The presence of one independent variable and one dependent correspond to K=1, M=1. Using a generalized approach to optimization, Eq. (20) is transformed into the following equation:

$$\left(1 - u\right)\left(\left(x_1 - 1\right)^3 - x_2^2\right) = 0 \tag{22}$$

In this case, only two main strategies TOS and MTOS and possible compound strategies can be defined.

Numerical analysis of the Shekel function (19) for given coefficients and $c_0 = 0$ made it possible to reveal the presence of four minima, one of which is global, at the points corresponding to the first four pairs of coefficients a_{ij}. Let us consider three variants of the distribution of the minima of the Shekel function.

3.4.1. Option 1

The minima correspond to the following coefficients: c_1 = 0.1, c_2 = 0.2, c_3 = 0.3, c_4 = 0.2, c_5 = 0.3. The values of the minima are as follows: C_{min1} = -10.6454, C_{min2} = - 5.8458, C_{min3} = - 4.2235 and C_{min4} = -5.6889. The first minimum is global and corresponds to the coordinates: x_1 = 1.1, x_2 = 0.0316. The coefficient c_0 in Eq. (19) is taken equal to 10.6454.

Function optimization results (19) under constraints (20)-(21) for TOS, MTOS and composite strategies that includes two main strategies with a control vector (0)(1) are given in Table 3, Fig. 3 and Fig. 4. The traditional TOS strategy comes to a local minimum with *F*=4.75 and coordinates x_1 = 2.0, x_2 = 1.0. That is, we can state that this strategy does not find a solution to the problem. At the same time, the MTOS and composite strategy find a global minimum equal to zero with coordinates x_1 = 1.10, x_2 = 0.0316. Table 3 shows the results of the optimization process for different accuracy δ of minimizing the objective function *F* for MTOS and a composite strategy with control vector (0)(1) and switching point Sp =3. A comparison of these strategies shows a slight advantage of the composite strategy while increasing the required accuracy of solving the problem.

Table 3. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ of function *F* for two strategies: MTOS and composite strategy (0)(1) with the optimal switching point Sp=3.

Precision *δ*	Number of generations (Processor time (s))	
	Control vector (1)	Control vector (0)(1)
10^{-1}	22 (0.05)	24 (0.058)
10^{-2}	38 (0.086)	38 (0.087)
10^{-3}	45 (0.10)	42 (0.098)
10^{-4}	65 (0.144)	55 (0.128)
10^{-5}	79 (0.174)	56 (0.131)
10^{-6}	80 (0.18)	58 (0.136)
10^{-7}	91 (0.202)	79 (0.175)
10^{-8}	909 (2.014)	812 (1.802)
3×10^{-9}	-	37659 (83.573)

Fig. 3 shows the trajectories of the optimization process, including two components x_1 and x_2 of the vector *X*, calculated by Eq. (12) for three strategies, TOS, MTOS and a composite strategy with a control vector (0)(1).

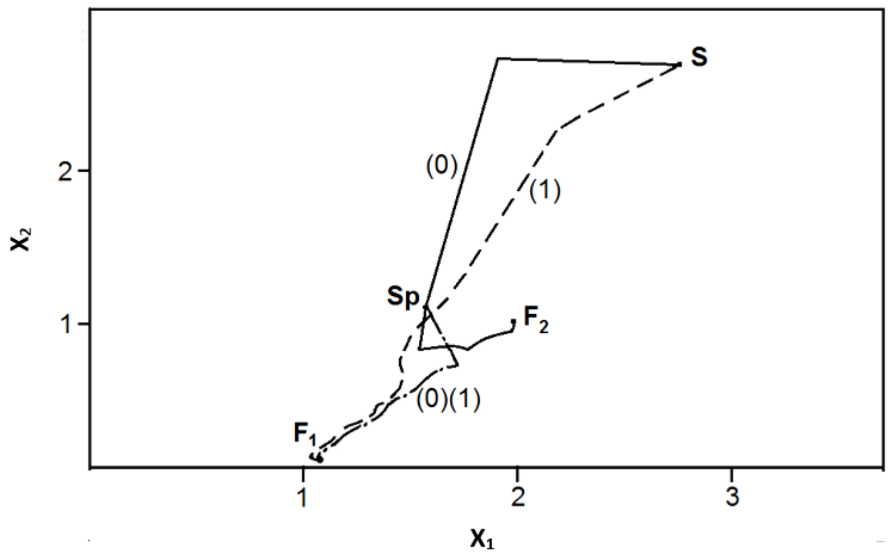


Fig. 3. Trajectories of the optimization process for three strategies (0), (1) and composite strategy (0)(1).

Point S corresponds to the starting point of the optimization process, F_1 is the final point of the optimization process, corresponding to MTOS and the composite strategy (0)(1) and is the global minimum point, F_2 is the final point of the optimization process, corresponding to TOS and being one of the local minima. Sp is the switching point from strategy (0) to strategy (1). It is important to emphasize that the TOS corresponding to the control vector (0) has a "hard" trajectory in the sense that condition (20) must always be satisfied on this trajectory. At the same time, the other two strategies work under the conditions of two independent variables x_1 and x_2, and condition (20) may not be satisfied on the entire trajectory, except for the final point. In this sense, these two strategies are more stochastic, which ultimately leads to the possibility of "skip past" local minima and find a global one.

The dependence of the generalized objective function *F* on the number of generations is shown in Fig. 4 for three strategies TOS, MTOS and a composite strategy with a control vector (0)(1) for an accuracy of $\delta=10^{-5}$. Sp is the switching point from one strategy to another.

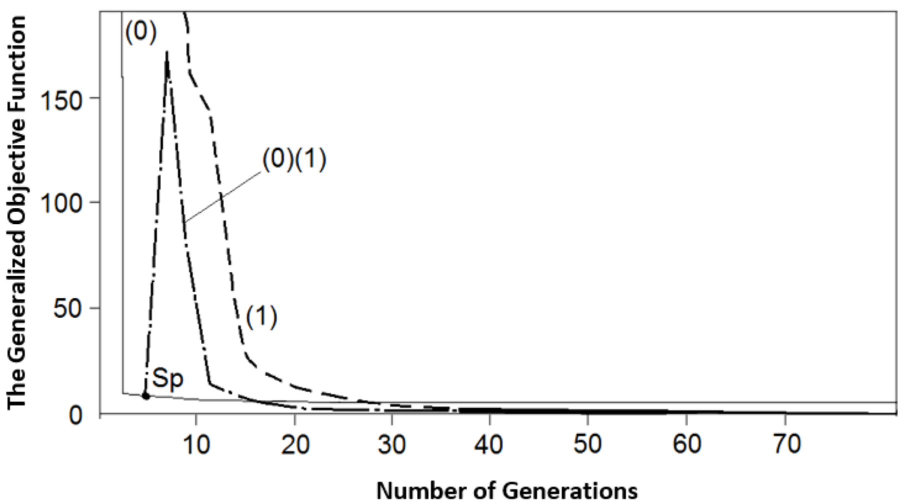


Fig. 4. Dependence of the function *F* under successive generational change for strategies (0), (1) and composite strategy (0)(1).

The function *F* for TOS decreases to 4.75 and then does not change, which corresponds to a local minimum. At the same time, for the other two strategies, the function *F* decreases to the values 10^{-8}–10^{-9} that gives a high accuracy of the optimization process implementation, since it corresponds to the global minimum.

3.4.2. *Option 2*

The minima correspond to the following coefficients: c_1 = 0.15, c_2 = 0.1, c_3 = 0.3, c_4 = 0.2, c_5 = 0.3. The values of the minima are as follows: C_{min1} = -7.3399, C_{min2} = - 10.8316, C_{min3} = - 4.2280 and C_{min4} = -5.6896. The second minimum is global and corresponds to the coordinates: x_1=2.0, x_2=1.0. The coefficient c_0 in Eq. (19) was set equal to 10.8316.

Optimization results of function (19) under constraints (20)-(21) for three strategies TOS, MTOS and a composite one that includes two main strategies with a control vector (0)(1) are given in Table 4.

Table 4. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ of function *F* for three strategies: TOS, MTOS and composite strategy (0)(1) with the optimal switching point Sp=1.

Precision δ	Number of generations (Processor time (s)) Control vector (0)	Control vector (1)	Control vector (0)(1)
10^{-2}	26 (0.041)	32 (0.073)	43 (0.098)
10^{-3}	31 (0.048)	60 (0.137)	76 (0.174)
10^{-4}	37 (0.058)	97 (0.222)	79, (0.181)
10^{-5}	350 (0.549)	99 (0.227)	85 (0.194)
10^{-6}	1212 (1.903)	949 (2.173)	201 (0.460)
3×10^{-7}	-	9179 (21.020)	666 (1.525)
10^{-7}	-	-	8998 (20.605)
2×10^{-8}	-	-	13457 (30.816)

All three strategies find the global minimum corresponding to the point with coordinates x_1 = 2.0, x_2 = 1.0, however, the accuracy of finding this minimum is different for these strategies. TOS finds the minimum with a marginal accuracy of 10^{-6}, MTOS with an accuracy of 3×10^{-7} and a compound strategy with an accuracy of 2×10^{-8}.

3.4.3. *Option 3*

The minima correspond to the following coefficients: c_1 = 0.2, c_2 = 0.1, c_3 = 0.07, c_4 = 0.15, c_5 = 0.3. The values of the minima are as follows: C_{min1} = -5.6751, C_{min2} = -10.8296, C_{min3} = -15.1971 and C_{min4} = -7.4365. The third minimum is global and corresponds to the coordinates: x_1=3.0, x_2=2.828. The coefficient c_0 in Eq. (19) was set equal to 15.1971.

Optimization results of function (19) under constraints (20)-(21) for two strategies MTOS and a composite one that includes two main strategies with a control vector (0)(1) are given in Table 5.

Table 5. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ for two strategies: MTOS and composite strategy (0)(1) with the optimal switching point Sp=1.

Precision δ	Number of generations (Processor time (s))	
	Control vector (1)	Control vector (0)(1)
10^{-1}	33 (0.075)	32 (0.073)
10^{-2}	52 (0.119)	48 (0.110)
10^{-3}	61 (0.140)	61 (0.140)
10^{-4}	66 (0.151)	79 (0.181)
10^{-5}	73 (0.167)	79 (0.181)
5×10^{-6}	6655 (15.240)	81 (0.185)
4×10^{-6}	94449 (216.288)	82 (0.186)
10^{-6}	-	366 (0.838)
2×10^{-7}	-	29672 (67.949)

In this case, as well as in the first variant, the traditional strategy does not find a global minimum, but stops in a local minimum with coordinates x_1=2.0, x_2=1.0. MTOS and the composite strategy find the global minimum corresponding to the point with coordinates x_1= 3.0, x_2 = 2.828. At the same time, the composite strategy finds a minimum with a maximum accuracy of 2×10^{-7}, which is an order of magnitude better than the MTOS strategy.

The analysis of this example allows us to understand the specifics of optimizing a multi-extremal function in the presence of restrictions. In this case, the use of the traditional strategy does not always allow one to find the global minimum, since the process can loop in local minima. At the same time, some strategies emerging from the generalized approach can overcome this problem and find the global minimum with a high degree of accuracy.

3.5. Example 4

Let us optimize the circuit of a four-node nonlinear voltage divider shown in Fig. 5. The conductivities y_1, y_2, y_3, y_4, y_5 are positive and represent a set of parameters for a given circuit (K=5) that are defined as independent. Voltages in circuit nodes V_1, V_2, V_3, V_4 are dependent parameters (M=4). The aim of circuit optimization is to obtain the required values of all nodal voltages $V_{10}, V_{20}, V_{30}, V_{40}$ by selecting conductivities.

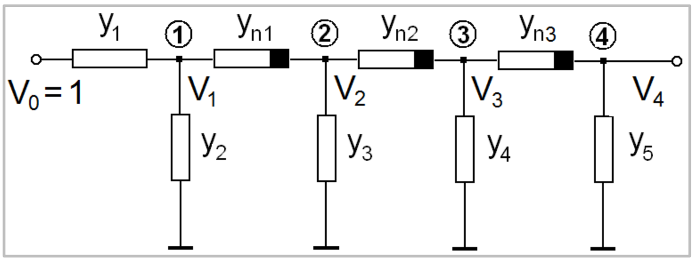


Fig. 5. Four-node nonlinear passive circuit.

Given that the voltage at the input of the divider is 1 V, these constants in the normalized form have the following values: V_{10}=0.7, V_{20}=0.4, V_{30}=0.2, V_{40}=0.1.

In mathematical terms, this problem can be represented as a problem of minimizing some objective function. Let us define the objective function of the optimization process by means of the following equation:

$$C(X) = \sum_{i=1}^{M} \left[(V_i - V_{i0})^2 \right] \tag{23}$$

The mathematical model of the circuit in this case acts as a set of restrictions. Let's define non-linear elements by the following expressions: $y_{n1} = a_{n1} + b_{n1} \cdot (V_1 - V_2)^2$, $y_{n2} = a_{n2} + b_{n2} \cdot (V_2 - V_3)^2$ and $y_{n3} = a_{n3} + b_{n3} \cdot (V_3 - V_4)^2$, where $a_{n1} = a_{n2} = a_{n3} = 1$, and $b_{n1} = b_{n2} = b_{n2} = 0.9$. Vector X includes nine components $(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9)$, where: $x_1^2 = y_1$, $x_2^2 = y_2$, $x_3^2 = y_3$, $x_4^2 = y_4$, $x_5^2 = y_5$, $x_6 = V_1$, $x_7 = V_2$, $x_8 = V_3$ and $x_9 = V_4$. These equations for the components x_1, x_2, x_3, x_4, x_5 always make it possible to obtain positive conductivities. This removes the problem of the mandatory positive definiteness of each component of the vector X. The first five components of this vector can have both positive and negative values. In this case, the conductivities are always positive.

Eq. (23) is transformed into the following form:

$$C(X) = \sum_{i=1}^{M} \left[(x_{K+i} - V_{i0})^2 \right] \tag{24}$$

Considering the Kirchhoff laws, the mathematical model of the circuit can be represented by four equations of the nodal voltage method, and the functions $g_j(X)$ are given using the following equations:

$$\begin{aligned}
&g_1(X) \equiv -x_1^2 + \left(x_1^2 + x_2^2\right)x_6 + \left\{a_{n1} + b_{n1}(x_6 - x_7)^2\right\}(x_6 - x_7) = 0 \\
&g_2(X) \equiv x_3^2 x_7 + \left\{a_{n1} + b_{n1}(x_6 - x_7)^2\right\}(x_7 - x_6) + \left\{a_{n2} + b_{n2}(x_7 - x_8)^2\right\}(x_7 - x_8) = 0 \\
&g_3(X) \equiv x_4^2 x_8 + \left\{a_{n2} + b_{n2}(x_7 - x_8)^2\right\}(x_8 - x_7) + \left\{a_{n3} + b_{n3}(x_8 - x_9)^2\right\}(x_8 - x_9) = 0 \\
&g_4(X) \equiv x_5^2 x_9 + \left\{a_{n3} + b_{n3}(x_8 - x_9)^2\right\}(x_9 - x_8) = 0
\end{aligned} \tag{25}$$

Therefore, we must minimize the function $C(X)$ given by expression (24) with additional conditions (25). The control vector U has four components: $U = (u_1, u_2, u_3, u_4)$.

Applying Eqs. (6) and (7), gives the following equation for the generalized objective function F:

$$F(X,U)=C(X)+\frac{1}{\sigma}\sum_{j=1}^{4}u_j g_j^2(X) \tag{26}$$

The number of structural basis strategies is quite large and equals 16. Of course, there are a large number of possible combinations of different strategies, but, as was shown in [7], when using deterministic optimization methods, the best results should be expected from a combination of TOS and MTOS strategies with the control vector (00…0) and (11…1).

Table 6 shows dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ for three strategies: TOS, MTOS and composite strategy (0000)(1111) with the optimal switching point Sp=6 between strategies (0000) and (1111).

Table 6. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ for three strategies: TOS, MTOS and composite strategy (0000)(1111) with the optimal switching point Sp=6.

Precision δ	Number of generations (Processor time (s)) Control vector (0000)	Control vector (1111)	Control vector (0000)(1111)
4×10^{-3}	77 (0.298)	72 (0.074)	68 (0.087)
5×10^{-4}	80 (0.31)	77 (0.079)	69 (0.088)
3.77×10^{-4}	107 (0.414)	81 (0.083)	70 (0.089)
3.765×10^{-4}	176809 (684.25)	81 (0.083)	70 (0.089)
3.76×10^{-4}	-	81 (0.083)	70 (0.089)
3×10^{-4}	-	84 (0.086)	72 (0.091)
10^{-4}	-	93 (0.096)	75 (0.094)
10^{-5}	-	111 (0.114)	82 (0.101)
10^{-6}	-	126 (0.13)	84 (0.104)
2×10^{-7}	-	148 (0.152)	86 (0.106)
10^{-7}	-	-	88 (0.108)
4×10^{-8}	-	-	164 (0.186)

It can be stated that the use of MTOS and the composite strategy makes it possible to obtain a significant gain compared to TOS both in terms of the number of generations and processor time to achieve an accuracy of 3.765×10^{-4}. It should be noted that this is the ultimate accuracy that a traditional optimization strategy can achieve.

MTOS with a control vector (1111) and a composite strategy with a control vector (0000)(1111) have an advantage over TOS of more than 2000 times in the number of generations and more than 8000 times in processor time. TOS does not find a solution if the required error is reduced to a value less than 3.765 10^{-4}. In contrast, MTOS and the composite strategy find solutions up to a precision of $2×10^{-7}$ or $4×10^{-8}$ for the first and second strategies, respectively. The number of GA generations as a function of the position of the switching point Sp for the composite strategy (0000)(1111) for accuracy $\delta = 10^{-5}$ is presented in Table 7.

Table 7. Number of generations as a function of the switching point Sp of the composite strategy (0000)(1111).

Switch point Sp	4	5	6	7	8	9	10	11
Number of generations G	98	85	82	84	89	112	109	119

The optimal value of the switching point between strategies Sp = 6. That is, the strategy with the control vector (0000) works for the first five steps, and the subsequent ones with the vector (1111). The dependences of the generalized objective function *F* on the successive change of generations for the strategies with the control vector (0000), (1111) and the composite strategy (0000)(1111) with a given error $\delta = 2×10^{-7}$ are shown in Fig. 6.

Fig. 6 shows the dependence of the generalized objective function *F* under successive generational change for strategies with the control vector (0000), (1111), and composite strategy (0000)(1111) for a given error $\delta = 2×10^{-7}$.

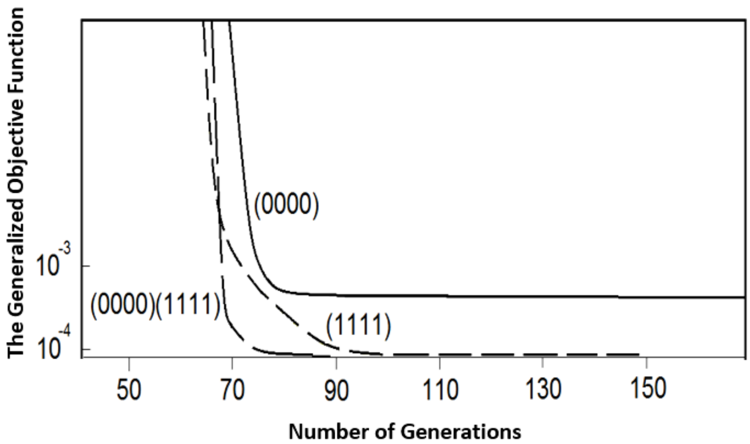


Fig. 6. Dependence of the generalized objective function *F* under successive generational change for strategies (0000), (1111) and composite strategy (0000)(1111).

It can be seen from the figure that TOS does not provide good accuracy of the solution, unlike MTOS and the composite strategy. Conversely, the MTOS and the composite strategy give a solution of the problem with high accuracy ($2×10^{-7}$) in a relatively small number of generations. It is important to emphasize that TOS cannot solve the problem with such accuracy in a foreseeable period.

A new population with different properties is formed for a composite strategy at the switching point Sp. At this point, the population structure changes drastically, and the optimization process leaves the local minimum trap. For this reason, this strategy achieves the minimum of the objective function with greater precision than other strategies.

3.6. Example 5

Let us analyze the procedure for optimizing the two-cascade transistor amplifier shown in Fig. 7.

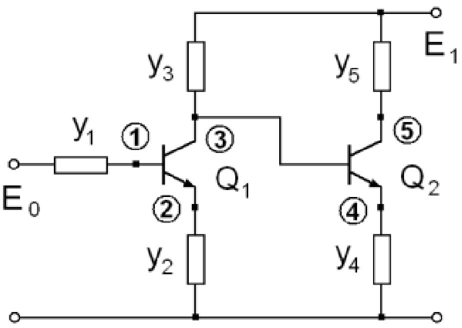


Fig. 7. The two-cascade amplifier.

In this case, we define five variables as independent: y_1, y_2, y_3, y_4 and y_5 (K=5) and five variables as dependent: V_1, V_2, V_3, V_4 and V_5 (M=5).

We can define all ten components of the vector X by the next equations: $x_1^2=y_1$, $x_2^2=y_2$, $x_3^2=y_3$, $x_4^2=y_4$, $x_5^2=y_5$, $x_6=V_1$, $x_7=V_2$, $x_8=V_3$, $x_9=V_4$, $x_{10}=V_5$. The transistor model is approximated by the static Ebers-Moll model [37]. Since voltages across transistor junctions are important parameters, the cost function $C(X)$ is now determined by the squares of the differences between the calculated values of the voltages and previously specified values for all transistor junctions using the following equation:

$$C(X)=\sum_{i=1}^{2}\left[(V_{EBi}-V_{EB0i})^2+(V_{CBi}-V_{CB0i})^2\right] \tag{27}$$

where V_{EBi} and V_{CBi} are the calculated voltages at the emitter and collector junctions and V_{EB0i} and V_{CB0i} are the specified voltages at these junctions. We define these values as follows: V_{EB01}= -0.29 V, V_{CB01}=5.3 V, V_{EB02}= -0.3 V, V_{CB02}=6.6 V. In this case, we get a gain of 6000 or more. In this example, five control functions make up the control vector $U=(u_1,u_2,u_3,u_4,u_5)$. The structural basis of the optimization strategies includes 32 strategies. The mathematical model of circuit (28) includes five equations:

$$\begin{aligned}
g_1(X) &\equiv I_{B1}+(x_6-E_0)x_1^2=0\\
g_2(X) &\equiv I_{E1}+x_7x_2^2=0\\
g_3(X) &\equiv I_{E2}+x_9x_4^2=0\\
g_4(X) &\equiv I_{C2}+(x_{10}-E_1)x_5^2=0\\
g_5(X) &\equiv I_{C1}+I_{B2}+(x_8-E_1)x_3^2=0
\end{aligned} \tag{28}$$

where I_{B1}, I_{B2}, I_{E1}, I_{E2}, I_{C1} and I_{C2} – are the base, emitter, and collector currents, respectively, for both transistors. System (28) is transformed into system (29).

$$(1-u_j)g_j(X)=0,\ j=1,2,3,4,5. \tag{29}$$

The function $F(X,U)$ can be represented by the next equation:

$$F(X,U)=C(X)+\frac{1}{\sigma}\sum_{j=1}^{5}u_jg_j^2(X) \tag{30}$$

To achieve better results, it is possible to use one, two or more switching points between strategies. Table 8 gives the generation number, as well as the processor time, when the function *F* reaches its minimum value with an accuracy of δ for three different strategies: TOS with control vector (00000), MTOS with control vector (11111), as well as a combined strategy with control vector (11111)(00000)(11111) and two switching points Sp1=5 and Sp2=9.

Table 8. Dynamics of changes in the number of generations and processor time (s) of the GA depending on the required precision δ for three strategies: TOS, MTOS and composite strategy (11111)(00000)(11111) with two switch points Sp1 and Sp2.

Precision *δ*	Control vector (00000)	Number of generations (Processor time (s)) Control vector (11111)	Control vector (0000)(1111), Sp1=5, Sp2=9
5×10^{-2}	28563 (931.89)	52 (0.16)	38 (0.235)
10^{-2}	389533 (12708)	56 (0.172)	43 (0.25)
5×10^{-3}	1691364 (55181)	59 (0.182)	47 (0.268)
10^{-3}	-	65 (0.2)	52 (0.278)
10^{-4}	-	80 (0.246)	62 (0.309)
10^{-5}	-	88 (0.271)	66 (0.321)
10^{-6}	-	94 (0.289)	78 (0.358)
1.7×10^{-7}	-	134 (0.412)	87 (0.385)
1.03×10^{-7}	-	-	114 (0.469)

As we can see, the MTOS and the combined strategy require significantly fewer generations than TOS to obtain the same precision. Besides, the TOS does not provide a minimum of the objective function with sufficient accuracy. TOS does not find a solution in an acceptable time if the required error is 5×10^{-3} or less. The MTOS finds a solution with an accuracy of 1.7×10^{-7}, and the composite strategy with an accuracy of 1.03×10^{-7}. Table 8 shows that the composite strategy performs better than MTOS and allows you to find a solution both in fewer generations and in less processor time.

It should be emphasized that the combined strategy is the champion in the case of the requirement to obtain the highest possible accuracy of the optimization process.

The switching point for the combined strategy affects the final result. Table 9 shows the effect of the second switching point Sp2, with the previously defined first point Sp1=5, for a given error $\delta=10^{-6}$.

Table 9. Number of generations as a function of the switching point Sp2 and Sp1=5 for the composite strategy (11111)(00000)(11111).

Switch point Sp2	6	7	8	9	10	11	12	13
Number of generations G	85	82	92	78	81	87	99	118

It is clear that the switching point ultimately determines the number of generations required to achieve a given accuracy. The minimum number of generations is reached if the switching point Sp2=9. The dependencies of the function *F* are shown in Fig. 8 under successive generational change for strategies (00000), (11111) and composite strategy (11111)(00000)(11111) for a given error $\delta = 1.7\times10^{-7}$.

It should be noted that when using TOS, there is no way to solve the optimization problem with an error less than 5×10^{-3}. It can be seen that MTOS and the composite strategy give a large gain in solution accuracy compared to TOS.

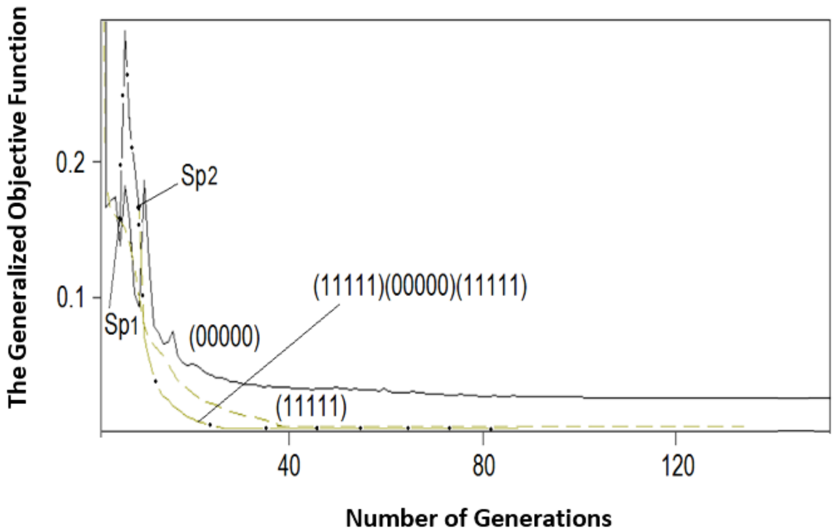


Fig. 8. Dependence of the generalized objective function *F* under successive generational change for strategies (00000), (11111) and composite strategy (11111)(00000)(11111).

For this accuracy, MTOS has a time gain of 303192 times compared to TOS, and the combined strategy has a gain of 205899 times. Thus, for this accuracy, MTOS is a more efficient strategy. However, for accuracy above 10^{-6}, the combined strategy is the most effective.

Using the generalized optimization approach within the genetic algorithm is a mechanism that contributes to changing the internal structure of the vector *X*, and at the same time, changing the structure of the principal function of the GA - the fitness function. This effect manifests itself within the optimization process, since it depends on the structure of the control vector *U*, which can be changed at any step of the optimization process. In this case, the GA has the opportunity to get around local minima and continue the search for a global minimum.

New strategies that appear within the idea of generalized optimization help to increase the accuracy of the solution and reduce the processor time. This can be seen from a comparison of the results obtained using TOS, MTOS and a combined strategy.

When analyzing a composite strategy, the most important element is the optimal switching points between different strategies of structural basis. This problem is solved relatively easily in the case of using the two main strategies TOS and MTOS, and in this case we can talk about a quasi-optimal strategy. In a more general case, it is required to introduce additional conditions that allow, based on certain criteria, to choose the most promising combinations of different strategies.

This approach opens up new possibilities in the problem of finding and constructing global minimization algorithms. However, to construct a quasi-optimal algorithm, additional analysis is required based on the study of the properties of different strategies.

The results obtained in this section show that changing the mechanism for calculating the fitness function during the operation of the GA leads to the exit from local minima and overcoming premature convergence.

In this case, it is possible to improve the accuracy of the solution, which can be transformed into a decrease in the generation number and processor time. It can be noted that the optimization strategies of the generalized approach can be used to improve the algorithms for solving both non-linear programming problems in general and electronic systems optimization problems.

4. CONCLUSIONS

Recently, based on control theory, we developed a generalized approach to the problem of optimizing electronic circuits using deterministic methods such as the gradient method, Newton's method, etc. This made it possible to determine many different optimization strategies by introducing a control vector and to formulate the problem of finding the optimal strategy by optimizing the structure of this vector. It was shown that this approach provides a significant acceleration of the optimization procedure through the use of various strategies and the formation of composite strategies.

The application of a similar approach using a genetic algorithm as the basis of an optimization procedure leads to a change in the structure and main parameters of this algorithm. The results of this investigation demonstrate the possibility of introducing the idea of generalized optimization into the body of the genetic algorithm, which leads to a change in the structure of chromosomes and the fitness function during the operation of the algorithm and the formation of a set of different optimization strategies. In turn, the emergence of a set of strategies inside the GA makes it possible to use various strategies of this set, as well as to form their combinations, which can significantly improve the characteristics of the optimization process. The results obtained showed that changing the main parameters of the GA makes it possible to bypass local minima and overcome premature convergence. An analysis of the optimization procedure for some electronic circuits showed the effectiveness of this approach. In this case, it becomes possible to increase the optimization accuracy by 3–4 orders of magnitude and reduce processor time by 3-5 orders of magnitude compared to traditional GA. Thus, it can be emphasized that new optimization strategies that appear within the framework of the presented methodology have good prospects both for improving the process of solving a nonlinear programming problem in general, and especially for optimizing electronic systems. It can be assumed that such a methodology for solving the optimization problem, based on a generalized approach, can be extended to other stochastic optimization methods, which may be the subject of future research. In this case, an improvement in the performance of the optimization process is also expected.

REFERENCES

[1] G. Stehr, M. Pronath, F. Schenkel, H. Graeb, K. Antreich, "Initial sizing of analog integrated circuits by centering within topology-given implicit specifications," *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, 2003, doi: 10.1109/ICCAD.2003.159696.

[2] A. Girardi, T. De-Oliveira, S. Ghissoni, P. Aguirre, L. Severo, "A comprehensive review on automation-based sizing techniques for analog IC," *Journal of Integrated Circuits and Systems*, vol. 17, no. 3, pp. 1-14, 2022, doi: 10.29292/jics.v17i3.642..

[3] M. Hershenson, S. Boyd, T. Lee, "Optimal design of a CMOS op-amp via geometric programming," *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, vol. 20, no. 1, pp. 1–21, 2001, doi: 10.1109/43.905671.

[4] M. Ogura, V. Preciado, "Optimal design of switched networks of positive linear systems via geometric programming," *IEEE Transactions on Control of Network Systems*, vol. 4, no. 2, pp. 213-222, 2017, doi: 10.1109/TCNS.2015.2489339.

[5] S. Koziel, J. Bandler, K. Madsen, "Space-mapping-based interpolation for engineering optimization," *IEEE Transactions on Microwave Theory and Techniques*, vol. 54, no. 6, pp. 2410–2421, 2006, doi: 10.1109/TMTT.2006.875298.

[6] J. Gong, F. Gillon, N. Bracikowski, "Comparison of three space mapping techniques on electromagnetic design optimization," *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, vol. 37, no. 2, pp. 565-580, 2018, doi: 10.1108/COMPEL-11-2016-0512.

[7] A. Zemliak, "Analog circuit optimization on basis of control theory approach," *COMPEL: The International Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 33, no. 6, pp. 2180–2204, 2014, doi: 10.1108/COMPEL-10-2013-0324.

[8] S. Kirkpatrick, C. Gelatt, M. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983, doi: 10.1126/science.220.4598.671.

[9] M. Moutchou, H. Mahmoudi, A. Abbou, "Induction machine identification based on a new technique of simulated annealing optimization," *Proceedings of International Conference on Renewable and Sustainable Energy*, 2014, doi: I: 10.1109/IRSEC.2014.7059914.

[10] P. Grabusts, J. Musatovs, V. Golenkov, "The application of simulated annealing method for optimal route detection between objects," *Procedia Computer Science*, vol. 149, pp. 95–101, 2019, doi: 10.1016/j.procs.2019.01.112.

[11] B. Liu, Y. Wang, Z. Yu, L. Liu, M. Li, Z. Wang, J. Lu, F. Fernandez, "Analog circuit optimization system based on hybrid evolutionary algorithms," *Integration*, vol. 42, no. 2, pp. 137–148, 2009, doi: 10.1016/j.vlsi.2008.04.003.

[12] S. Ganesh, S. Gurunathan, "Evolutionary algorithms for programming pneumatic sequential circuit controllers," *Procedia Manufacturing*, vol. 11, pp. 1726-1734, 2017, doi: 10.1016/j.promfg.2017.07.299.

[13] H. Nenavath, R. Jatoth, "Hybridizing sine cosine algorithm with differential evolution for global optimization and object tracking," *Applied Soft Computing*, vol. 62, pp. 1019–1043, 2018, doi: 10.1016/j.asoc.2017.09.039.

[14] A. Slowik, H. Kwasnicka, "Evolutionary algorithms and their applications to engineering problems," *Neural Computing and Applications*, vol. 32, pp. 12363-12379, 2020, doi: 10.1016/j.tcs.2018.05.015.

[15] R. Poli, "Analysis of the publications on the applications of particle swarm optimisation", *Journal of Artificial Evolution and Applications*, vol. 2008, p. 685175, pp. 1-10, 2008, doi: 10.1155/2008/685175.

[16] J. Robinson, Y. Samii, "Particle swarm optimization in electromagnetic," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 2, pp. 397–407, 2004, doi: : 10.1109/TAP.2004.823969.

[17] Y. Zhou, S. Pei, "A hybrid co-evolutionary particle swarm optimization algorithm for solving constrained engineering design problems," *Journal of Computers*, vol. 5, no. 6, pp. 965-972, 2010, doi: 10.4304/jcp.5.6.965-972.

[18] V. Kureichik, A. Kazharov, "Algorithms of evolutionary swarm intelligence for solving graph partition problem," *Proceedings of the V All-Russia Science and Technology Conference: Problems of Advanced Micro- and Nano-electronic Systems Development*, 2012,

[19] M. Zaman, M. Gaffar, M. Alam, S. Mamun, M. Abdul Matin, "Synthesis of antenna arrays using artificial bee colony optimization algorithm," *International Journal of Microwave and Optical Technology*, vol. 6, no. 8, pp. 234–241, 2011.

[20] A. Sallem, B. Benhala, M. Kotti, M. Fakhfakh, A. Ahaitouf, M. Loulou, "Application of swarm intelligence techniques to the design of analog circuits: evaluation and comparison," *Analog Integrated Circuits and Signal Processing*, vol. 75, no. 3, pp. 499-516, 2013, doi: 10.1007/s10470-013-0054-6.

[21] F. Li, X. Cai, L. Gao, "Ensemble of surrogates assisted particle swarm optimization of medium scale expencive problems," *Applied Soft Computing*, vol. 74, pp. 291-305, 2019, doi: 10.1016/j.asoc.2018.10.037.

[22] M. Khan, A. Bajpai, "Genetic algorithm and its application in mechanical engineering," *International Journal of Engineering Research and Technology*, vol. 2, no. 5, pp. 677–683, 2013, doi: 10.17577/IJERTV2IS50329.

[23] M. Castellanos, L. Reyes, J. Jimenez, C. Gomez, H. Huacuja, A. Alvim, "A grouping genetic algorithm with controlled gene transmission for the bin packing problem," *Computer Operation Research*, vol. 55, pp. 52–64, 2015, doi:10.1016/j.cor.2014.10.010.

[24] Y. Chuang, C. Chen, C. Hwang, "A simple and efficient real-coded genetic algorithm for constrained optimization," *Applied Soft Computing*, vol. 38, pp. 87–105, 2016, doi: 10.1016/j.asoc.2015.09.036.

[25] L. Corso, A. Gasparin, H. Gomes, "Reliability based design optimization using a genetic algorithm: application to bonded thin films areas of copper/polypropylene," *Chilean Engineering Journal*, vol. 24, no. 3, pp. 510–519, 2016.

[26] A. Khan, R. Mir, "Optimization of constrained function using genetic algorithm," *Computer Engineering and Intelligent Systems*, vol. 8, no. 2, pp. 11–15, 2017.

[27] V. Kobak, A. Zhukovskiy, A. Kuzin, "Modifications of the tournament selection investigation in the inhomogeneous minimax problem solution by using modified Goldberg model,", *Electronic Scientific Journal "Engineering Journal of Don"*, no. 2, pp. 1-10, 2018.

[28] A. Hosseinabadi, J. Vahidi, B. Saemi, A. Sangaiah, M. Elhoseny, "Extended genetic algorithm for solving open-shop scheduling problem," *Soft Computing*, vol. 23, pp. 5099–5116, 2019, doi: 10.1016/j.compeleceng.2018.01.040.

[29] A. Debbah, H. Kherfane, M. Berkani, "Non-Linear robust control optimization to improving the aerodynamics in gas turbine," *Journal of Science and Technology*, vol. 25, no. 2, pp. 78-95, 2019.

[30] C. Goh, Y. Li, "GA automated design and synthesis of analog circuits with practical constraints," *Proceedings of IEEE Congress on Evolutionary Computations*, 2001, doi: 10.1109/CEC.2001.934386.

[31] M. Carneiro, P. Carvalho, N. Deltimple, L. Brito, L. Menezes, E. Kerherve, S. Araujo, A. Rocira, "Doherty amplifier optimization using robust genetic algorithm and unscented transform," *Proceedings of Annual IEEE Northeast Workshop CAS*, 2011, doi: 10.1109/NEWCAS.2011.5981223.

[32] A. Kalentyev, D. Garays, L. Babak, "Genetic-algorithm-based synthesis of low-noise amplifiers with automatic selection of active elements and dc biases," *44th European Microwave Conference*, 2014, doi: 10.1109/EuMC.2014.6986724.

[33] V. Danilchenko, E. Danilchenko, V. Kureichik, "Automated structural-parametric synthesis of single-stage microwave transistor amplifier based on genetic algorithm," *Proceedings of the IX All-*

Russia Science and Technology Conference: Problems of Advanced Micro- and Nano-electronic Systems Development – MES2020, 2020, doi:10.31114/2078-7707-2020-4-83-89.

[34] A. Zemliak, "A modified genetic algorithm for system optimization," *COMPEL - The international journal for computation and mathematics in electrical and electronic engineering*, vol. 41, no. 1, pp. 499-516, 2022, doi: 10.1108/COMPEL-08-2021-0296.

[35] I. Kashirskiy, Y. Trokhimenko, *General Optimization for Electronic Circuits*, Kiev: Tekhnika, 1979,

[36] E. Ochotta, R. Rutenbar, L. Carley, "Synthesis of high-performance analog circuits in ASTRX/OBLX," *IEEE Transactions on Computer-Aided Design of Integrated Circuits*, vol. 15, no. 3, pp. 273-294, 1996, doi: 10.1109/43.489099.

[37] G. Massobrio, P. Antognetti, *Semiconductor Device Modeling with SPICE*, New York: McGraw-Hill, 1993.