БЪЛГАРСКА АКАДЕМИЯ НА НАУКИТЕ • BULGARIAN ACADEMY OF SCIENCES

ПРОБЛЕМИ НА ТЕХНИЧЕСКАТА КИБЕРНЕТИКА И РОБОТИКАТА, **57**
PROBLEMS OF ENGINEERING CYBERNETICS AND ROBOTICS, **57**

София • 2006 • Sofia

New Methods for Circuit Optimization in the Software System Stopcomcir

Galia Marinova[1], *Vassil Guliashki*[2]

[1] *Technical University of Sofia, Faculty of Telecommunications, 1000 Sofia*
E-mail: gim@tu-sofia.bg
[2] *Institute of Information Technologies, 1113 Sofia*
E-mail: vgoul@iinf.bas.bg

I. Introduction

The software system STOPCOMCIR (see [13]) is created for statistical optimization of integrated circuits. In conformity with the statistical design methodology, presented detailed in [15], several optimization strategies could be performed by this system. The statistical design methodology is based on nominal design and includes design centering and optimal tolerancing. The optimal tolerancing goal is to obtain the optimal values for tolerances for the parameters of the circuits elements in order to avoid fail by reducing tolerances where necessary and in order to reduce cost by increasing tolerances where it is possible. The design centering goal is to find the optimal values for the parameters of circuits elements, ensuring maximal intersection between the circuit performance area and the area defined by the constraints and to obtain maximal yield (minimal fail) as a result.

To formulate the arising optimization problems the following notation is used throughout this paper:

$x = (x_1, \ldots, x_n)^T$ is the vector of circuit designable parameters, which values have to be optimized, and which accept discrete values only,

Z_+^n is the set of nonnegative integral n-dimensional vectors,

$C = \{ x \in Z_+^n \mid l_j \le x_j \le u_j; \quad j = 1,\ldots,n\}$ is the Circuit constraints area or the region constraining the vector of circuit designable parameter values,

l_j and u_j are bounds of x_j, such that $((u_j - l_j)/(2u_j))100 = \mathrm{tol}_j$ determines the tolerance of x_j,

$J(x)$ is the output signal or the performance function of the circuit under consideration, Lb and Ub are the bounds (lower and upper) of $J(x)$, Lb and Ub are constant integers.

$Jd(x)$ is the desired output signal function (the desired performance), very often $Jd(x) = (Lb + Ub)/2$.

$P = \{J(x) |, Lb \le J(x) \le Ub\}$ is the performance area of the studied circuit or the acceptability region in the J-space,

$\theta = (\theta_1, \theta_2, \ldots, _k)^T \in R^k$ is the vector of random variables (called "noise" factors), characterizing statistical variations. They represent statistical variations of RLC elements, variations of device model parameters, and environmental effects, such as temperature, supply voltages, etc. Usually θ_i are statistically independent random variables and the other random variables can be related to them through some statistical models,

$f(\theta)$ is the joint probability density function (p.d.f.) of θ.

$e(x,\theta) = (e_1, e_2, \ldots, e_M)^T \in R^M$ is the vector of circuit or simulator variables and parameters used as input parameters to a circuit, process, or system simulator.

$\delta(e(x, \theta))$ is a set indicator function, defined as

$$\delta(e(x, \theta)) = \begin{cases} 1 \text{ if } e(x, \theta)) \in A, \text{ where } A \text{ denotes acceptability region,} \\ 0 \text{ otherwise.} \end{cases}$$

It is assumed, that there is only one circuit output J of interest.

The parametric yield $Y(x)$ is the ratio of the fabricated circuits with an acceptable performance to the total number of circuits manufactured. It is defined (see[15]) as the probability of circuit meeting the design specifications:

$$Y(x) = \int_{R^k} \delta(x, \theta) f(\theta) d\theta. \quad (1)$$

Usually the Monte Carlo method is used to evaluate the integral (1). An unbiased estimator of $Y(x)$ is:

$$Y_{MC}(x) = \frac{1}{m} \sum_{i=1}^{m} \delta(x, \theta^i), \quad (2)$$

where $\{\theta_i, i=1,2,\ldots,m\}$ is a set of m samples generated randomly according to the p.d.f. $f(\theta)$.

As shown in [3] for the class of resistive networks, where the performance function is multilinear, in case the resistors R_i satisfy an essentiality requirement, either the uniform or Dirac impulse distribution is used in the associated Monte Carlo simulation.

Below are listed the strategies for statistical optimization implemented in the system STOPCOMCIR:

- Design centering – optimization of nominal values of the design elements (Fig.1).

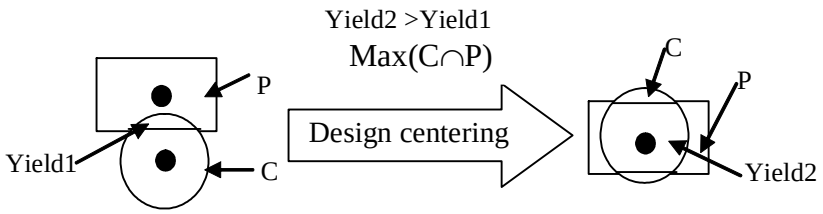

Fig. 1. Design Centering of integrated circuits

The corresponding design centering optimization problem is formulated as follows:

$$\text{Min } F = \sum_{i=1}^{m} \delta_i \tag{3}$$

subject to:

$$\delta_i = \begin{cases} 1 \text{ if } J(x) \notin [Lb, Ub]; \\ 0 \text{ if } J(x) \in [Lb, Ub]; \end{cases} \tag{4}$$

$$l_j \le x_j \le u_j; \quad j = 1,\ldots, n; \tag{5}$$

$$x \in Z_+^n. \tag{6}$$

The Monte Carlo method is used to evaluate F here and m is a positive integer number equal to the number Monte Carlo simulations performed by means of statistical simulator IESD (see [5, 6]) on the circuit under consideration. The optimal solution of (3)-(6) is $F = 0$. Some implementations of design centering as a part of statistical design methodology in electronics are presented in [6, 7, 8].

- Optimal tolerancing - optimization of tolerance values of the circuit elements (see Fig. 2).

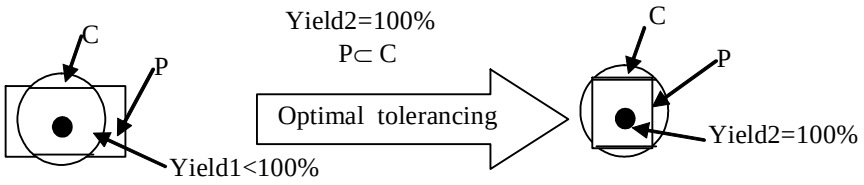

Fig. 2. Optimal Tolerancing

The optimal tolerancing problem can be defined as

$$\text{Max } F_T = (\text{to } l_j) \tag{7}$$

subject to:

$$\sum_{i=1}^{m} \delta_i = 0; \tag{8}$$

$$\delta_i = \begin{cases} 1 \text{ if } J(x) \notin [Lb, Ub]; \\ 0 \text{ if } J(x) \in [Lb, Ub]; \end{cases} \tag{9}$$

$$\text{tol}_j \in T; \quad T = \{1, 2, 5, 10, 15\}; \tag{10}$$

$$u_j = x_j + x_j . \text{tol}_j / 100; \tag{11}$$

$$l_j = x_j - x_j . \text{tol}_j / 100; \tag{12}$$

(13) $$l_j \le x_j \le u_j; \quad j = 1,\ldots,n;$$

(14) $$x \in Z_+^n .$$

➢ Optimal synthesis – selection of an optimal realization from a set of design solutions covering the nominal specification.

The basic circuit synthesis problem can be defined as follows:

(15) $$\text{Min } F_J = (J(x) - Jd(x))^2$$

subject to

(16) $$l_j \le x_j \le u_j; \quad j = 1,\ldots, n;$$

(17) $$x \in Z_+^n .$$

➢ Combined strategy, which combines some of the three predefined strategies.

The problems (3)-(6), (7)-(14) and (15)-(17) are combinatorial optimization problems because discrete optimal values of circuit designable parameters and tolerances have to be found among limited number of permitted range options. As proven in the theory of computational complexity (see [2]), this kind of problems belongs to the class of NP-hard optimization problems. It means, that the computational efforts to obtain exactly the optimal solution of such problems grow exponentially with the increase of problem size (i. e. of the number n of circuit designable parameters). For this reason some heuristic methods, having polynomial complexity, are developed and are implemented or will be implemented as software modules in the system STOPCOMCIR [13]. Another important direction for research is the decrease of the search space dimension, which leads to reduction of the computational efforts and problems with larger size can be solved.

II. Optimization methods in STOPCOMCIR

The optimization methods integrated in STOPCOMCIR are:

- Heuristic method [4, 8] for design centering and/or optimal tolerancing based on the best and worst samples in the set of randomly generated and Monte Carlo simulated circuits.
- Fast design centering method – FDC [9], designed to solve the problem (3)-(6). It is based on Nelder and Mead simplex method (see [14]). The percent of failed circuits is evaluated performing Monte Carlo analysis on the circuit by means of statistical simulator IESD [5, 6].
- Design centering method with space dimension reduction – DCSDR [10, 11], designed to solve the problem (3)-(6). It performs design centering in a reduced search space, taking into account that the circuit output very often depends on the variation only of small number $k < n$ "sensitive" elements. Hereby only the parameters corresponding to the "sensitive" elements are optimized and as a result the efficiency of DCSDR is considerably better than that one of the FDC method.
- Method for statistical estimation of multiple design solutions of a given specification. An illustration of this method is presented in [12], where 3 different power supply solutions for a frequency converter circuit are estimated statistically.

III. New optimization methods for the system STOPCOMCIR

To complete the system STOPCOMCIR methods for circuit synthesis, solving the problem (15)-(17) must be included in it. Here it is necessary to be taken into account that usually the circuit output function $J(x)$ is not known explicitly in an analytical form and its values are evaluated by means of a simulator. This means that neither first nor second derivatives of $J(x)$ are available. Therefore optimization methods based only on evaluation of $J(x)$ value should be used for statistical optimization of integrated circuits in this case. Many complex circuits produce multimodal output $J(x)$, for which stochastic gradient optimization strategies fail to reach the global optimum (see [30]). In case $J(x)$ is a complex multimodal function and/or the performance area P represents not a compact set, but is divided up into several sub-areas, it is worthwhile to be used global search technique, which is capable to find the global optimal solution.

There are several different structural stochastic optimization techniques in the literature. The most important among them are the simulated annealing (see [31, 20-23]), the evolutionary algorithms, such like genetic algorithms (GAs) (see [32, 33, 16, 17] and the swarm intelligence algorithms, such as particle swarm optimization (PSO) (see [34-36]). All these algorithms are inspired by the behavior of the living organisms in some organized systems or by natural processes. Their main features are the robustness of the search process and the convergence to the global optimum. For this reason one genetic-type method and one combined simulated annealing – PSO method are proposed here and will be included in the system STOPCOMCIR. These methods may lead to finding a good initial solution x in the circuit design. But it could happen that great part of circuits generated in a Monte Carlo simulation with nominal solution x have $J(x) \notin [Lb, Ub]$. It is reasonable a method for rough design centering in few steps to be used and after that to run a method for precise design centering. The design centering may lead to solution, where function $J(x)$ differs considerably from $Jd(x)$. Therefore a method for fine design centering, such like FDC or better – DCSDR should be applied to find out the optimal nominal parameters. After that a method for tolerances optimization should be used. In conformity with these ideas here are proposed some new methods, which will be implemented in the system STOPCOMCIR.

The new optimization methods are described below:

- Evolutionary programming (EP) method.

The EP methods are based on the simulation of the mechanism of natural evolution. Well known in EP are the genetic algorithms (GAs) (see [17]). They use a population of individuals (solutions) in the search space and perform search for a global optimal solution by means of consecutive crossovers between the individuals and renovation of the population. GAs do not need highly domain-specific knowledge (only the objective (fitness) function needs to be evaluated during the search process). The main characteristic of GAs is the very good balance between efficiency and efficacy for a broad spectrum of problems (including nonsmooth, multimodal and nonconvex problems). For this reason special genetic-type method will be included in STOPCOMCIR. Unfortunately as a heuristic technique GAs cannot guarantee the obtaining of an optimal solution. This disadvantage could be compensated to a great degree by means of some modifications in the trivial scheme of GAs. A good idea is for example the adjunction of some form of local search to the genetic approach as

considered in [18]. For complex problems the genetic-type optimization may be very slow, requiring hundreds of iterations. To overcome this shortcoming a combination of an EP method with a clustering algorithm (see [19]) is proposed and realized in [16], which enables identifying of the subpopulations that are generated during the search process. In this way the repeated search in one and the same subpopulation could be avoided and the optimization could be accelerated. Clustering should be performed periodically, after a certain number *limgen* of iterations (generations) are passed. A new idea is proposed here to make the method robust in search of global optimum: Instead mutation operator, after each *limgen* of iterations a scatter search is performed along the directions determined by the middle point of segment, connecting the initial best and the current best solution, and by the centers of cluster subpopulations. Below is presented the "pseudo-code" form of the proposed hybrid genetic method:

Hybrid Genetic Method

Generate an initial population POP_0 of *popsize* individuals;

Set $i := 0$; (iteration counter);

Set clustering period *limgen*; Set iteration limit i_{count}; Stopping rule: if $i > i_{count}$ Stop;

While no stopping rule is met **do**

Evaluate the individuals in POP_i; Set $i := i+1$;

For j=1, *popsize*/2; **do**

By means of "Roulette wheel" selection *select* two individuals I_1 and I_2 in POP_{i-1};

Apply the *crossover* operator to I_1 and I_2 for creating offspring O_1 and O_2;

Decide whether or not O_1 and O_2 should enter POP_i for replacing older solutions;

EndFor

Create the population POP_i from POP_{i-1}, replacing the worst individuals in POP_{i-1} with the best generated children individuals**;**

If i is multiple of a given integer *limgen*, perform clustering;

Perform a local search in the neighborhood $N(x_l^*)$ of the best individual x_l^* for each cluster l;

Add the explored clusters to the list of forbidden zones;

Perform a scatter search along the scatter directions and beyond the centers of scatter subpopulations. Replace the worst individuals in the population by the new found better individuals during the scatter search.

EndWhile

Chose the solution (individual) in POP_{icount}, corresponding to the best objective function value as a final solution.

• Simulated annealing method. In contrast to conventional gradient descent optimization methods, simulated annealing search technique avoids the trap of local minima in the design space by means of a heuristic that is analogous to the physical process of annealing a metal, so that the atoms are organized in a metal lattice without flaws and have a minimum energy state. The simulated annealing circuit synthesis begins at a high temperature and cools at an optimum rate to find the global minimal solution in the design space. Several synthesis approaches that optimize the

performance of CMOS RF and analog integrated circuits by means of simulated annealing algorithm have been developed in [20-23]. A kind complex filters are synthesized by means of this technique in [31].

The particle swarm optimization (PSO) was developed in 1995 (see [34]), based on the observed swarm intelligence of insects, birds, etc. The algorithms of this type try to mimic a process of group communication of individual knowledge in order to achieve the optimum (for example food location or flight goal during birds migration). The basic idea of PSO algorithms is to search efficiently the solution space by swarming the particles toward the best fit solution, found at the previous stages (epochs) of the search process. At each epoch the fitness of each particle is evaluated according to the selected fitness function. The algorithm stores the most fit parameters of each particle and replaces them when corresponding better parameters are found. The coding and the implementation of PSO algorithms is relative simple. Another important advantage of these algorithms is their robustness with smaller population size than that one used in the genetic algorithms, which conditions their better computational complexity.

In [20] is proposed a combination of an adaptive simulated annealing method for circuit synthesis with a particle swarm optimization procedure. The adaptive simulated annealing includes a tunneling technique for avoiding the trap of local optima and an adaptive temperature coefficient algorithm. Here is proposed a simple method, combining simulated annealing with a PSO technique and with a local search procedure. In this way may be obtained good computational efficiency and robustness of the search process.

The proposed method is based on the following simulated annealing rule for computation the next design solution:

$$x_{\text{next}} = x_{\text{curr}} + (1\text{–temp})p, \tag{18}$$

where p is the search step and "temp" is an adaptive temperature coefficient.

Here is proposed a new type of tunneling search to improve the efficiency of the method: If there are several known local optima xloc_i, then tunneling search directions z_k are computed as follows:

$$z_k = M_k - x_{\text{next}}, \tag{19}$$

where M_k are the middle points of the segments xloc_i xloc_j, for $i \neq j$.

Below is presented the "pseudo-code" form of the proposed simulated annealing method.

Simulated annealing method

perform initial simulation of popsize random chosen solutions, scattered in the acceptability region;

perform local search around each of them;

Let *xglob* be the best solution (with the best objective function value) found and let x_{prev} be the worst solution found.

Set x_{curr} := xglob. Denote the other *psize*=*popsize*-2 local optimal solutions by xloc_1, xloc_2, …, $\text{xloc}_{\text{psize}}$.

Set i := 0; (iteration counter); Set iterations limit *itlim*;

ITERATION

Compute the C^2_{psize} middle points M_k, k=1,…, C^2_{psize}; of the segments xloc_l xloc_j,

where $l, j = 1,\ldots, C^2_{psize}$; $l \neq j$.
Set temp := 1;
$i := i+1$;
Compute $p := x_{curr} - x_{prev}$.
For lk=1,10; **do**
temp = temp – 0.1;
Perform simulation at: $x_{next} = x_{curr} + (1–temp)p$.
If $F_J(x_{next}) < F_J(x_{curr})$ then
Perform local search around x_{next} and if in the obtained local optimum x_{lk}
$F_J(x_{lk}) < F_J$(xglob) set xglob = x_{lk}.
For k=1, C^2_{psize}; **do**
Perform simulation at: $x_k = x_{next} + (1–temp)(M_k – x_{next})$.
If $F_J(x_k) < F_J(x_{curr})$ then
Perform local search around x_k and if in the obtained local optimum x_{jk}
$F_J(x_{jk}) < F_J$(xglob) set xglob = x_{jk}.
EndFor
EndFor
ENDofITERATION
Set $x_{prev} := x_{curr}$. Set x_{curr} := xglob.
Set $xloc_j := xloc_j + 0.3(xglob – xloc_j)$, j=1,…, C^2_{psize};
Compute the $1C^2_{psize}$ middle points M_k, k=1,…, C^2_{psize}; of the segments $xloc_l$ $xloc_j$.
If i=itlim then STOP, else
Repeat ***ITERATION*** until no improvement of xglob is obtained during the current iteration.
Then STOP.

•Rough design centering method.

Wide known is the Center-of-gravity method (see [15]), which has been investigated by several authors [24] and [25, 26]. The version of the method, proposed in [24] is presented in its essence below and will be included in the system STOPCOMCIR:

Center-of-Gravity Method

Perform random sampling in x_0 by means of Monte Carlo method.

Compute the gravity center G_g of "good" solutions x_i, for which $J(x) \in [Lb, Ub]$ or $\delta_i = 0$ (see [4]).

Compute the center G_f of bad or "fail" solutions x_j, for which $J(x) \notin [Lb, Ub]$ or $\delta_j = 1$ (see [4]).

The new nominal point is searched along the direction from G_f to G_g, or

$$x_{new} = x_0 + (G_f - G_g) \tag{20}$$

Set: $x_0 := x_{new}$.

Repeat the above procedure until no further yield improvement (or fail reducing) is achieved.

The new idea for this method is that it should be applied in a search space with reduced dimension like the DCSDR method (see [10, 11]), especially in case the number of circuit parameters n is great. In this way the efficiency of the method could be essentially improved.

It is shown in [27] that this method may lead to yield improvement rather than to yield maximization. The procedure may be repeated only two or three times in case after that a method for precise or fine design centering will be applied.

•Precise design centering method

The basic idea in this case is to move the nominal solution so deeply as possible in the performance area P (see Fig. 1). As fine design centering may serve a method, searching the geometric center of the acceptability region and putting the nominal solution to coincide with this center. Such method for example is the radial exploration method (see [15]), which has been proposed in [28-29]. The method is presented below and will be included in the system STOPCOMCIR.

Radial Exploration Method

Let the starting solution $x_0 \in P$.

Perform search along n radial lines through x_0 in parameters space – increasing and decreasing each parameter according to suitable discrete scale for it – in order to locate the intersection points x^-_i and x^+_i, i=1,…, n; of these lines with the boundary of the acceptability region P (where $J(x) = Lb$ or $J(x) = Ub$). Compute the middle points M_i of the segments $x^-_i\, x^+_i$, i=1,…, n; Compute the gravity center G_0 of these points.

Generate randomly n new vectors v_i, i=1,…, n; and repeat the above procedure along the n new lines through x_0 determined by the vectors v_i, i=1,…, n; let the new gravity center is denoted by G_1.

Depending on the dimension n of the problem this procedure may be repeated several times. Let in this way be computed k gravity centers.

Then compute the gravity center G of the obtained gravity centers G_i, i = 0, …, k–1; and put the nominal solution x to coincide with G.

This is a heuristic method, but it could be quite efficient as show the obtained practical results (see [29]), especially for linear circuits, because in this case circuit analysis along the radial directions can be performed more efficiently than the standard circuit analyses for separate points in the parameters space. The efficiency of the method could be even better by means of search in a space with reduced dimension, as it is proposed for the Center-of-Gravity method.

• Tolerances optimization method

The goal here is to construct a method for solving the problem (7)-(14). One simple heuristic algorithm for tolerances optimization consists in the consecutive exploration of each parameter and the enlargement of its tolerance so much as possible. This procedure may be inaccurate in some cases and the best tolerances values may not be found.

Below is proposed a method, where combinations of tolerance values are explored at the same time, so that the probability for obtaining the best tolerances is increased.

Tolerances Optimization Method

Compute the "sensitivity" measures σ_i, i=1,…,n; of the corresponding circuit elements like in Step 3 of DCSDR method (see [11]).

Make a list of parameters, arranged according their "sensitivity" measures – in increasing order.

Set the tolerances for elements j, having $\sigma_j < 3\%$ to be large (for example from 10% or 15%).

If there are no parameters k with $\sigma_k > 3\%$ STOP, else:

Explore the most “sensitive” parameter – its index is first in the “sensitivity” list of indices.

Enlarge its tolerance so much as possible and let the obtained threshold value is denoted by “tolmax”.

Explore consecutive the other parameters with $\sigma_k > 3\%$. Enlarge their tolerance values so much as possible. Make an index list *I* of elements, which have obtained the minimal tolerance value.

Make an attempt to decrease the “tolmax” value by 1% and to increase the values of all parameters, which indices belong to *I* at the same time. If the attempt is successful repeat it, else:

Make an attempt to decrease by 1% simultaneously “tolmax” and the tolerance of the next “sensitive” parameter according the computed σ-values. At the same time increase the values of all other all parameters, which indices belong to *I*. If the attempt is successful repeat it, else STOP.

IV. Conclusions

In conclusion it should be mentioned that the theory of statistical optimization of integrated circuits has been continuously renovated by new search techniques during the last two decades. Some new very promising stochastic structural optimization methods are created, where new ideas are proposed to improve the robustness and the efficiency of the methods. The new proposed methods are: one hybrid genetic method and one combined simulated annealing-PSO method, as well as one rough and one precise design centering method and one tolerances optimization method. It is expected that these methods will be capable to find out the global optimal solution of complex integrated circuits with relative large size. Taking into account the good practical results presented in the literature for similar kind methods it is worthwhile to implement the proposed new methods in the software system for statistical optimization of integrated circuits STOPCOMCIR as new software modules. After that the system could be used to attack new circuit synthesis problems.

References

1. Jurisic, D., G. Moschytz, N. Mijat. Low-Sensitivity, Low Power 4th Order Low-Pass Active RC Active Allpole Filter Using Impedance Tapering. – In: MELECON’2004, Proc. Vol. **I**, 12th IEEE Mediteranian Electrotechnical Conference, May 12-15, 2004, Dubrovnik, Croatia, 107-110.
2. Garey, M. R., D. S. Johnson. Computers and Intractability: A Guide to the Theory of NP-Completeness. San Francisco, W. H. Freeman, 1979.
3. Kettani H., B. R. Barmish. A new Monte Carlo Circuit Simulation Paradigm with Specific Results for Resistive Networks. – IEEE Transactions on Circuits and Systems, I: Regular Papers, Vol. **53**, June 2006, No 6, 1289-1299.
4. Marinova, G. Statistical Optimization of Electronic Circuits in the Simulator IESD. – In: Proc. of “Prilojenie na matematikata v tehnikata”, 1992, Sofia, Bulgaria, 144-148 (in Bulgarian).
5. Marinova, G. Statistical Approach in Electronics Design. Ph. D. Thesis, Technical University of Sofia, Bulgaria, 1994.
6. Marinova, G. I. Statistical Design Simulator for Electronic Design – IESD. – In: The European Design and Test Conference, Users Forum, Paris, March 11-14, 1996, p. 297.
7. Marinova, G. I. Statistical Optimization of TV Modules in the Simulator IESD. – In: Proc. of 6th International Conference on Telecommunications in Modern Satellite, Cable and

Broadcasting Services TELSIKS'2003, Nish, Serbia and Montenegro, October 1-3, 2003, 143-146.
8. M a r i n o v a, G. I., I. D. D i m i t r o v. Statistical Analysis and Optimization of Voltage Regulator Circuit Using IESD and ORCAD Environment. – In: Proc. of XXXVIII International Scientific Conference on Information, Communication and Energy Systems and Technologies ICEST'2003, Sofia, Bulgaria, 2003, 478-482.
9. M a r i n o v a, G. I., V. G. G u l i a s h k i. Fast Design Centering Method for Electronic Circuits. – In: Proc. of 6th International Conference on Telecommunications in Modern Satellite, Cable and Broadcasting Services TELSIKS'2003, Nish, Serbia and Montenegro, 2003, 701-704.
10. M a r i n o v a, G., V. G u l i a s h k i, D. D i m i t r o v. A Design Centering Method with Space Dimension Reduction. – In: Proc. of ELEKTRONIKA'2004 Conference, Sofia, Bulgaria, May 21-22, 2004, 134-139.
11. M a r i n o v a, G., V. G u l i a s h k i. Improved Design Centering in a Reduced Search Space for Electronic Circuits Optimization. – In: Proc. of Papers, "XL International Scientific Conference on Information, Communication and Energy Systems and Technologies ICEST'2005", Nish, Serbia and Monte Negro, Jun 29 - July 01, 2005, 166-169.
13. M a r i n o v a, G., V. G u l i a s h k i. Software System for Statistical Optimization of Communication Circuits. – In: Proc. of "Balcan Conference of Young Scientists", Plovdiv, Bulgaria, Jun 16. – 18, 2005, 98-105.
14. R e k l a i t i s, G. V., A. R a v i n d r a n, K. M. R a g s d e l l. Engineering Optimization. Methods and Applications. John Wiley and Sons, 1983, 91-93.
15. Z h a n g, J. C., M. A. S t y b l i n s k i. Yield and Variability Optimization of Integrated Circuits. Klewer Academic Publishers, 1995.
16. D a m a v a n d i, N., S. S a f a v i-N a e i n i. A Hybrid Evolutionary Programming Method for Circuit Optimization. – In: Proc. IEEE Transactions on Circuits and Systems – I: Regular Papers, Vol. **52**, May 2005, No 5, 902-910.
17. G o l d b e r g, D. E. Genetic Algorithms in Search, Optimization and Machine Learning. Addison Wesley, Reading, Mass, 1989.
18. P i r l o t, M. Heuristic Search Methods. – In: Oper. Res. Designing Practical Solutions; Tutorial and Research Review Papers, Euro XIII/OR 36, The Joint EURO/Oper. Res. Society Conference; University of Strathclyde, Glasgow, 19-22, VII. 1994. 180-201.
19. T o r n, A. Cluster Analysis Using Seed Points and Density-Determined Hyperspheres as an Aid to Global Optimization. – IEEE Tran. Syst. Man., Cybern., Vol. **SMC-7**, August 1977, No 8, 610-616.
20. P a r k, J., K. C h o i, D. J. A l l s t o t. Parasitic-Aware RF Circuit Design and Optimization. – In: Proc. IEEE Transactions on Circuits and Systems – I: Regular Papers, Vol. **51**, October 2004, No 10, 1953-1965.
21. G u p t a, R., D. J. A l l s t o t. Parasitic-Aware Design and Optimization of CMOS RF Integrated circuits. – In: Proc. IEEE Radio Frequency Integrated Circuits Symp., June 1998, 325-328.
22. L e e n a e r t s, D., G. G i e l e n, R. A. R u t e n b a r. CAD Solutions and Outstanding Challenges for Mixed-Signal and RF IC Design. – In: Proc. IEEE/ACM Int. Conf. Computer-Aided Design, November 2001, 270-277.
23. K r a s n i c k i, M. J., R. P h e l p s, J. R. H e l l u m s, M. M c C l u n g, R. A. R u t e n b a r, L. R. C a r l e y. ASF: A Practical Simulation-Based Methodology for the Synthesis of Custom Analog Circuits. – In: Proc. IEEE/ACM Int. Conf. Computer-Aided Design, November 2001, 350-357.
24. S o i n, R. S., R. S p e n s e. Statistical Exploration Approach to Design Centering. – In: IEEE Proc., Part G, Vol. **127**, December 1980, 260-269.
25. K j e l l s t r o m, G., L. T a x e n. Stochastic Optimization in System Design. – IEEE Transaction on Circuits and Systems, Vol. **CAS-28**, July 1981, 702-715.
26. K j e l l s t r o m, G., L. T a x e n, P. O. L i n d b e r g. Discrete Optimization of Digital Filters Using Gaussian Adaptation and Quadratic Function Minimization. – IEEE Transaction on Circuits and Systems, Vol. **CAS-34**, October 1987, No 10, 1238-1242.
27. S t y b l i n s k i, M. A. Problems of Yield Gradient Estimation for Truncated Probability Density Functions. – IEEE Trans. on Computer-Aided Design, Vol. **CAD-5**, January 1986, No 1, 30-38.

28. T a h i m, K. S., R. S p e n c e. A Radial Exploration Approach to Manufacturing Yield Estimation and Design Centering. – IEEE Trans. on CAS, Vol. **CAS-26**, September 1979, No 9, 768-774.
29. T a h i m, K. S., R. S p e n c e. A Radial Exploration Algorithm for the Statistical Analysis of Linear Circuits. – IEEE Trans. on CAS, Vol. **CAS-27**, May 1980, No 5, 421-425.
30. K r u s i e n s k i, D. J., W. K. J e n k i n s. Design and Performance of Adaptive Systems, Based on Structured Stochastic Optimization Strategies. – IEEE Circuits and Systems, Vol. **5**, First Quarter 2005, No 1, 8-20.
31. R a d e c k i, J., J. Konrad, D. D u b o i s. Design of Multidimensional Finite Wordlength FIR and IIR Filters by Simulated Annealing. – IEEE Trans. Circuits and Systems II: Analog and Digital Signal Processing, Vol. **42**, June 1995, No 6.
32. T a n g, K. S., K. F. M a n, Q. H e. Genetic Algorithms and their Applications. –IEEE Signal Processing Magazine, November 1996, 22-37.
33. N e u b a u e r, A. Genetic Algorithms for Adaptive Non-Linear Predictors. – IEEE International Conference on Electronics, Circuits and Systems, Vol. **1**, 7-10 September 1998, 209-212.
34. K e n n e d y, J., R. C. E b e r h a r t. Particle Swarm Optimization. – In: Proc. of IEEE International Conference on Neural Networks, Piscataway, N. J., 1995, 1942-1948.
35. E b e r h a r t, R., Y. S h i. Particle Swarm Optimization: Developments, Applications and Resources. – In: Proc. Congr. Evolutionary Computation (CEC 01), Vol. **1**, 2001, 81-86.
36. K r u s i e n s k i, D. J., W. K. J e n k i n s. The Application of Particle Swarm Optimization to Adaptive IIR Phase Equilization. – In: Proc. of the International Conference on Acoustics, Speech and Signal Processing, Montreal, Canada, 17-21 May, 2004, II-693-II-696.

Новые методы для оптимизации схем в софтуерной системе Stopcomcir

Галя Маринова[1], *Васил Гуляшки*[2]

[1] *Технический университет, София*

[2] *Институт информационных технологий, 1113 София*

(Р е з ю м е)

В работе предлагаются несколько новых оптимизационных методов, которые будут использованы для дальнешнего расширения софтуерной системы Stopcomcir, предназначенной для статистической оптимизации интегральных схем. В системе реализованы разные оптимизационные стратегии, решающие оптимизационные задачи разных типов, возникающие в статистическом проектировании интегральных схем. Система Stopcomcir основана на знаниях и ориентирована объектно к аналоговым или к смешанным аналого-цифровым схемам для радио- и телекоммуникации. Несколько оптимизационных методов уже включены как софтуерные модули в системе Stopcomcir. Ожидаются новые экспериментальные результаты.