A Hybrid Evolutionary Programming Method for Circuit Optimization

Nader Damavandi, *Member, IEEE,* and Safieddin Safavi-Naeini, *Member, IEEE*

***Abstract*—A hybrid *evolutionary programming* (EP) method is presented for global optimization of complex circuits. The conventional EP is integrated with a clustering algorithm to improve the robustness of the algorithm for complex multimodal circuit optimization problems. The EP generates populations around the regions of the search space which can potentially contain a minimum but may be overlooked. The clustering algorithm is used to identify these regions dynamically. In order to improve the speed of optimization, the EP is combined with a gradient-based search method in an efficient fashion. The local search is performed from the center of each identified cluster in order to find the minimum in the region very fast. The hybrid algorithm can also reduce the search space by avoiding the search in the areas that were previously investigated. This feature greatly improves the speed of optimization and prevents the premature convergence as well. The algorithm performed very well in several benchmark problems including a test function minimization and global optimization of a complex RF diplexer circuit.**

***Index Terms*—Computer-aided design (CAD), clustering, evolutionary programming (EP), global optimization.**

I. Introduction

SEVERAL optimization methods have successfully been used for solving different types of electrical engineering problems especially in the field of analog and RF/microwave circuits design [1]–[3]. Computer-aided design (CAD) and diagnosis of complex circuits in modern electronics usually involve optimization of highly multimodal objective functions. In many cases, handling of these problems is only possible by using a global optimization method. However, most available global optimization techniques used by many circuit designers are not efficient enough and usually fail to provide the optimal solution in a timely manner.

Among the global optimization methods *evolutionary algorithms* (EAs) have recently attracted much attention for solving complex optimization problems [4], [5]. The EAs, and stochastic methods in general, are good at exploring the search space and locating the region of global minimum, but they are slow at exploitation of the solution. On the other hand, the deterministic local search methods, such as *steepest descent*, *quasi-Newton* and *conjugate gradient* are fast at finding the local solution when they start within the region of minimum, but they may fail to provide the global solution and can be easily trapped into local minima. The deterministic and stochastic methods have complementary properties and it is natural to anticipate a better performance from an optimization method that combines the best features of both. Some attempts have been made to combine the deterministic and stochastic methods into hybrid approaches [7]–[9].

In this paper, we try to improve the diversity of a certain type of EA, called evolutionary programming (EP), in order to make it more suitable for global optimization of complex circuits. Then, the method is hybridized with a gradient-based search method for improving the speed of optimization. In general, the EAs are very efficient in solving difficult nonlinear multidimensional optimization problems with complex landscape. However, depending on the form of landscape, there are two cases, which cannot be handled by EA efficiently. The first case is when the global minimum lies in a very narrow valley and at the same time there exists a strong local minimum with a wide basin of attraction such that the value of objective function is close to the value of objective function at global minimum. We call a function with this behavior "deceptive" function since any algorithm that is based on EA will have difficulty in locating its global minimum. The second case is when we are looking for not only a single global minimum of the objective function but also the major local minima, which are within a certain accuracy level. These points can also be accepted as the solution and the knowledge of their locations may be required for further analysis. The current formulation of the EP algorithm can not keep a stable subpopulation for all the possible minima in the space and it will converge to a single minimum.

An approach called *niching technique* has been previously used for increasing the diversity of EAs [13]–[15], [18]. In general, different versions of this technique depend on some fixed parameters which should be assigned at the beginning of the algorithm. Determination of these parameters needs knowledge of shape of search domain and number of local minima in advance, which are usually not known before optimization and are difficult to calculate accurately. Some efforts on developing adaptive niching has also been reported in [16] and [17].

As an alternative approach, we will use clustering algorithm in order to increase the diversity of the EP and prevent premature convergence. Furthermore, as we will see later, the idea of clustering can be used for combining the EP with local search methods efficiently. The basic idea here is to let the EP form the subpopulations and determine the regions of local minima by cluster analysis and then apply a hill climbing search from a point within each region and find the local minimum for each region very fast.

A brief review of the EP algorithm will be introduced in Section II, and the details of the proposed approach for combining

Manuscript received March 17, 2004; revised August 9, 2004. This paper was recommended by Associate Editor D. P. Mehta.

The authors are with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, ON N2L 3G1, Canada (e-mail: damavandi@ieee.org).

Digital Object Identifier 10.1109/TCSI.2005.846216

clustering algorithm and the EP are presented in Section III. Hybridization of the modified EP with a local search method will be discussed in Section IV, and the results of applying the hybrid EP to two examples will appear in Section V.

II. Review of Evolutionary Programming

The EP, like *genetic algorithms* (GAs), works based on simulation of the mechanism of natural evolution. The EP uses the real-valued optimization variables, which makes its implementation easier compared to the conventional GA. Moreover, in the EP, mutation is the only variation operator and no cross-over is performed. Also, the adaptive mutation in EP can enhance its performance compared to GA [10], [11]. Nonetheless, since the fundamentals behind the hybrid algorithm which will be developed in this work is based on the general behavior of EAs, other type of EAs such as GA can also be used as the main optimizer instead of EP.

In order to review the formulation of EP algorithm, let us assume that the optimization problem has been reduced to

$$\begin{aligned}&\underset{\boldsymbol{x}}{\text{minimize}}\, f(\boldsymbol{x})\\ &\text{s.t. } \boldsymbol{x} \in \{x_{\min}, x_{\max}\}^n\end{aligned} \tag{1}$$

where f is a scalar objective function, which may be discontinuous, and $\boldsymbol{x} = [x(1), x(2), x(3), \cdots, x(\mathrm{n})]^T$ is the vector of optimization variable. In EP, a pair of n-dimensional vectors, consisting of parameter-vector $\boldsymbol{x}$ and its *variation vector* $\boldsymbol{\sigma} = [\sigma(1), \sigma(2), \cdots, \sigma(n)]^T \in \mathbb{R}^{+^n}$, is called an *individual* $\boldsymbol{a} = (\boldsymbol{x}, \boldsymbol{\sigma})$. At the beginning of algorithm, μ individuals are generated using a uniform random distribution to form the initial population. A fitness value is assigned to each individual using the scaled version of objective function $F = G(f(\boldsymbol{x}))$. In most problems, the fitness function is the same as objective function, i.e., $F = f(\boldsymbol{x})$.

In EP, each parent $(\boldsymbol{x}_i, \boldsymbol{\sigma}_i)$ creates a single offspring $(\boldsymbol{x}'_i, \boldsymbol{\sigma}'_i)$ based on the following mutation mechanism:

$$\begin{aligned}x'_i(j) &= x_i(j) + \sigma_i(j)\, N_j(0,1)\\ \sigma'_i(j) &= \sigma_i(j)\, e^{\left[\tau' N(0,1) + \tau N_j(0,1)\right]}, \qquad j = 1, 2, \cdots, n\end{aligned} \tag{2}$$

where $x(j)$ and $\sigma(j)$ are the jth components of the solution vector and the standard deviation vector, respectively. In (2), $N(0,1)$ is a random variable with Gaussian distribution of mean zero and standard deviation one, and $N_j(0,1)$ means that the random variable is generated anew for each value of j. The values for scalars τ and τ' are defined as [10]

$$\begin{aligned}\tau &= \frac{1}{\sqrt{2\sqrt{n}}}\\ \tau' &= \frac{1}{\sqrt{2n}}.\end{aligned} \tag{3}$$

In the next stage, every individual (parents and children) compete in a tournament with q randomly selected opponents based on the measure of their fitness values and scored accordingly. Then the μ individuals out of the union of the parents and children with highest scores are selected to be the parents of next generation. The algorithm goes to reproduction phase and the above process is repeated. The algorithm is terminated either when a maximum number of generations are achieved or when there has been no improvement in the objective function value after a certain number of generations.

III. Enhancement of Evolutionary Programming Using Cluster Analysis

The behavior of population and the pattern of its formation in the process of EP optimization reveal that the individuals form clusters as the algorithm proceeds [18], [19]. These clusters can easily disappear if there is not enough pressure for creation of a stable subpopulation. This behavior is illustrated in Fig. 1 for a typical two-dimensional (2-D) multimodal objective function. Moreover, the traditional EP will normally arrive at the same local solution several times. This means that much of the effort spent on global optimization is the unnecessary reinvestigation of a local solution that has already been found, whereas this effort could be spent on exploring more space and thus increasing the probability of arriving at a global solution. Cluster analysis can be used to alleviate these two difficulties very efficiently.

We use Törn *density clustering* algorithm [20] for identifying the subpopulations that are generated during the process of EP. One of the advantages of the Törn algorithm is its simplicity compared to other approaches. However, other clustering techniques such as *single linkage* or *density linkage* methods [21] could also be employed as well. The density clustering is very well suited for use with EP since the subpopulations formed during the process of evolution indeed represent regions with high point density. The density clustering algorithm works by growing hyperspheres around best individuals (seed points) until the point density in those hyperspheres starts to become smaller than the average point density of the whole space containing all points. In this method, we use the square of the Euclidean distance as the dissimilarity measure.

In optimizing circuits, all optimization variables are not in the same range and need to be normalized before performing the cluster analysis. This can be easily done by the following normalization before the clustering begins:

$$\tilde{x}_i(j) = \frac{x_{\max}(j) - x_i(j)}{x_{\max}(j) - x_{\min}(j)}, \qquad i = 1, 2, \cdots, p; \quad j = 1, 2, \cdots, n \tag{4}$$

where it is assumed that there are p observations (typically $p = \mu$ in EP) on n variables. The resultant variables will be dimensionless and in the range of 0 and 1.

Let us define the density of reference region E containing all individuals and determine the stopping rule for the growth of each cluster. The point density ρ in E is

$$\rho = \frac{\mu}{V} \tag{5}$$

where V is the volume of E. The point density in a region determined by the points in a cluster is greater than ρ. The cluster is grown by enlarging the region around the seed point $\tilde{\boldsymbol{x}}_0$ as long as the point density remains greater than ρ. For EP, the initial seed point will be the point with the best fitness. The consecutive seed points will be the points with best fitness among all the remaining points. The region will be enlarged successively until the point density in the region remains at least ρ. These natural

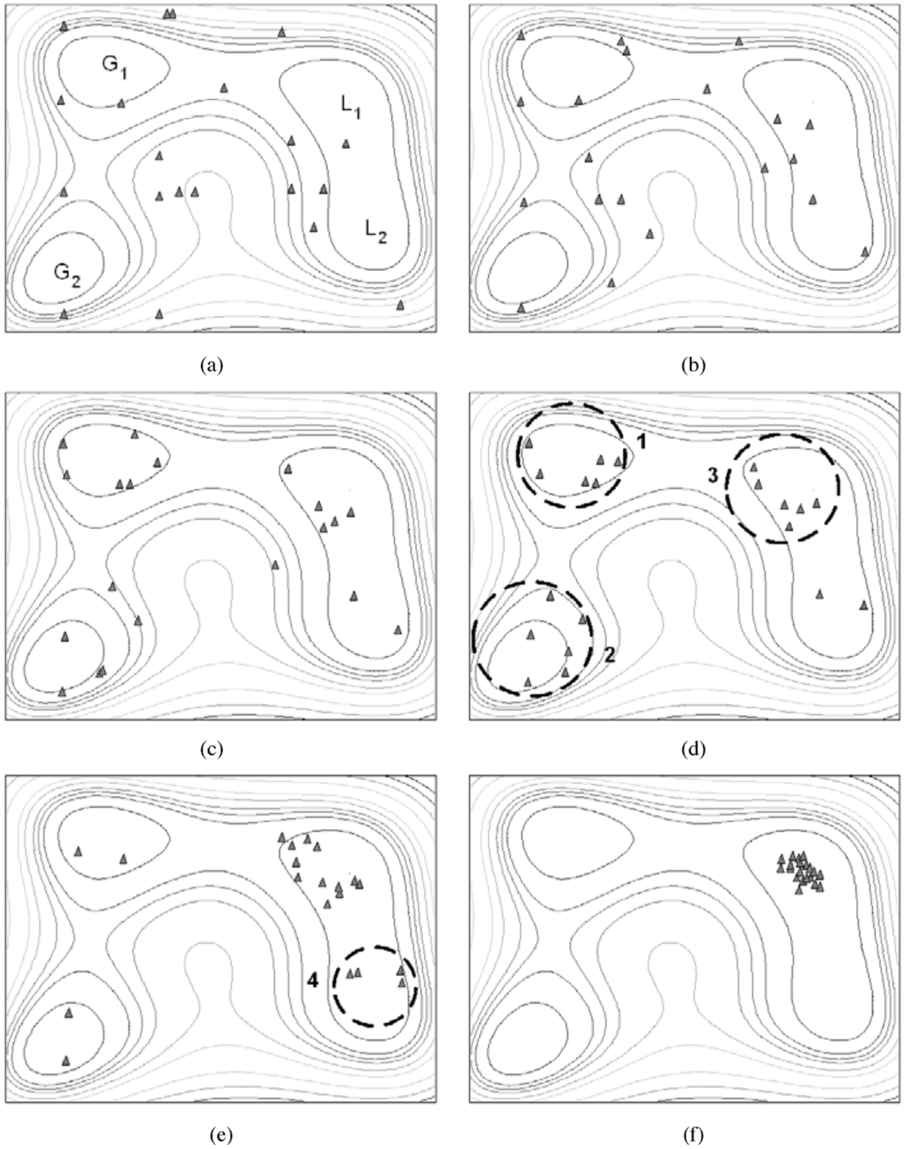


Fig. 1. Illustration of convergence of EP to a single minimum for a sample multimodal 2-D objective function. The individuals are shown at generation. (a) Zero, i.e., initial population. (b) 10. (c) 20. (d) 30. (e) 40. (f) 100. The subpopulations are marked by dashed circles. Two global minima are marked as G_1 and G_2, and two local minima are marked as L_1 and L_2.

regions are concentric hyperspheres. Let $S(n, r_i, \tilde{\boldsymbol{x}})$ be the hypersphere with radius r_i centered at $\tilde{\boldsymbol{x}}$ containing i points. The radius r_i can be calculated as

$$r_i = M(n, \rho) i^{\frac{1}{n}}, \qquad i = 1, 2, \cdots \tag{6}$$

where

$$M(n, \rho') = \begin{cases} \dfrac{\left(\frac{m!}{\rho}\right)^{\frac{1}{n}}}{\sqrt{\pi}}, & n = 2m \\ \left[\dfrac{\Gamma\left(m+\frac{3}{2}\right)}{\rho \pi^{m+\frac{1}{2}}}\right]^{\frac{1}{n}}, & n = 2m+1 \end{cases} \tag{7}$$

and $m = 1, 2, \cdots$.

Using these definitions, different steps of the clustering algorithm can be constructed as follows.

Step 1) Start from $\tilde{\boldsymbol{x}}_0$ and $S(n, r_1, \tilde{\boldsymbol{x}}_0)$.

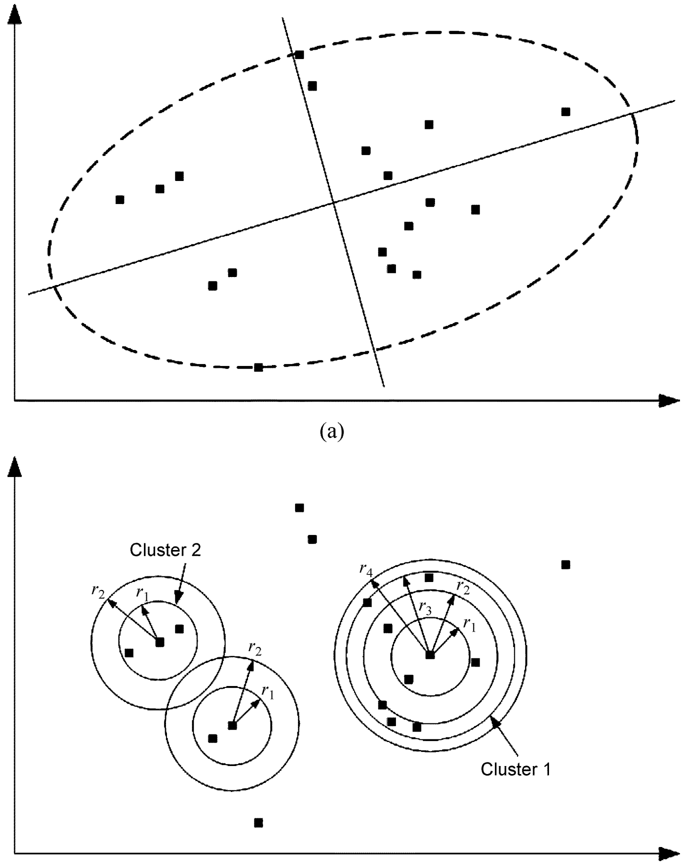
Step 2) Compute the distances between seed point $\tilde{\boldsymbol{x}}_0$ and all the other individuals in the population, i.e., $\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2, \cdots, \tilde{\boldsymbol{x}}_\mu$.

Step 3) Classify points with distances to $\tilde{\boldsymbol{x}}_0$ between successive radii r_i and r_{i+1}, starting with 0 and r_1, until a pair r_k and r_{k+1} is found such that there is no distance falling in the region $[r_k, r_{k+1})$. All points inside the sphere $S(n, r_k, \tilde{\boldsymbol{x}}_0)$ belong to the cluster with $\tilde{\boldsymbol{x}}_0$ as its seed point.

Step 4) Repeat for unclassified points with a new seed point.

Accuracy of the above cluster analysis depends on the choice of reference region E and especially the volume of E. One choice is the smallest coordinate-oriented body containing a given proportion of all the points. In [20], this volume for a hyperellipse body, which its main axes coincide with the principle axes of the data and contains at least 90% of the points, is calculated as the product of the square roots of the largest eigenvalues of the covariance matrix of the data points, i.e.,

$$V = 4^{n'} \prod_{i=1}^{n'} \lambda_i^{\frac{1}{2}} \tag{8}$$



(a) (b)

Fig. 2. Illustration of clustering algorithm for a sample 2-D population. (a) The principle component analysis finds the volume of smallest hyperellipse containing all the points. (b) Hyperspheres are grown from the points with highest fitness until there is no point between two consecutive hyperspheres.

where the eigenvalues λ_i are sorted in a descending order, i.e., $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{n'}$. The constant factor $4^{n'}$ is found empirically to satisfy a wide range of data distributions. We use the standard principle component analysis [22] to find these eigenvalues. If some of the eigenvalues are zero or very small, the points will actually be contained in an n'-space $(n' < n)$ and as a result, we only use the first n' eigenvalues that are nonzero or very small. The criterion for inclusion or exclusion of small eigenvalues in the reference volume calculation is found by progressively checking the size of the threshold distance r_1 as the dimension of space is increased. If the points are approximately contained in a n'-space, we will have

$$r_1(1) \leq r_1(2) \leq \cdots \leq r_1(n') > r_1(n'+1). \tag{9}$$

The distribution of population for a sample 2-D case is illustrated in Fig. 2(a). The smallest ellipse that can contain all the points is shown in this figure with its axes coinciding with the principle axes of the data matrix. The growth of concentric hypersphere (in this case circles) from the points with highest fitness is pictured in Fig. 2(b). In this figure, the radius r_i represent the radius of the circle that should contain at least i points in order to have a point density which is greater or equal to the average point density of the whole ellipse. The circles are successively enlarged until there is no point between two consecutive circles. Only clusters with population of 3 or more are considered for further investigations.

IV. Hybrid EP

Clusters of populations will naturally form around regions of local or global minima. It is therefore reasonable to consider a fast local search in these regions for finding the exact minima. This can be done for example by a gradient-based optimization such as quasi-Newton method or steepest descent. The point with best fitness in each cluster, i.e., the center of cluster, will be used as the starting point of each local search. If the local search does not provide an acceptable solution, the cluster is marked as a *forbidden zone* and the local search for the remaining clusters are performed. After all clusters are checked, a new cycle of EP algorithm will start. The populations that existed inside the previous clusters will be moved outside all the forbidden zones by randomly generating new points. No new points will be allowed inside the forbidden zones during the EP process. This will prevent reinvestigation of regions of space that did not provide any solution, hence greatly improving the robustness of the optimization and adding to its speed as well. Depending on the landscape of optimization, the local search from center of each cluster may converge to either single or multiple solutions. In this paper, we have used quasi-Newton method for the local search, however, when the calculation of gradients is costly or impossible, the local search can be performed using direct search methods such as simplex or pattern search.

The flowchart of hybrid EP is shown in Fig. 3. As can be seen, clustering is performed periodically, after a certain number of generations are passed. This period is called *clustering period*. The EP needs time to explore the space and form clusters, so any attempt to perform clustering before the actual formation of subpopulations will be against the potential benefits of using EP. An attempt to perform the clustering very early, i.e., small clustering period, will lead to a false identification of subpopulations. In some cases, the global solution may be very close to the region of local minimum. If the clustering period is not large enough, the algorithm may identify both regions as one cluster and after a possible unsuccessful local search the cluster will be marked as a forbidden zone, which results in losing the global solution because this region will never be searched again. Consequently, it is important to choose a clustering period that is large enough so the EP has time to completely form stable subpopulations. For example, in Fig. 1, one could have performed the clustering after 100 generations instead of 30 generations, which would have resulted in identification of only one cluster.

The probability of missing a global minimum due to a cluster formation can be reduced by choosing larger clustering period. The price that will be paid for using a larger clustering period is obviously the decrease in the speed of convergence because as clustering period is increased more time is spent on the EP process. Depending on the complexity of the problem, one may need to sacrifice the speed for achieving a higher probability of capturing the global minimum. Our studies show that in general, the clustering period should be increased with the number of optimization variables. For many circuit optimization problems with the number of unknown variables ranging from 10 to 50, the clustering periods of 10 to 100 will be suitable choices. Larger clustering period also will lead to the clusters with smaller sizes. If the clusters become too small, the algorithm has to spend much time on excluding the regions of local minima. One possible remedy

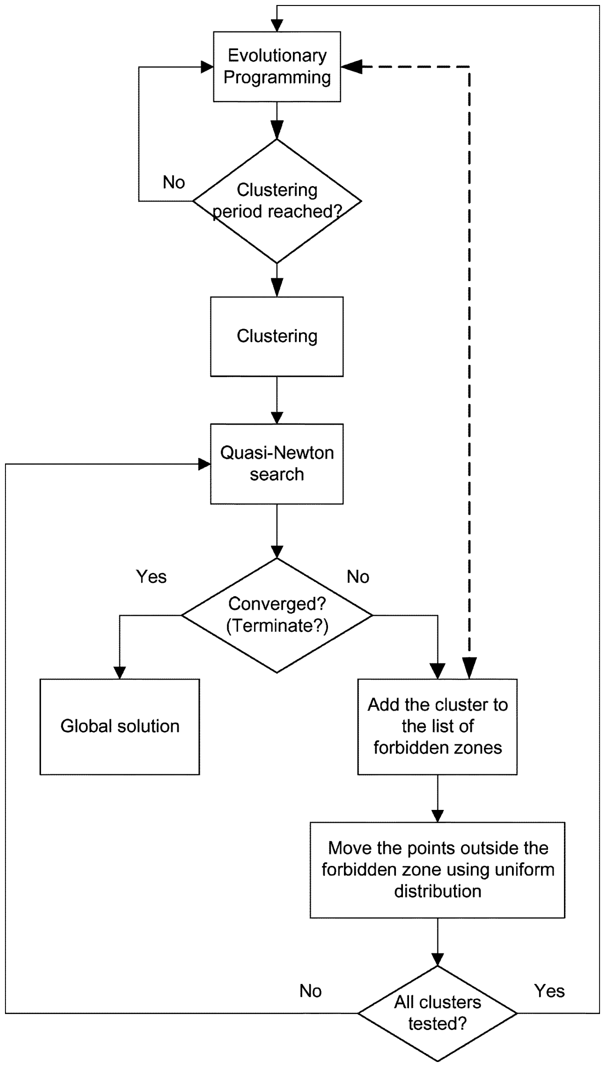


Fig. 3. Flowchart of hybrid EP algorithm.

in these cases is to perform the clustering on combination of parents and children instead. This will cause the data point spread more and potential clusters grow bigger.

Another issue in hybrid EP is the implication of the initial solution on the performance of algorithm. Generally, random search methods do not use the initial estimate of the solution. However, in many practical cases, such as circuit optimization, a rough estimate of the initial solution may be available. In order to take advantage of this knowledge, the EP algorithm is modified to accept the initial solution. At the beginning of the program, this initial solution will be added to the pool of initial parents. If the initial solution is close to a local solution, a subpopulation will soon form around this local minimum and the clustering algorithm will identify it. This cluster will then be marked as forbidden zone and it will be excluded from the search in the next stages.

V. EXAMPLES

The hybrid EP has been successfully applied to many practical problems such as computer aided design and diagnosis of filters and diplexers, parameter extraction of passive RF circuits and modeling of planar antennas [19]. In order to demonstrate the capabilities of our method, the results for two examples are presented.

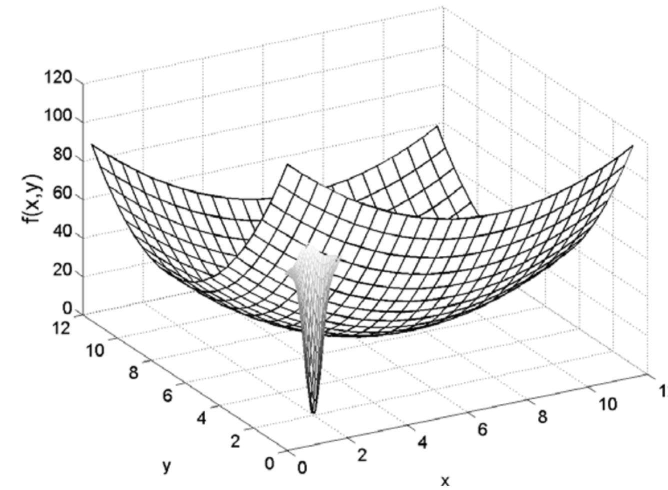


Fig. 4. 3-D plot of function in example A.

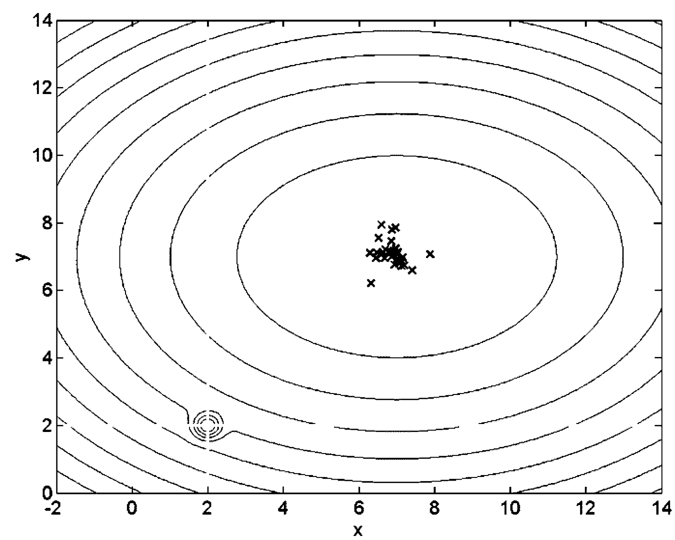


Fig. 5. Contour of function in Example A and the distribution of populations after 20 generations using conventional EP. Individuals are marked by $\mathbf{x}$.

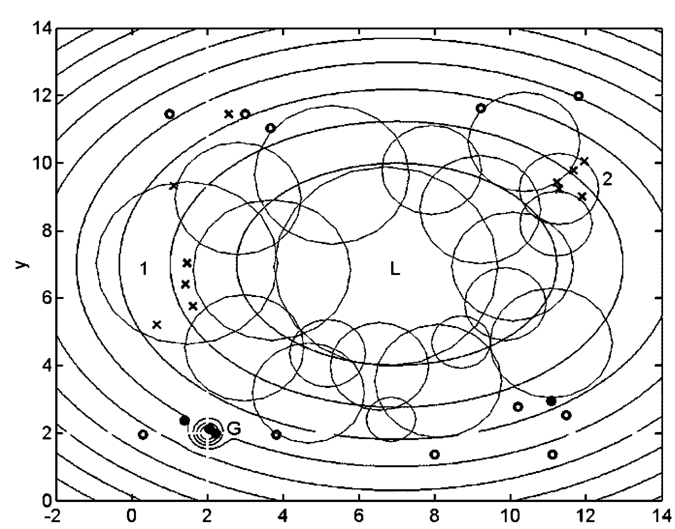


Fig. 6. Contour of function in Example A and the distribution of population after 20 generations using hybrid EP. All identified clusters are shown by large circles. The populations at nineteenth generation are shown by $\mathbf{x}$ and new parents are shown by $\mathbf{o}$.

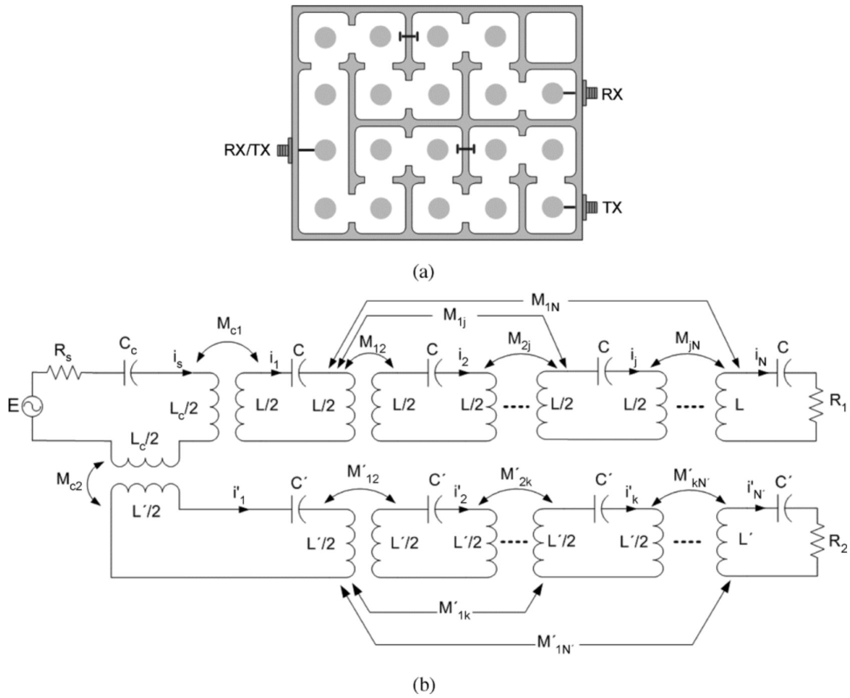


Fig. 7. (a) Top view of the diplexer in example B. (b) General circuit model of diplexer.

A. 2-D Function Minimization

The first example is a 2-D function minimization. This example is chosen to visually show the process of a hybrid EP optimization. The function is defined as

$$f(x,y) = \left[1 - \left|\frac{\sin\left[\pi(x-2)\right]\sin\left[\pi(y-2)\right]}{\pi^2(x-2)(y-2)}\right|^5\right] \times \left[2+(x-7)^2+2(y-7)^2\right]. \quad (10)$$

The three-dimensional (3-D) plot of this function is shown in Fig. 4. This function has a global minimum at (2,2) with a very narrow basin of attraction and a very strong local minimum at (7,7) with wide basin of attraction. The value of function is zero at global minimum and 2 at local minimum.

This function is categorized as a deceptive function as explained in Section I. We first try to optimize this function using the conventional EP. The size of population μ is set to 15 and the size of tournament is 3. The distribution of populations after 20 generations for a sample trial is shown in Fig. 5. As can be seen, the EP has converged to the local minimum and the population was not able to gather around the global minimum in the future generations either. Next, the hybrid EP is used for minimization of the same function. The clustering period is set to 3 and the population size is the same as before. Fig. 6 shows the population distribution after 20 generations for a sample trial. The last two identified clusters are marked by 1 and 2. The global and local minima are marked by $G$and L. As can be seen in Fig. 6, all individuals in the last generation are outside the 19 marked forbidden zones. This has enabled the algorithm to generate points that have higher chances of getting close to the global solution. The algorithm successfully found the global solution at the last generation very fast using a quasi-Newton local search on the identified cluster around the global minimum.

B. Diplexer Synthesis

Next example is the synthesis of an RF diplexer for the DCS1800 mobile base station using an optimization approach. This example is a challenging large complex multimodal optimization problem which can be considered as a good benchmark for testing the performance of the developed global optimizer. The diplexer consists of two ninth degree cross-coupled coaxial resonator filters [23]–[25] and a common loaded resonator for connecting two filters to the antenna port. Physical geometry and the general circuit model of diplexer are shown in Fig. 7.

The receive frequency band is 1710–1785 MHz and the transmit band is 1805–1880 MHz. The diplexer is required to have an equiripple response satisfying the following constraints on the magnitude of its scattering parameters in form of $|S_{ij}| < G_{uk}$

$$\begin{aligned}
|S_{11}| &< -20 \text{ dB}, \ \text{for } 1710 < f < 1785 \text{ MHz}\\
|S_{11}| &< -20 \text{ dB}, \ \text{for } 1805 < f < 1880 \text{ MHz}\\
|S_{21}| &< -72 \text{ dB}, \ \text{for } 1600 < f < 1692 \text{ MHz}\\
|S_{21}| &< -71 \text{ dB}, \ \text{for } 1803 < f < 1990 \text{ MHz}\\
|S_{31}| &< -72 \text{ dB}, \ \text{for } 1600 < f < 1787 \text{ MHz}\\
|S_{31}| &< -71 \text{ dB}, \ \text{for } 1899 < f < 1990 \text{ MHz}.
\end{aligned} \quad (11)$$

The objective function is defined using an l_2 norm

$$U(\boldsymbol{x}) = \sum_{k\in J_u}\sum_{n=1}^{N_{uk}} |e_{uk}(\boldsymbol{x}, f_n)|^2 \quad (12)$$

where

$$\begin{aligned}
e_{uk}(\boldsymbol{x}, f) &= R_k(\boldsymbol{x}, f) - G_{uk}(f), \quad k = 1,2,3\\
R_1(\boldsymbol{x}, f) &= |S_{11}|, \quad R_2(\boldsymbol{x}, f) = |S_{21}|, \quad R_3(\boldsymbol{x}, f) = |S_{31}|\\
J_u(\boldsymbol{x}) &\triangleq \{k | e_{uk}(\boldsymbol{x}, f) \geq 0, k \in I_u\}\\
I_u &= \{1,2,3\}
\end{aligned} \quad (13)$$

and N_{uk} is the number of frequency points used for each response. In this example, we used $N_{uk} = 1000$. Based on the definition of error vectors in (13), the value of objective function at the global solution will be exactly zero.

Due to the complexity of the circuit and the large number of optimization variables with high correlations, the landscape of optimization is very multimodal and most conventional optimization algorithms will trap into a local minimum. However, it is shown that the hybrid EP can successfully optimize the complete circuit and provide the global solution very efficiently. In this diplexer, a cross-coupling between resonator 3 and 6 in the receive-band filter is used to provide the required rejection by generating two transmission zeros on both sides of the passband. Similarly, two transmission zeros for transmit response are generated using a cross-coupling between resonators 4 and 7 in the transmit-band filter. For synthesis of the diplexer, first the values of L, L' and L_c are set to 4 nH and the values of C and C' are calculated for each filter to have the desired center frequencies at 1747.5 and 1842.5 MHz. The value of common resonator capacitance C_c is initially set for a resonance at the upper frequency edge, i.e., 1880 MHz. There are 42 optimization variables including the elements of the two coupling matrices, the input/output impedances R_1, R_2, and R_s, two couplings M_{c1} and M_{c2}, and the common resonator capacitance C_c. The diagonal elements of both coupling matrices are also considered variable to accommodate for the shift in the resonance frequency of each resonator. These shifts are necessary for fine tuning of the diplexer response. During the optimization, the upper and lower limits of all adjacent couplings are set to 0.01 and 0.4 nH, respectively. For the diagonal elements of coupling matrices these limits are -0.4 and 0.4 nH. The limits for cross-coupling elements are set to -0.1 and 0. nH. The box constraints for the rest of variables are

$$\begin{aligned} 0.1 < M_{c1} < 3 \text{ nH} \\ 0.1 < M_{c2} < 3 \text{ nH} \\ 1.0 < C_c < 3 \text{ pF} \\ 50 < R_s < 500\ \Omega \\ 1 < R_1 < 50\ \Omega \\ 1 < R_2 < 50\ \Omega. \end{aligned} \tag{14}$$

The initial estimates of the circuit components for each filter are determined by first synthesizing each filter separately using an approach similar to [24]. The initial response of diplexer is shown in Fig. 8 along with the mask response based on (11). As can be seen, the interaction of filters through the common resonator causes severe deterioration of the responses.

For optimization of this circuit, we used the population size of $\mu = 30$, tournament size of $q = 6$ and the clustering period of 100. The algorithm is not very sensitive to the size of population. However, very large (more than 100) or very small (less than 10) population sizes should be avoided. Very large population size will delay propagation of good individuals and small population will not be able to investigate the domain of search effectively. The best values for the tournament size is in the range of 3–10 for the population size of 20–40 as discussed in [10], [11]. Application of the hybrid EP to this problem resulted in a successful identification of the global solution after 100 generations in average. In this process 15 clusters were identified containing

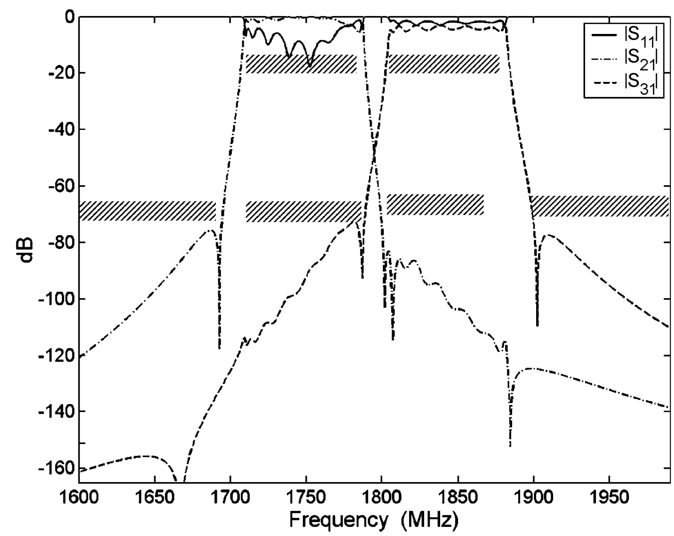


Fig. 8. Initial response of the diplexer before optimization.

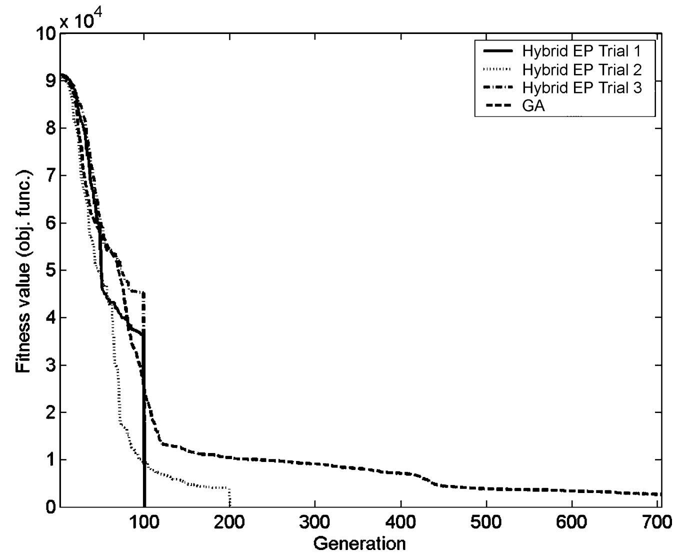


Fig. 9. Fitness versus generation for the best individuals in three trials of diplexer optimization using hybrid EP compared to the conventional GA.

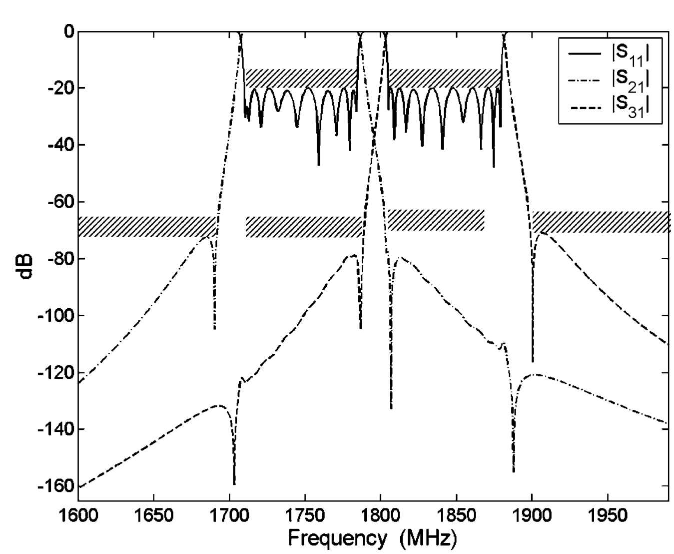


Fig. 10. Response of optimized diplexer at global solution.

TABLE I
FINAL VALUE OF DIPLEXER CIRCUIT COMPONENTS AFTER OPTIMIZATION

Optimization variable	Initial value	Global minimum	Optimization variable	Initial value	Global minimum
M_{c1}	1.397747 nH	2.387857 nH	R_s	16.492807 Ω	368.087934 Ω
M_{c2}	1.325648 nH	2.341246 nH	R_1	1.832534 Ω	1.808298 Ω
C_c	1.965402 pF	1.738125 pF	R_2	1.832534 Ω	1.813527 Ω
M_{11}	0.338306 nH	0.065204 nH	M'_{11}	0.007514 nH	-0.067656 nH
M_{12}	0.139775 nH	0.132521 nH	M'_{12}	0.132565 nH	0.124852 nH
M_{22}	-0.000172 nH	0.012852 nH	M'_{22}	-0.000163 nH	-0.003520 nH
M_{23}	0.100195 nH	0.100631 nH	M'_{23}	0.095125 nH	0.093706 nH
M_{33}	-0.000103 nH	0.008479 nH	M'_{33}	-0.000098 nH	0.003289 nH
M_{34}	0.092193 nH	0.093258 nH	M'_{34}	0.088740 nH	0.088618 nH
M_{36}	-0.015574 nH	-0.014984 nH	M'_{44}	-0.000081 nH	0.004222 nH
M_{44}	-0.000086 nH	0.006125 nH	M'_{45}	0.085548 nH	0.084746 nH
M_{45}	0.106617 nH	0.106462 nH	M'_{47}	-0.014543 nH	-0.014757 nH
M_{55}	-0.000069 nH	0.006695 nH	M'_{55}	-0.000065 nH	0.004350 nH
M_{56}	0.090132 nH	0.090137 nH	M'_{56}	0.100938 nH	0.101208 nH
M_{66}	-0.000086 nH	0.006374 nH	M'_{66}	-0.000081 nH	0.004687 nH
M_{67}	0.093533 nH	0.093546 nH	M'_{67}	0.087535 nH	0.086838 nH
M_{77}	-0.000103 nH	0.006630 nH	M'_{77}	-0.000098 nH	0.004441 nH
M_{78}	0.100246 nH	0.100821 nH	M'_{78}	0.095124 nH	0.094613 nH
M_{88}	-0.000172 nH	0.009700 nH	M'_{88}	-0.000163 nH	0.004811 nH
M_{89}	0.139706 nH	0.140410 nH	M'_{89}	0.132565 nH	0.131728 nH
M_{99}	-0.037039 nH	0.008308 nH	M'_{99}	-0.037001 nH	0.006477 nH

several local minima. The variations of best fitness value in different generations for three trials are shown in Fig. 9. As can be seen, the global solution has been found after performing local search inside the clusters at generations 100 and 200. This figure also shows the fitness versus generation for the conventional GA with the same population size as EP, the crossover rate of 70% and the mutation rate of 1%. The result for GA is an average of three trials and obviously shows the slower rate of convergence. The GA converged to a local minimum with fitness value of approximately 1500 after 700 generations.

The response of optimized diplexer at global solution found by hybrid EP is shown in Fig. 10 and as can be seen it satisfies the required specifications very well. The response also shows an equiripple behavior at this global solution. The final values of optimization variables are shown in Table I.

To compare the performance of hybrid EP with other methods, we also tested the gradient search method, sequential quadratic programming (SQP), and pure random search for solving this problem starting from the same initial estimate that was used for the hybrid EP and GA. Both the gradient method and SQP were trapped into a local minimum with unacceptable response. The diplexer response at one of these local minima found by gradient search method is shown in Fig. 11. The pure random search method could not find the global solution after 50 000 iterations in average either.

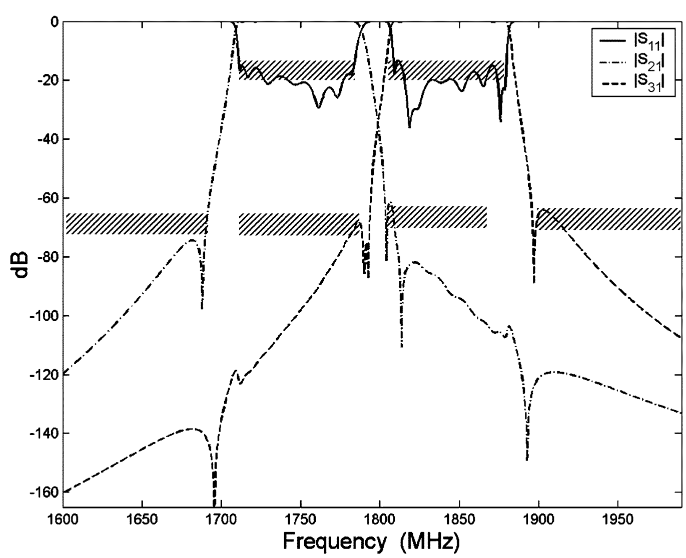


Fig. 11. Response of optimized diplexer at a local minimum.

VI. CONCLUSION

A robust and efficient hybrid EP algorithm was proposed for global optimization of complex circuits. The conventional EP algorithm was modified using a novel integration with clustering algorithm to improve the diversity and reduce the possibility of premature convergence in multimodal problems. It was shown that a local search from the center of each identified cluster can be performed to find the minimum of the objective function at that region very fast. The clustering approach was also used to reduce the size of search domain during the EP process, which helped to improve the speed and robustness of the algorithm. The results for optimization of a mathematical function and a complex RF diplexer circuit were presented. In both cases the hybrid EP was successful in finding the global solution very efficiently. In general, the hybrid EP is suitable for optimization of multimodal objective functions when other conventional approaches suffer from premature convergence.

REFERENCES

[1] S. W. Director, "Survey of circuit-oriented optimization techniques," *IEEE Trans. Circuit Theory*, vol. 18, no. 1, pp. 3–10, Jan. 1971.

[2] R. E. Massara, *Optimization Methods in Electronic Circuit Design*. New York: Wiley, 1991.

[3] K. C. Gupta, R. Garg, and R. Chadha, *Computer-Aided Design of Microwave Circuits*. Dedham, MA: Artech House, 1981.

[4] M. Thompson and J. K. Fidler, "Application of the genetic algorithm and simulated annealing to *LC* filter tuning," *Proc. IEE Circuits, Devices Syst.*, vol. 148, pp. 177–182, Aug. 2001.

[5] J. Liao, M. Li, X. Chen, and J. Hu, "The use of genetic algorithms for statistical circuit design," in *Proc. IEEE Int. Conf. Electronics, Circuits, and Systems*, vol. 2, Dec. 2000, pp. 975–978.

[6] A. Huber, F. Weyhnuller, K. Reiss, and D. A. Miynski, "An evolutionary algorithm for optimization of power supply systems on multichip modules," in *Proc. 6th IEEE Int. Conf. Electronics, Circuits, and Systems, 1999*, vol. 1, Sept. 5–8, 1999, pp. 397–400.

[7] P. Preux and E. G. Talbi, "Toward hybrid evolutionary algorithms," *Int. Trans. Oper. Res.*, vol. 6, pp. 557–570, 1999.

[8] S. Papadopoulos, R. J. Mack, and R. E. Massara, "A hybrid genetic algorithm method for optimizing analog circuits," in *Proc. 43rd IEEE Midwest Symp. Circuits and Systems*, vol. 1, Aug. 2000, pp. 140–143.

[9] M. Okamoto, T. Nonaka, S. Ochiai, and D. Tominaga, "Nonlinear numerical optimization with use of a hybrid genetic algorithm incorporating the modified Powell method," *Appl. Math. Comput.*, vol. 91, pp. 63–72, 1998.

[10] T. Bäck, *Evolutionary Algorithms in Theory and Practice*. Oxford, U.K.: Oxford Univ. Press, 1996.

[11] D. B. Fogel, *Evolutionary Computation, Toward a New Philosophy of Machine Intelligence*. New York: IEEE Press, 1995.

[12] N. Damavandi and S. Safavi-Naeini, "A robust model parameter extraction technique based on meta-evolutionary programming for high speed/high frequency package interconnects," in *Proc. Canadian Conf.Electrical and Computer Engineering IEEE-CCECE*, vol. 2, 2001, pp. 1151–1155.

[13] D. E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Reading, MA: Addison-Wesley, 1989.

[14] D. E. Goldberg and J. Richardson, "Genetic algorithms with sharing for multimodal function optimization," in *Proc. 2nd Int. Conf. Genetic Algorithms*, 1987, pp. 41–49.

[15] K. Deb and D. E. Goldberg, "An investigation of niche and species formation in genetic function optimization," in *Proc. 3rd Int. Conf. Genetic Algorithms*, Dec. 1989, pp. 42–50.

[16] D. E. Goldberg and L. Wang, "Adaptive niching via coevolutionary sharing," in *Genetic Algorithms and Evolutionary Strategies in Engineering. and Computer Science*, 1st ed, D. Quagliarella, J. Periaux, C. Poloni, and G. Winter, Eds. Chichester, U.K.: Wiley, 1997, pp. 21–38.

[17] S. W. Mahfoud, "Niching methods for genetic algorithms," Ph.D. dissertation, Dept. General Eng., Univ. of Illinois at Urbana-Champaign, Urbana, IL, 1995.

[18] N. Damavandi and S. Safavi-Naeini, "Evolutionary programming with niching technique for efficient model parameter extraction," in *Proc. IEEE Antennas Propagation Soc. Int. Symp.*, vol. 4, 2001, pp. 680–683.

[19] N. Damavandi, "Optimization of RF/microwave filters and diplexers using hybrid evolutionary programming and model calibration technique," Ph.D. dissertation, Dept. Elect. Comput. Eng., Univ. of Waterloo, Waterloo, ON, Canada, 2004.

[20] A. Törn, "Cluster analysis using seed points and density-determined hyperspheres as an aid to global optimization," *IEEE Tran. Syst. Man, Cybern.*, vol. SMC-7, no. 8, pp. 610–616, Aug. 1977.

[21] A. Törn and A. Žilinskas, "Global optimization," in *Lecture Notes in Computer Science*, G. Goos and J. Hartmanis, Eds. Berlin, Germany: Springer-Verlag, 1989, p. 350.

[22] C. Chatfield and A. J. Collins, *Introduction to Multivariate Analysis*. New York: Chapman and Hall, 1980.

[23] M. El Sabbagh, K. A. Zaki, H.-W. Yao, and M. Yu, "Full-wave analysis of coupling between combline resonators and its application to combline filters with canonical configurations," *IEEE Trans. Microwave Theory Tech.*, vol. 49, no. 12, pp. 2384–2393, Dec. 2001.

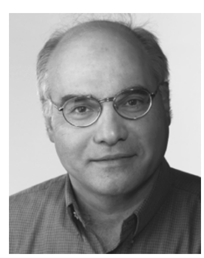
[24] W. A. Atia, K. A. Zaki, and A. E. Atia, "Synthesis of general topology multiple coupled resonator filters by optimization," in *Proc. IEEE Int. Microwave Symp.* , 1998, pp. 821–824.

[25] A. A. Omar, S. Safavi-Naeini, and S. K. Chaudhuri, "Design and synthesis of microwave coupled resonator diplexers," in *Proc. IEEE Antennas Propagation Soc. Int. Symp.*, vol. 1, Jun. 2002, pp. 376–379.

Nader Damavandi (S'89–M'95–S'99–M'04) received the B.Sc. degree in electrical engineering from K. N. Toosi University of Technology, Tehran, Iran, the M.Sc. degree (with distinction) in electrical engineering from Tehran Polytechnic University, Tehran, Iran, and the Ph.D. degree in electrical and computer engineering from University of Waterloo, Waterloo, ON, Canada, in 1991, 1994, and 2004, respectively.

From 1994 to 1999, he was a RF/Microwave and Antenna Researcher in the Iran Telecommunication Research Center, Tehran, Iran. In May1999, he joined the University of Waterloo, Waterloo, ON, Canada, where he carried out research in RF/microwave and Photonics Group. He is currently a Post-Doctoral Fellow in the Department of Electrical and Computer Engineering, University of Waterloo. His research interests include optimization techniques, EM modeling, RF/Microwave filters, RF circuit design and antenna design for wireless applications.

Dr. Damavandi is the recipient of Canadian Wireless Telecommunication Award and has held the Ontario Graduate Scholarship (OGS) for two consecutive years.



Safieddin Safavi-Naeini (S'75–M'78) received the B.Sc. degree in electrical engineering from the University of Tehran, Tehran, Iran, in 1974, and the M.Sc. and Ph.D. degrees in electrical engineering from the University of Illinois at Champaign-Urbana, in 1975 and 1979, respectively.

From 1980 to 1995, he was an Assistant Professor and then an Associate Professor in the Electrical Engineering Department, University of Tehran. In 1996, he joined the University of Waterloo, Waterloo, ON, Canada, where he is currently a Professor in the Department of Electrical and Computer Engineering and also holds the NSERC/RIM Industrial Research Chair in "Intelligent Integrated Radio/Antenna Systems and Novel Electromagnetic Media Technologies." He has been a Scientific and Technical Consultant to numerous national and international telecom industrial and research organizations over the last 20 years. He has authored or coauthored over 130 papers in technical journals and conferences. His research interests and activities include numerical electromagnetics applied to analysis and design optimization of RF/microwave/millimeter-wave systems and circuits, antenna and propagation, wireless communication systems, very high-speed digital circuits, and photonics.