Genetic Algorithm Optimization Applied to Electromagnetics: A Review

Daniel S. Weile and Eric Michielssen, *Member, IEEE*

Invited Review Paper

***Abstract*—Genetic algorithms are on the rise in electromagnetics as design tools and problem solvers because of their versatility and ability to optimize in complex multimodal search spaces. This paper describes the basic genetic algorithm and recounts its history in the electromagnetics literature. Also, the application of advanced genetic operators to the field of electromagnetics is described, and design results are presented for a number of different applications.**

***Index Terms*—Genetic algorithms.**

I. Introduction

During the latter half of the nineteenth century, the biological sciences underwent a revolution when Charles Darwin discovered the processes by which nature selects and optimizes organizms fit for life [1]. About the same time, Gregor Mendel learned the basic laws of genetic inheritance which elucidate by what means evolution takes place [2]. The advent of computers and powerful computational techniques now enables us to apply Nature's optimization processes in the form of genetic algorithms (GA's) to devices built using Maxwell's equations.

The objective functions that arise in electromagnetic optimization problems are often highly nonlinear, stiff, multiextremal, and nondifferentiable. In addition, they are almost always computationally expensive to evaluate. Historically, the vast majority of research efforts related to the design of electromagnetic systems using optimization methods has relied on deterministic optimization methods (DOM's) [3]. DOM's are known to have important drawbacks when applied to multiextremal and stiff optimization problems [4] and often lead to highly interactive and expensive design procedures. GA's (along with Monte Carlo techniques and simulated annealing) belong to a small but growing class of so-called global optimizers which are stochastic in nature and, therefore, less prone to converge to a weak local optimum than DOM's [5]–[10]. In various forms, GA's have been applied to problems in fields ranging from engineering to economics and artificial intelligence and their use is becoming widespread in electromagnetics [11]–[13]. GA's can easily be applied to problems involving nondifferentiable functions and discrete search spaces. Also, they have an inherent parallel nature which renders them attractive for electromagnetic optimization. Motivated by Darwin's theory of descent with modification by natural selection, GA's *evolve* a population of candidate designs toward a goal. The stochastic nature of GA's should not be confused with directionlessness as evolution is emphatically not a random walk; the principle of survival of the fittest leaves little room for interpretation.

This study considers the history and current directions of GA's in electromagnetics. In addition, it explains the GA itself and provides the reader with guidelines for the implementation of a successful GA. Section II introduces GA's and their operators. Section III recounts the application of GA's in the electromagnetics literature. Advanced genetic techniques which demonstrate the wide applicability of GA's to electromagnetic device optimization are discussed in Section IV. Conclusions and avenues for future research on the application of GA's to electromagnetic problems are presented in Section V.

II. Genetic Algorithms: An Introduction

In their most basic form, GA's are function optimizers, i.e., methods for seeking extrema of a given objective function $f(\mathbf{x})$ where $\mathbf{x} = \{x_l | l = 1, 2, \cdots, N_x\}$. In the electromagnetic community, function optimizers of all types are used primarily in design and in solving inverse problems. In design problems, the x_l are parameters describing salient features of a design, and $f(\mathbf{x})$ is a measure of system performance to be either maximized or minimized. The x_l can be continuous or discrete in nature, and complex, real or integer valued. Whatever the nature of the search space, though, GA's differ from most traditional optimization methods in two ways. First, they do not necessarily operate directly on the design parameters $\mathbf{x}$, and second, they simultaneously optimize entire *populations* of designs at once, not a single design at a time. We discuss both of these characteristics in turn.

In general, GA's do not operate directly on the parameter vector $\mathbf{x}$, but on a symbolic representation $\mathbf{p}$ of $\mathbf{x}$, known as a *chromosome*. A chromosome is a concatenation of *genes*

Manuscript received August 7, 1996; revised September 30, 1996. This work was supported in part by the National Science Foundation under Grant ECS-9502138, The AFOSR under Grant F49620–96–1–0025, and an NSF Graduate Fellowship.

The authors are with the Department of Electrical and Computer Engineering, Center for Computational Electromagnetics, University of Illinois at Urbana-Champaign, Urbana, IL 61801 USA.

Publisher Item Identifier S 0018-926X(97)02284-9.

which decode to the x_l, and is symbolically denoted as

$$\mathbf{p} = \{g_j | j = 1, 2, \cdots, N_{gl}\} \tag{1}$$

where N_{gl} is the genetic length and there is a correspondence between the x_l and the g_j given by

$$\begin{aligned}\mathbf{x} &\leftrightarrow \\ \mathbf{p} &\leftrightarrow \overbrace{g_1 g_2 \cdots g_{N_1}}^{x_1} \overbrace{g_{N_1+1} g_{N_1+2} \cdots g_{N_2}}^{x_2} \cdots \\ &\overbrace{g_{N_{N_x-1}+1} g_{N_{(N_x-1)}+1} \cdots g_{N_{N_x}}}^{x_{N_x}}\end{aligned} \tag{2}$$

where $N_{gl} \equiv N_{N_x}$. In allelic GA's, the genes g_j are selected from a finite alphabet (most often the binary alphabet $\{0, 1\}$), whose members are known as *alleles*, which together decode to a discrete or (discretized) continuous parameter. For example, in binary-coded GA's, real-valued parameters x_l can be decoded through a simple linear transformation

$$x_l = x_l^{\min} + 2^{-N_{l-1}} \frac{x_l^{\max} - x_l^{\min}}{2^{N_l - N_{l-1}} - 1} \sum_{j=N_{l-1}+1}^{N_l} g_j 2^{j-1} \tag{3}$$

where x_l takes values ranging from $x_l^{\min}$ to $x_l^{\max}$, and $N_0 = 0$. This type of allelic decoding can be avoided entirely, however, through the use of real-coded GA's which use real genes g_j (instead of selecting them from an alphabet) for representing real-valued parameters through the simple genotype to phenotype mapping $x_l = g_l$ so that $N_{gl} = N_{N_x} = N_x$ [14].

The second difference between GA's and more standard optimizers is that GA's do not work on a single chromosome at a time, but on a whole population of N_{pop} chromosomes

$$\mathbf{P} = \{\mathbf{p}_i | i = 1, \cdots, N_{\text{pop}}\}. \tag{4}$$

The GA itself is composed of operators which produce a succession of populations whose members will have generally improving objective function values. Given a population $\mathbf{P}^k = \{\mathbf{p}_i^k | i = 1, \cdots, N_{\text{pop}}\}$, a single GA iteration starts by evaluating the vector $\mathbf{F}^k = \{f_i^k | i = 1, \cdots, N_{\text{pop}}\}$ of objective function values f_i^k associated with chromosomes $\mathbf{p}_i^k$. The GA then applies the genetic operators of *selection, crossover* and *mutation* to $\mathbf{P}^k$ to produce $\mathbf{P}^{k+1}$.

To commence this iteration, the GA generates a random initial population $\mathbf{P}^0$, although *a priori* knowledge sometimes is invoked to seed $\mathbf{P}^0$ with good chromosomes. With the creation of $\mathbf{P}^0$, the population enters the main GA loop which is iterated on each successive population $\mathbf{P}^k$. Genetic manipulation of $\mathbf{P}^k$ begins with the selection of good chromosomes on the basis of their objective function values $\mathbf{F}^k$. The selection operator produces a new population $\mathbf{P}_{\text{S}}^k = \text{S}(\mathbf{P}^k)$, also of N_{pop} designs which, on average, will be better than those in $\mathbf{P}^k$. This can be achieved by many different schemes, but the most common methods are *roulette-wheel, ranking,* and *stochastic binary tournament selection*. For simplicity, we discuss selection in the context of maximization problems, but all of the selection schemes described can be applied to minimization problems either directly or with a change of objective function through a linear transformation. Roulette wheel selection randomly places each design in the next population with a probability proportional to its fitness value

$$\text{S}(\mathbf{P}^k) = \bigcup_{i=1}^{N_{\text{pop}}} \text{s}(\mathbf{P}^k) \tag{5}$$

where $\text{s}(\mathbf{P}^k) = \mathbf{p}_j^k$ with probability

$$\Pr[\text{s}(\mathbf{P}^k) = \mathbf{p}_j^k] = \frac{f_j^k}{\sum_{n=1}^{N_{\text{pop}}} f_n^k}. \tag{6}$$

Ranking selection comes in many different forms, but a popular version places n copies of the best N_{pop}/n designs into $\mathbf{P}_{\text{S}}^k$. The parameter n controls the greediness of the ranking algorithm; a large n will force a fast convergence at the likely expense of a thorough global search. Faster than both ranking and roulette-wheel selection is stochastic binary tournament selection which iteratively chooses pairs of chromosomes from $\mathbf{P}^k$ and places the better one in $\mathbf{P}_{\text{S}}^k$ until it is satiated. Invoking Darwin's ideas, the selection operator is responsible for the algorithm's convergence, and is the only operator which involves the objective function values $\mathbf{F}^k$.

Selection is followed by crossover, which serves to hybridize design traits by creating a new population $\mathbf{P}_{\text{C}}^k = \text{C}(\mathbf{P}_{\text{S}}^k)$, again of size N_{pop}. Crossover can be denoted as the union over a composition of operators:

$$\text{C}(\mathbf{P}_{\text{S}}^k) = \bigcup_{i=1}^{N_{\text{pop}}/2} \text{c}[\text{ch}(\mathbf{P}_{\text{S}}^k), \text{ch}(\mathbf{P}_{\text{S}}^k)] \tag{7}$$

where the operator "ch"$(\mathbf{P})$ chooses a random chromosome from $\mathbf{P}$ and the operator "c" maps a pair of chromosomes $\mathbf{p}_1 = \{g_{1j} | j = 1, 2, \cdots, N_{gl}\}$ and $\mathbf{p}_2 = \{g_{2j} | j = 1, 2, \cdots, N_{gl}\}$ to another pair according to the rule

$$\text{c}(\mathbf{p}_1, \mathbf{p}_2) = \begin{cases} \hat{\mathbf{p}}_1, \hat{\mathbf{p}}_2 \text{ with probability } p_{\text{cross}} \\ \mathbf{p}_1, \mathbf{p}_2 \text{ with probability } 1 - p_{\text{cross}} \end{cases} \tag{8}$$

where the hybrids $\hat{\mathbf{p}}_1$ and $\hat{\mathbf{p}}_2$ are given by

$$\begin{aligned}\hat{\mathbf{p}}_1 &= \{g_{11}, g_{12}, \cdots, g_{1k}, g_{2(k+1)}, \cdots, g_{2N_{gl}}\} \\ \hat{\mathbf{p}}_2 &= \{g_{21}, g_{22}, \cdots, g_{2k}, g_{1(k+1)}, \cdots, g_{1N_{gl}}\}.\end{aligned} \tag{9}$$

The crossover locus k in (9) is chosen randomly from integers between 1 and $N_{gl} - 1$. This specific single locus implementation is known as one-point crossover. Motivated by the fact that one-point crossover disallows the possibility of alleles residing at opposite ends of the chromosome from ever staying together during a cross, alternative mechanisms have been proposed. For example, in two-point crossover the chromosomes to be mated are split at two loci and exchange the central set of genes, allowing the alleles on the ends to stay together. In uniform crossover, each of the parents' genes is exchanged with a given probability. The crossover operator for real-coded GA's has not been as well codified or studied as those for binary-coded GA's, but one technique for accomplishing it can be found in [15]. Crossover is the main search tool of the GA since it combines chromosomes which contain genetic information which is known to be useful.

Finally, the mutation operator creates a new population $\mathbf{P}_{\text{M}}^k = \text{M}(\mathbf{P}_{\text{C}}^k)$ of size N_{pop} by randomly perturbing genes. If we denote the members of the population after crossover as $\tilde{\mathbf{p}}_i^k$, mutation for allelic algorithms can be defined as

$$\mathrm{M}(\mathbf{P}_{\mathrm{C}}^k) = \bigcup_{i=1}^{N_{\mathrm{pop}}} \mathrm{m}(\tilde{\mathbf{p}}_i^k) \tag{10}$$

where, given a chromosome $\mathbf{p} = \{g_j | i = 1, \cdots, N_{gl}\}$, $\mathrm{m}(\mathbf{p}) = \{\mu(g_j) | j = 1, \cdots, N_{gl}\}$, and

$$\mu(g) = \begin{cases} \hat{g} \text{ with probability } p_{\mathrm{mut}} \\ g \text{ with probability } 1 - p_{\mathrm{mut}} \end{cases} \tag{11}$$

where $\hat{g}$ is any other legal allele. In real-coded algorithms, mutation is very similar to that described for allelic GA's, except that $\hat{g}$ is, in general, a small random perturbation of its initial value g [15]. Mutation is included to prevent premature convergence by ensuring allelic diversity at each gene position. In general, it is not considered a search tool as, unlike crossover, it alters chromosomes blindly.

The population resulting from the application of the mutation operator serves as the starting point for the next generation or $\mathbf{P}^{k+1} = \mathbf{P}_{\mathrm{M}}^k$; hence, a single GA generation can be described as $\mathbf{P}^{k+1} = \mathrm{M}\{\mathrm{C}[\mathrm{S}(\mathbf{P}^k)]\}$. If the best member of $\mathbf{P}^k$ is not in $\mathbf{P}^{k+1}$, an *elitism* operator is very often included to insert this best member into $\mathbf{P}^{k+1}$ [6]. This process is iterated until no improvement is observed, a set goal is met, or a set number of generations has passed. Though the algorithm described here forms the basis of all GA's, a plethora of advanced operators (see Section IV) may be included to ameliorate the performance of the simple GA.

Despite the successes of GA's and their facile implementation, GA theory remains largely enigmatic. Currently, there is no general GA theory to prove population convergence to global or even local optima, nor is there any clear delineation between problems on which GA's work well or poorly. Still, some questions about the implementation of GA's can be answered, either based on the incomplete models developed by GA theorists, or from the experimental results garnered by GA researchers. Therefore, we catalogue these implementation guidelines here.

Most of the results of theoretical GA studies revolve around the concept of *schemata*, or similarity templates, and apply to binary-coded GA's using roulette-wheel selection and single-point crossover. A schema is a set of chromosomes which are similar in some way. For instance, the schema $1{*}{*}0{*}{*}{*}{*}{*}{*}$ is the set of binary 10-b chromosomes with *defined bits* of one in the first position and zero in the fourth position, and *don't cares* "$*$" elsewhere. There are two important theorems dealing with the processing of schemata by the GA [7], [8], [16]–[18].

- The *schema theorem* states that schemata with above average objective function values will receive exponentially increasing representation in successive populations, especially those schemata which have very few defined bits located close to one another.
- The *implicit parallelism theorem* states that the GA processes $O(N_{\mathrm{pop}}^3)$ schemata in each generation. It also shows that for allelic GA's, more schemata are processed if the alphabet is small. Historically, this was the motivation for using binary codings in the first GA's.

These theorems imply (indirectly) the following implementation guidelines.

- GA's work best if the optimal solution is a member of schemata which have few defined bits that are packed tightly together, and the involved schema have a low-standard deviation. That is, since GA's search primarily by combining pieces of good chromosomes, codings should be devised so that similar chromosomes have similar objective function values. In this way, the GA is not deceived in creating chromosomes which are much worse than the sum of their parts.
- For most practical applications, a population size proportional to the chromosome length should suffice [19], although there is never any harm in using a larger population. The implicit parallelism theorem implies that doubling the population may more than halve the run time; thus, if the objective function calculation is fast enough for the use of a large population to be feasible, a large population should be used.

Because of the incomplete nature of GA theory, much knowledge about the successful implementation of GA's comes from experience and experiment.

- For binary-coded GA's, the crossover and mutation rates most often quoted in the literature are $0.6 \leq p_{\mathrm{cross}} \leq 0.9$, and $0.001 \leq p_{\mathrm{mut}} \leq 0.01$. Generally, p_{mut} corresponds to at most the mutation of one or two alleles per chromosome and at least a few chromosomes per generation. Real-coded GA's often use higher mutation rates.
- Problems with real design parameters are often (but not always) better handled by real-coded GA's. In general, codings for physical problems work best if they resemble the parameter they model.
- The inclusion of a local optimizer to either: 1) seed the initial population; 2) optimize the intermediate populations; or 3) touch up the final result, can speed the optimization process by removing the local refinement burden from the GA.
- Binary tournament selection generally works faster than roulette-wheel selection, and it avoids convergence troubles [20].
- When the GA stagnates, raising the mutation rate often yields useful information; if the objective function value of the best design changes significantly, the GA may have been temporarily stuck in a local optimum and continued optimization is likely to be useful. If not, restarting the GA with a new initial population will yield a better chance for improvement than attempting to push further.
- Many difficult problems will succumb to GA optimization after the inclusion of an advanced operator. Sharing (described in Section IV) often induces the GA to produce novel results [21].

III. Genetic Algorithms in Electromagnetics: A Retrospective

Since the early part of this decade, GA's have been applied with growing frequency to the design of electromagnetic devices of increasing complexity. This section presents a review of research in this area since 1992. Because of the rapid growth of GA applications in electromagnetics, this review cannot be complete. Though the applications reviewed come from all areas of electromagnetic research, for the purpose of this review, we divide them into four categories: antennas, stratified medium structures, static devices, and miscellaneous.

A. Genetic Algorithm Antenna Design

Antenna design (in particular, array design) is the most studied application of GA's in electromagnetics. A plethora of studies have investigated GA-based methods of reducing the sidelobes of an array by thinning [22]–[26], amplitude or phase tapering [27]–[35], or element position perturbations [36]. Closely related to this problem is reducing the scattering of strip arrays by thinning or perturbation [28], [37]. Most of these studies involve digital parameters and, thus, are textbook cases for binary or other finite alphabet GA's. For instance, an array built with ten 3-b phase shifters can be encoded into a chromosome of 30-b, composed of a concatenation of ten 3-b phase shifter settings. Additionally, many of these studies draw on the implementation guidelines listed in Section II. For example, in [22], Buckley combines the GA technique with a local optimizer which is used to seed the GA. GA's have also designed arrays with specified null locations [38] or a specific pattern shape [39].

Besides arrays, single antenna elements have been designed as well. For example, in [40]–[44], a technique is described for the fast GA optimization of wire antennas loaded with passive resonant circuits (RLC) to broaden their frequency response. Reflector antennas [45], [46] and Yagi antennas [47] have also been optimized by GA's.

Perhaps the most intriguing application of GA's to antenna design is the so-called "genetic antenna" [48]. In this study, a wire antenna was designed to radiate circularly polarized (CP) waves by choosing endpoints for straight wire segments comprising the antenna and then "connecting the dots." The GA produces counterintuitive antennas with radical bends, but they do indeed radiate CP waves over a considerable frequency band.

B. Design of Layered Electromagnetic Devices

Among the earliest GA studies in electromagnetics is the design of multilayered optical filters by minimizing the difference between the observed and desired filter characteristic [15], [49], [50]. Because [15] realizes filters by layering two preselected materials, the only design parameters considered are layer thicknesses which are real valued and arrayed into a real-coded chromosome. In accordance with the previous section's guidelines, the authors make the observation that a real-coded GA outperforms a binary coded GA for this problem. Very similar to the design of optical filters is the design of absorbers consisting of a perfect electric conductor (PEC) coated with lossy materials to minimize reflection of impinging waves [51]–[54]. Despite the similarities among all of these devices, the GA's used to optimize them might be quite different depending on the nature of the problem. For instance, because [52] chooses materials from a finite database, an allelic chromosome is used instead of the real coding mentioned above.

Computationally more expensive than the above problems is the design of frequency selective surfaces (FSS's) [55], [56]. The optimization can proceed identically to that of layered optical filters or absorbers, except with a more complicated chromosome and time-consuming objective function evaluation. Due to the variety of design parameters encountered in these problems and the large computation time required, [57] uses hybrid chromosomes with both real and allelic parts to optimize the chromosome structure as much as possible.

C. Applications of Genetic Algorithms in Statics

Many applications of GA's involve shaping magnetic pole pieces or insulators to produce a desired magnetic or electric field distribution in a given region of space [58], [59]. Typically, chromosomes describe the location of several points in space which are interpolated to produce the shape of the device under consideration. The objective function to be minimized is some norm of the vector difference between the field produced by the device and the desired field at several points.

One notable study of this sort is [60], in which the authors optimize a pot-core transformer not only to produce a specific field pattern, but to minimize the device dimensions as well. The inclusion of a second design goal renders this problem difficult and poorly defined. As the stated goals are physically incommensurable, they are combined in an *ad hoc* manner to achieve a palatable result. This technique is very common in optimization, but may lead to suboptimal results or extensive tinkering to find a suitable algebraic combination of objectives.

Perhaps even more interesting is the application of GA's to magnetostatic inverse problems [61]–[63]. Such problems easily fit the GA mold; the unknowns (for instance the location, orientation, and size of a crack [61]) are encoded into chromosomes, and the objective function to be minimized is the difference between the field produced by the configuration described in the chromosome and the desired distribution. A last application is found in [64], where coils are designed to create fields sensitive to cracks in aircraft.

D. Miscellaneous Applications of Genetic Algorithms to Electromagnetics

In addition to the application areas mentioned above, GA's have been applied to the design of wireless communication networks [65], lenses [66], waveguide junctions [67], dielectric gratings [68], and RF coils [69].

Outside the realm of design, GA's have been used for solving two-dimensional inverse scattering problems [70], to extract resonances from data sets [71], [72], to produce K pulses for target identification [73], in scattering center analyses of radar targets [74], and to generate super-resolved spatial excitation patterns in two-level systems [75]. They have even been applied to the optimization of perfectly matched layers for finite-difference applications [76].

IV. Design Examples Using Advanced GA Operators

Most of the studies cited in Section III follow the basic GA pattern outlined in Section II with little or no deviation. However, as GA's are applied to electromagnetic design problems of increasing complexity, they will need to be augmented to optimize faster or more globally. This can

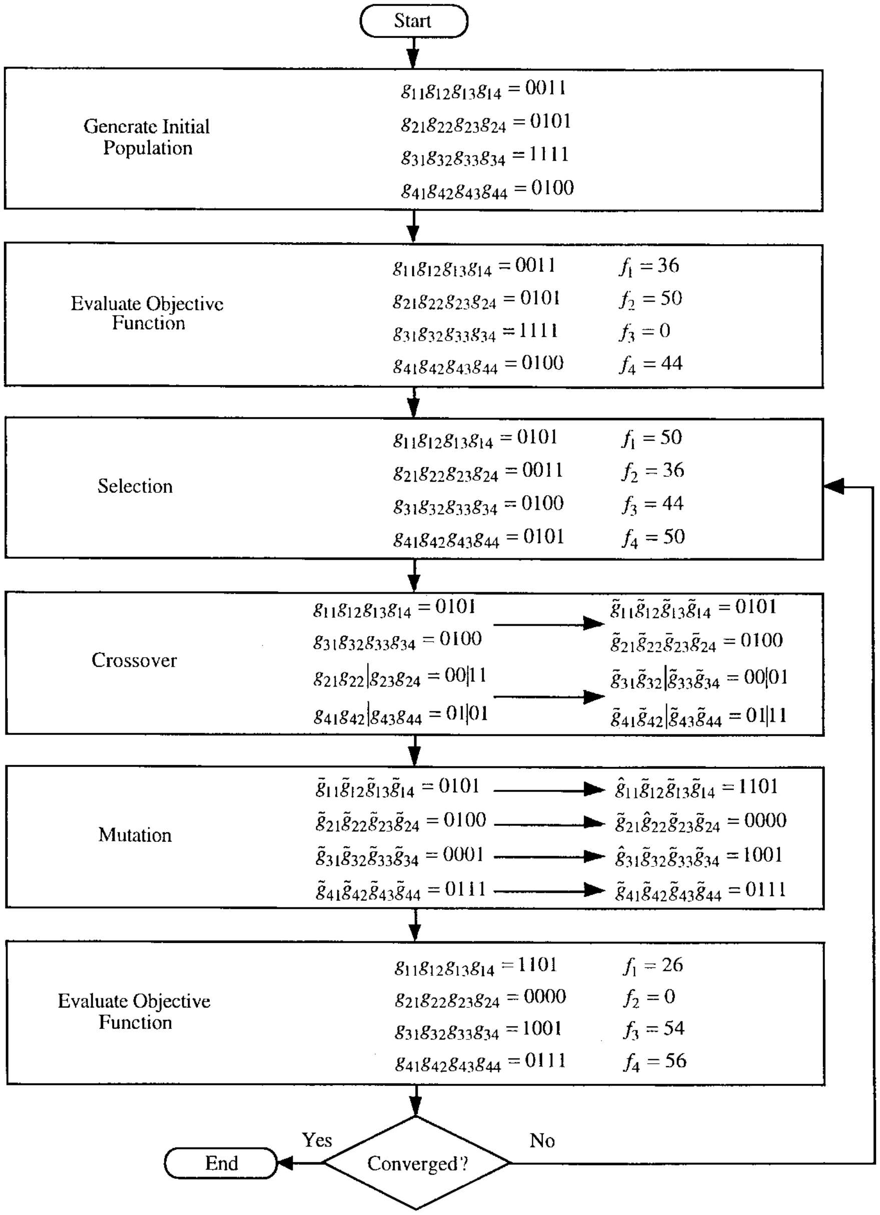

Start
Generate Initial Population
$g_{11}g_{12}g_{13}g_{14} = 0011$
$g_{21}g_{22}g_{23}g_{24} = 0101$
$g_{31}g_{32}g_{33}g_{34} = 1111$
$g_{41}g_{42}g_{43}g_{44} = 0100$
Evaluate Objective Function
$g_{11}g_{12}g_{13}g_{14} = 0011$ $f_1 = 36$
$g_{21}g_{22}g_{23}g_{24} = 0101$ $f_2 = 50$
$g_{31}g_{32}g_{33}g_{34} = 1111$ $f_3 = 0$
$g_{41}g_{42}g_{43}g_{44} = 0100$ $f_4 = 44$
Selection
$g_{11}g_{12}g_{13}g_{14} = 0101$ $f_1 = 50$
$g_{21}g_{22}g_{23}g_{24} = 0011$ $f_2 = 36$
$g_{31}g_{32}g_{33}g_{34} = 0100$ $f_3 = 44$
$g_{41}g_{42}g_{43}g_{44} = 0101$ $f_4 = 50$
Crossover
$g_{11}g_{12}g_{13}g_{14} = 0101$
$g_{31}g_{32}g_{33}g_{34} = 0100$
$g_{21}g_{22}|g_{23}g_{24} = 00|11$
$g_{41}g_{42}|g_{43}g_{44} = 01|01$
$\tilde{g}_{11}\tilde{g}_{12}\tilde{g}_{13}\tilde{g}_{14} = 0101$
$\tilde{g}_{21}\tilde{g}_{22}\tilde{g}_{23}\tilde{g}_{24} = 0100$
$\tilde{g}_{31}\tilde{g}_{32}|\tilde{g}_{33}\tilde{g}_{34} = 00|01$
$\tilde{g}_{41}\tilde{g}_{42}|\tilde{g}_{43}\tilde{g}_{44} = 01|11$
Mutation
$\tilde{g}_{11}\tilde{g}_{12}\tilde{g}_{13}\tilde{g}_{14} = 0101$
$\tilde{g}_{21}\tilde{g}_{22}\tilde{g}_{23}\tilde{g}_{24} = 0100$
$\tilde{g}_{31}\tilde{g}_{32}\tilde{g}_{33}\tilde{g}_{34} = 0001$
$\tilde{g}_{41}\tilde{g}_{42}\tilde{g}_{43}\tilde{g}_{44} = 0111$
$\hat{g}_{11}\tilde{g}_{12}\tilde{g}_{13}\tilde{g}_{14} = 1101$
$\tilde{g}_{21}\hat{g}_{22}\tilde{g}_{23}\tilde{g}_{24} = 0000$
$\hat{g}_{31}\tilde{g}_{32}\tilde{g}_{33}\tilde{g}_{34} = 1001$
$\tilde{g}_{41}\tilde{g}_{42}\tilde{g}_{43}\tilde{g}_{44} = 0111$
Evaluate Objective Function
$g_{11}g_{12}g_{13}g_{14} = 1101$ $f_1 = 26$
$g_{21}g_{22}g_{23}g_{24} = 0000$ $f_2 = 0$
$g_{31}g_{32}g_{33}g_{34} = 1001$ $f_3 = 54$
$g_{41}g_{42}g_{43}g_{44} = 0111$ $f_4 = 56$
Yes
No
End
Converged?

Fig. 1. A flowchart illustrating the genetic algorithm process optimizing the function $f(x) = x(15 - x)$. A 4-b chromosome is used, which decodes to an integer x between 0 and 15. After the initial population is generated and evaluated, selection eliminates the inferior chromosome 1111 and replaces it with 0101, which has a very high objective value. The chromosomes $\mathbf{p}_2$ and $\mathbf{p}_4$ then undergo crossover between the second and third genes, whereas the other chromosomes are not chosen to be crossed. Finally bits in the first three chromosomes are mutated, and the process begins anew by evaluating the objective function values of the new population.

be accomplished by using advanced GA techniques such as dominance and diploidy, community structure, sharing, or knowledge-based operators that can vastly improve GA performance at the expense of very small modifications of the algorithm [7]. Thus, we present three examples of advanced GA's that further tap the potential of the GA technique and also serve to give concrete examples of GA optimization at work. All of the examples presented use the standard binary coding [including the decoding of real parameters given in (3)] and one point crossover.

A. Genetic Algorithm Design of Broadband Loaded Wire Antennas in a Complex Environment

The most critical issue in creating a successful GA involves the coding of the chromosomes which must be designed so the GA is not misled. We illustrate this by applying a simple GA to the design of a wire antenna situated in an arbitrary, possibly complex, environment for operation over a broad range of frequencies [41]. The wire is to be augmented with a given number of parallel RLC loads whose locations and element values are to be determined by the GA to produce maximum gain over the frequency band of interest. In addition, the GA simultaneously designs a matching network to achieve an acceptable standing wave ratio (SWR) over the band. As GA's typically require many objective function evaluations, a generalization of the Sherman–Morrisson updating scheme [44] using numerical Green's functions was developed, which permits the fast analysis of loaded structures residing on a complex platform [41].

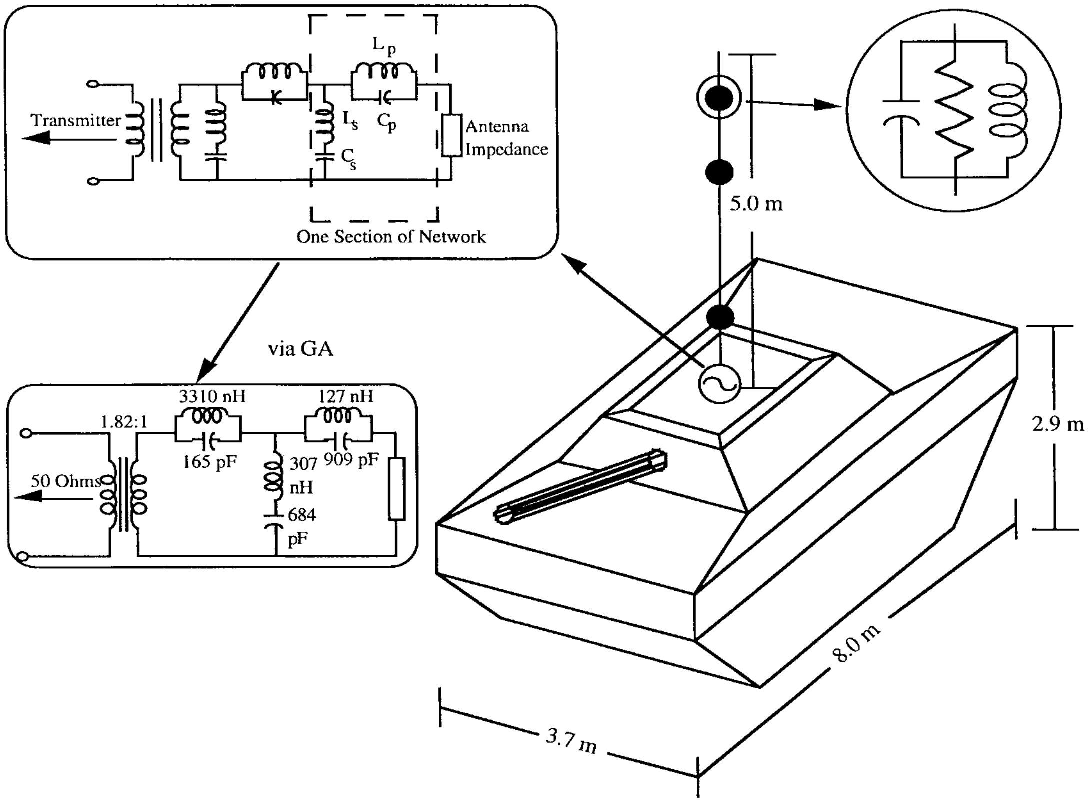


Fig. 2. An illustration of the broadband antenna design problem. The top inset is a general matching network topology, as described in Section III-A, and the bottom inset is an actual matching network optimized by the GA.

At first glance, it appears that this antenna can be optimized in a straightforward manner after incorporating load locations and element values into a chromosome using (3). The difficulty, however, lies in the design of the matching network. For a given network topology, encoding parameter values is straightforward, but often the topology itself requires optimization.

A simple way to optimize network topology is to encode all matching network component values with two extra control bits. The first of these bits could indicate whether the element in question is a capacitor or an inductor, and the second whether the element is series or shunt connected. The matching network could then be built element by element from the antenna back to the source. This would lead to a chromosome structured as follows. The first part of the chromosome would contain descriptions of the antenna loads and locations spanning the first N_{ant} bits. If the antenna has 2^{N_l} possible load locations, and N_{RLC} RLC loads with components that can be adequately described by N_c bits

$$N_{\text{ant}} = N_{\text{RLC}} \times (N_l + 3N_c). \tag{12}$$

Describing the matching network would require

$$N_{mn} = N_e \times (2 + N_c) \tag{13}$$

bits, where N_e is the number of elements in the matching network. A full antenna-matching network system would then have a chromosome of $N_{\text{ant}} + N_{mn}$ bits. This approach fails, however, as it violates an important requirement of the GA—namely, that designs with similar descriptions behave similarly. This requirement is a direct consequence of the schema theorem described in Section II. A design in the population with a good objective function value may be rendered useless if mutation happens to change a key capacitor in the matching network to an inductor.

A more GA friendly approach assumes a general matching-network template, as shown in Fig. 2. By allowing the most extreme values of the groups of genes describing the capacitances and inductances in the network to take on values of zero and infinity, elements may slowly evolve into shorts or opens as needed. The number of elements in the network may be kept to a minimum by penalizing designs with an excessive number of elements. The matching network part of the gene would now require

$$N_{mn} = 4N_s \times (N_c + N_d) \tag{14}$$

bits where N_s is the number of matching-network sections, and N_d is the number of bits added to allow for elements with zero and infinity values. The antenna loading part of the chromosome is unaltered.

With this coding, the problem becomes a straightforward application of a simple GA. The objective function to be maximized involves a sum of the system gain and penalties described more fully in [41] and [77]. As a demonstration of the technique, a whip residing on top of a tank was designed to radiate between 9 and 60 MHz with maximum gain at the horizon on the side of the tank (Fig. 2). This problem involves a 7 : 1 band, and the antenna is only 3/20 of a wavelength long at 9 MHz. Antennas for use over bandwidths of up to

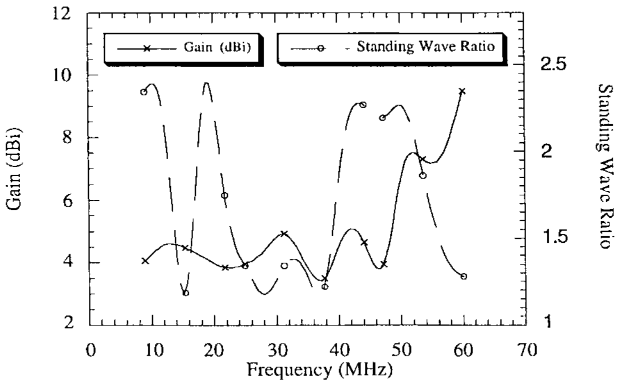


Fig. 3. The gain and SWR of the GA optimized antenna.

TABLE I
THE MATCHING NETWORK, LOAD LOCATIONS, AND COMPONENT VALUES SELECTED BY THE GA

Load Information for Tank Monopole				
Load Number	Distance from source (cm)	Resistance (Ω)	Capacitance (pF)	Inductance (nH)
1	369.	66.9.	103.	1830.
2	331.	168.	49.1	695.
3	361.	356.	107.	150.

15 : 1 have also been optimized with this technique [44]. The specific GA implemented to accomplish this used stochastic binary tournament selection with $N_{\text{RLC}} = 3$, $N_l = N_c = 7$, $N_s = 2$, and $N_d = 1$, resulting in a 148-b chromosome. The GA parameters used were $N_{\text{pop}} = 500$, $p_{\text{cross}} = 0.9$, and $0.0001 \leq p_{\text{mut}} \leq 0.005$. Capacitance, inductance, and resistance values were allowed to vary in between 1 and 110 pF, 3 and 5000 nH, and 1 and 1500 Ω, respectively. The optimized antenna gain and SWR after 100 generations are shown in Fig. 3. The matching network, load locations, and component values selected by the GA are shown in Fig. 3 and Table I, respectively.

B. Multiobjective Optimization of Broadband Absorbers

Often engineers face the task of satisfying conflicting goals in a single design. The problem then is not finding a design that is optimal with respect to a single design goal, but a design which makes tradeoffs between conflicting goals. Whenever this occurs, a multiobjective optimization technique is required. GA's are ideal for such problems because their reliance on a population allows them to return a wide variety of designs representing optimal tradeoffs.

A number of studies have focused on the design of microwave absorbers comprised of layered lossy materials with PEC backing (Fig. 4) [51], [52]. Most of these studies employ simple binary-coded GA's with chromosomes consisting of two groups of bits per layer describing each layer's material and thickness. Materials are often selected from a database [52], and the objective function is coded to minimize reflection coefficients R of TE and TM waves over a range of frequencies and angles of interest. But, besides suppressing reflection, absorbers often also have to be thin or light to be useful. To balance these competing goals, one might attempt to minimize an algebraic combination of goals $R + \alpha t$, where t represents the total thickness of the absorber. Unfortunately, finding a suitable algebraic combination of goals for use as an objective function can be a monumental task.

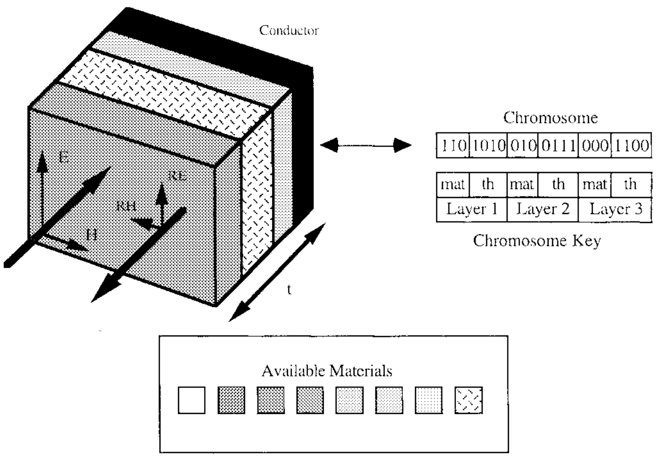


Fig. 4. A schematic of the broadband microwave absorber synthesis problem. The chromosome layout is also shown for a problem of three layers with a database of eight materials and 16 possible thickness values.

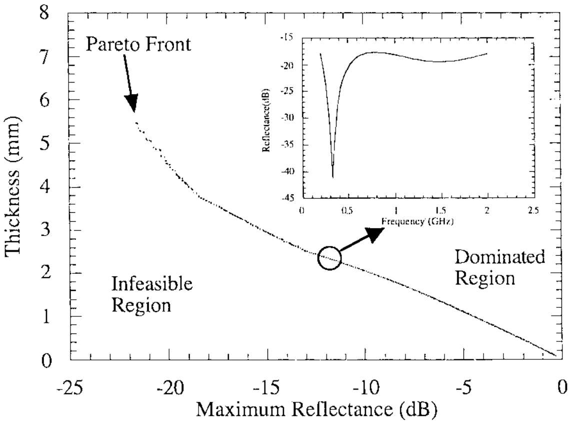
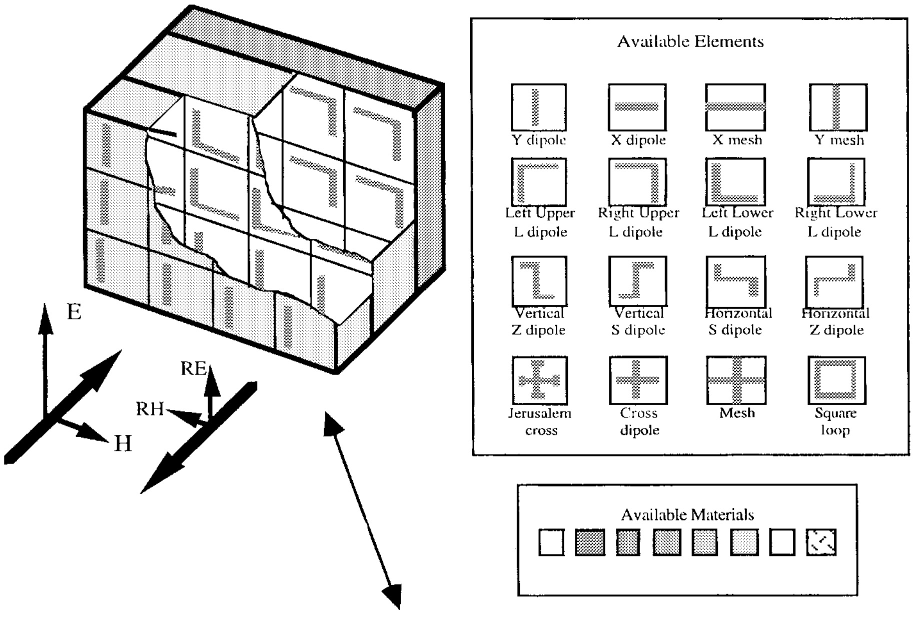


Fig. 5. The Pareto front constructed by the GA for the absorber problem. Inset is the frequency response of a particular design on the front for TE waves incident at 30°.

Pareto GA's avoid this problem entirely by returning not a single optimal design, but the set of designs that represents all of the best tradeoffs [78]–[80]. Within the context of the absorber design, this set can be defined as follows: 1) a design is termed *dominated* if there exists another design in the space of all possible designs which is thinner and less reflective; 2) a design which is not dominated is known as *nondominated, efficient,* or *Pareto optimal*; and 3) the *Pareto front* is the set of all nondominated designs, each of which is considered to be equally optimal. The simple GA, described in Section II, must be altered in two fundamental ways to return the Pareto front instead of a single point: 1) designs must be evaluated on the basis of their proximity to the Pareto front and 2) the GA must prevent the population from converging to a small subset of the front instead of the whole front.

Proper design evaluation is achieved using a scheme known as *nondomination ranking* [7], [81]. The population is ranked



Chromosome

0111	0101	100011	111	1010	0111	001011	011	1110	1001	111010	100	1010

cell	sh	dim	mat	th	sh	dim	mat	th	sh	dim	mat	th
Glob	Layer 1				Layer 2				Layer 3			

Chromosome Key

Fig. 6. The FSS design problem, along with a fairly general-shape database and a chromosome structure. The example pictured uses 4 b to specify the cell size (a global design parameter). The design parameters characterizing each layer are shape number from a 16-element database, shape dimension, and dielectric layer material and thickness.

by first removing all nondominated members and assigning them a rank of one. This process is repeated on the population remaining—its nondominated members are assigned a rank of two, and so on. To force the population members to spread out uniformly over the front, Pareto GA's use the *sharing* technique based on competition between organisms for environmental resources. In sharing, the objective function value of each population member is divided by a *niche count* which is a measure of the population density in its vicinity; that is, the niche count is high for designs that are very much like many others in the population. This disallows population bunching since chromosomes tending to group will be forced to die off [7], [82].

The Pareto GA combines nondomination ranking and sharing. In each generation, rank-one designs are assigned a common objective value and then shared. The lowest shared value of the rank-one designs is taken as the unshared objective function value for rank-two designs, and so on. This algorithm, described in more detail in [79], [81], was applied to the synthesis of five-layer absorbers with materials from the database in [52]. Tradeoffs were found between the maximum reflection for both polarizations between 0.2 and 2 GHz and 0 and 30° and the thickness of the absorber. For the results shown in Fig. 5, $N_{\text{pop}} = 8000$, $p_{\text{cross}} = 0.9$, and $p_{\text{mut}} = 0.005$. Because of well-known problems in GA's using sharing with tournament selection, roulette-wheel selection was used [83]. This Pareto GA has since been applied to the simultaneous optimization of beamwidth and sidelobe level in antenna arrays as well [84].

C. Community GA's for Periodic Structure Design

Wherever a GA is applied to a problem of great computational expense, it becomes necessary to accelerate either the device analysis, the GA convergence, or both. An example of a technique that achieves both is the community GA for optimizing the periodic structure shown in Fig. 6 [85]. This structure is not unlike the absorber described in the previous section, except that the materials used here are assumed lossless and a periodic metallization pattern is included in the layer interfaces. The GA not only chooses layer materials and thicknesses as before, but also the shape and size of the periodic metallizations, the periodic cell size, and the relative offset between metallization in different layers to achieve a desired transmission or reflection characteristic.

The analysis in this study proceeds via a spectral Galerkin technique [86] and, therefore, involves Floquet harmonics with transverse wave vectors given by

$$\begin{aligned}\mathbf{k}_t^{nm} &= \left(\frac{2\pi n}{\Delta x} - k_{xo}\right)\hat{\mathbf{x}} + \left(\frac{2\pi m}{\Delta y} - k_{yo}\right)\hat{\mathbf{y}} \\ &= k_x^{nm}\hat{\mathbf{x}} + k_y^{nm}\hat{\mathbf{y}} \end{aligned} \tag{15}$$

where Δx and Δy are the size of the periodic cell, k_{xo} and k_{yo} are the transverse components of the incident field wavevector, and m and n are integers. Now, envision a *community* of

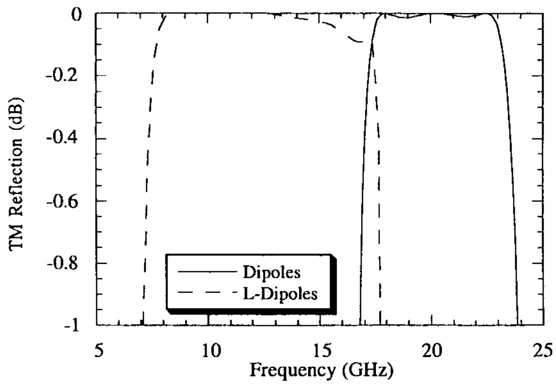


Fig. 7. Polarization response of two twist polarizers. The narrow band polarizer was designed with dipoles for an incident wave of $\theta = 5^o$ and $\phi = 45^o$, and the broadband polarizer was designed with L-dipoles for and incident wave of $\theta = 5^o$ and $\phi = 0^o$.

designs, all of which are identical, except in the thickness of their dielectric layers which are all nonetheless greater than some value $l_{\min}$. For those modes for which

$$\exp\left[-jl_{\min}\sqrt{k_o^2 - (k_x^{nm})^2 - (k_y^{nm})^2}\right] \ll 1 \qquad (16)$$

the interactions between different metallized layers can be safely ignored, and the intralayer interactions can be computed assuming that the metal is located between two dielectric half-spaces with dielectric constants equal to those of the surrounding layers. More importantly, interactions for which condition (2) is true can also be reused for the entire community.

This suggests a community structure for the GA which begins by creating a population of N_{popc} community chromosomes which describe element shapes, positions, etc. Each community chromosome is associated with a community of N_{pop} individual chromosomes which describe only dielectric slab thicknesses. The GA proceeds by first applying all of its operators to the community chromosomes (assuming for crossover that the individual chromosomes are attached to the right of the community ones). Then, it runs several generations operating on the individual chromosomes within the communities. Community structures are not unheard of in GA optimization [7], but they are usually used only to accelerate the convergence of the GA. Here, communities both accelerate analysis and convergence.

Using this technique, two different twist polarizers were designed to reflect incident TE waves as TM waves. GA parameters for this problem included $N_{\text{popc}} = 70$, $N_{\text{pop}} = 20$, $p_{\text{cross}} = 0.8$, and $0.001 \leq p_{\text{mut}} \leq 0.1$. The relatively high mutation rates result from the computational difficulty of this problem; they tend to make the GA converge faster, even if to a slightly suboptimal solution. Shapes were picked from the database shown in Fig. 6, the responses of the two designs are shown in Fig. 7, with the designs themselves described in Tables II and III.

TABLE II
DIPOLE TWIST POLARIZER DESIGN

Dipole Twist Polarizer Design		
Dielectric Layer	Dielectric Constant	Thickness (cm)
1	1.5	0.383
2	2.5	0.393
Metal Layer, Shape	Dipole Length (cm)	Offsets (cm)
1, X-dipole	0.642	X: 0.481, Y: 0.234
2, Y-dipole	0.257	X: 0.352, Y: 0.035
Cell Size = 0.745 cm		

TABLE III
L-DIPOLE TWIST POLARIZER DESIGN

L-Dipole Twist Polarizer Design		
Dielectric Layer	Dielectric Constant	Thickness (cm)
1	1.25	0.420
2	4.00	0.326
Metal Layer, Shape	Dipole Length (cm)	Offsets (cm)
1, Right lower L-dipole	0.453	X: 0.555, Y: 0.203
2, Left lower L-dipole	0.577	X: 0.215, Y: 0.298
Cell Size = 0.758 cm		

V. CONCLUSION

GA's show great promise in electromagnetic device design and problem solution. They have been applied successfully to the synthesis of novel antennas, inexpensive arrays, filters, absorbers, and FSS's, and have been used to tackle inverse scattering problems. GA's have been employed as electromagnetic optimizers because of their natural advantages over DOM's—they are highly suited for optimizing multimodal problems, and operate with ease on discrete parameters. They require no initial guesses to start the optimization process and often give rise to counterintuitive designs. Though often slower than DOM's, the speed of GA's is increasing due to several factors. GA's are increasingly being hybridized with other optimization techniques, either stochastic [87] or deterministic [88] in nature. Furthermore, as parallel computing resources become more widespread, more researchers will take advantage of the innate parallelism of the GA structure as objective function values of whole populations of designs can be calculated in parallel [56]. Finally, the application of messy GA's that overhaul the basic structure of the simple GA cannot be far off [89].

ACKNOWLEDGMENT

The authors would like to thank J. Veihl for his contribution of the mesh for Fig. 2.

REFERENCES

[1] C. Darwin, *The Origin of Species.* London, U.K.: Penguin, 1985.

[2] G. Mendel, *Experiments in Plant Hybridization.* Cambridge, MA: Harvard Univ. Press, 1950.

[3] R. Fletcher, *Practical Methods of Optimization.* Chichester, U.K.: Wiley Intersci., 1980.

[4] L. C. W. Dixon, *Global Optima Without Convexity, in Design and Implementation of Optimization Software.* Sijnoof Noordhof, The Netherlands: Aalphen aan den Rijn, 1978.

[5] L. Davis, *Genetic Algorithms and Simulated Annealing.* London, U.K.: Pittman, 1987.

[6] K. A. DeJong, "An analysis of the behavior of a class of genetic adaptive systems," Ph.D. dissertation, Univ. Michigan, 1975.

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

[8] J. H. Holland, *Adaptation in Natural and Artificial Systems.* Ann Arbor, MI: Univ. Michigan, 1975.

[9] ———, "Genetic algorithms," *Sci. Amer.*, vol. 267, pp. 66–72, July 1992.

[10] S. Forrest, "Genetic algorithms: Principles of natural selection applied to computation," *Sci.*, vol. 261, pp. 872–878, Aug. 1993.

[11] S. Begley, "Software au naturel," *Newsweek*, pp. 70–71, May 8, 1995.

[12] R. A. McIntyre, "Bach in a box: The evolution of four part Baroque harmony using the genetic algorithm," in *Proc. IEEE Conf. Evolutionary Computat.*, Orlando, FL, June 1994, pp. 852–857.

[13] J. Ventrella. "Disney meets Darwin—The evolution of funny animated figures," in *Proc. Comput. Animat.*, Geneva, Switzerland, Apr. 1995, pp. 35–43.

[14] D. E. Goldberg, "Real-coded genetic algorithms, virtual alphabets and blocking," *Complex Syst.*, vol. 5, no. 2, pp. 129–167, 1991.

[15] E. Michielssen, S. Ranjithan, and R. Mittra, "Optimal multilayer filter design using real coded genetic algorithms," *Proc. Inst. Elect. Eng., pt. J.*, vol. 139, no. 12, pp. 413–420, 1992.

[16] K. Deb and D. E. Goldberg, "Sufficient conditions for deceptive and easy binary functions," *Ann. Mathem. Artificial Intell.*, vol. 10, pp. 385–408, 1994.

[17] ———, "Analyzing deception in trap functions," in *Foundations of Genetic Algorithms 2,* L. D. Whitley, Ed. San Mateo, CA: Morgan Kauffman, 1993.

[18] D. E. Goldberg, "Genetic algorithms and Walsh functions: Part I—A gentle introduction," *Complex Syst.*, vol. 3, no. 2, pp. 129–152, 1989.

[19] D. E. Goldberg, K. Deb, and J. H. Clark, "Genetic algorithms, noise, and the sizing of populations," *Complex Syst.*, vol. 6, no. 3, pp. 333–362, 1991.

[20] D. E. Goldberg and K. Deb, "A comparative analysis of selection schemes used in genetic algorithms," in *Foundations of Genetic Algorithms*, G. J. E. Rawlins, Ed. San Mateo, CA: Morgan Kaufmann, 1991.

[21] D. E. Goldberg, personal communication, 1995.

[22] M. J. Buckley, "Linear array synthesis using a hybrid genetic algorithm," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Baltimore, MD, July 1996, pp. 584–587.

[23] R. L. Haupt, J. J. Menozzi, and C. J. McCormack, "Thinned arrays using genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Ann Arbor, MI, June 1993, pp. 712–715.

[24] R. L. Haupt, "Thinned arrays using genetic algorithms," *IEEE Trans. Antennas Propagat.*, vol. 42, pp. 993–999, July 1994.

[25] ———, "An introduction to genetic algorithms for electromagnetics," *IEEE Antennas Propagat. Mag.*, vol. 37, pp. 8–15, Apr. 1995.

[26] D. J. O'Neill. "Element placement in thinned arrays using genetic algorithms," in *Proc. OCEANS '94,* Brest, France, Sept. 1994, pp. II/301–306.

[27] F. Ares, S. R. Rengarajan, E. Villanueva, E. Skochinski, and E. Moreno, "Application of genetic algorithms and simulated annealing technique in optimising the aperture distributions of antenna array patterns," *Electron. Lett.*, vol. 32, no. 3, pp. 148–149, 1996.

[28] R. L. Haupt and A. S. Ali, "Optimized backscattering sidelobes from an array of strips using a genetic algorithm," in *Proc. Appl. Computat. Electromagn. Conf.*, Monterey, CA, Mar. 1994, pp. 266–270.

[29] R. L. Haupt, "Optimum quantised low sidelobe phase tapers for arrays," *Electron. Lett.*, vol. 31, no. 14, pp. 1117–1118, 1995.

[30] ———, "Comparison between genetic and gradient-based optimization algorithms for solving electromagnetics problems," *IEEE Trans. Magn.*, vol. 31, no. 3, pp. 1932–1935, May 1995.

[31] ———, "Speeding convergence of genetic algorithms for optimizing antenna arrays," in *Proc. 12th Ann. Rev. Progress Appl. Computat. Electromagn.*, Moneterey, CA, Mar. 1996, pp. 742–749.

[32] Y. Lu and K. K. Lan, "Genetic algorithms based pattern synthesis approach for arbitrary array designs," in *Proc. 12th Ann. Rev. Progress Appl. Computat. Electromagn.*, Monterey, CA, Mar. 1996, pp. 734–741.

[33] D. Marcano, F. Duran, and O. Chang, "Synthesis of multiple beam linear arrays using genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Newport Beach, CA, June 1995, pp. 938–941.

[34] M. Shimizu, "Determining the excitation coefficients of an array using genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Seattle, WA, June 1994, pp. 530–533.

[35] D. Marcano and A. Nieto, "Genetic algorithms for the synthesis of planar arrays," in *Proc. USNC/URSI Radio Sci. Meet.*, Baltimore, MD, July 1996, p. 11.

[36] R. Haupt. "Optimization of subarray amplitude tapers," in *Proc. IEEE Antennas Propagat. Society Int. Symp.*, Newport Beach, CA, June 1995, pp. 1830–1833.

[37] ———, "Optimization of aperiodic conducting grids," in *Proc. 11th Annu. Rev. Progress Appl. Computat. Electromagn.*, Monterey, CA, Mar. 1995, pp. 211–215.

[38] A. Tennant, M. M. Dawoud, and A. P. Anderson, "Array pattern nulling by element position perturbations using a genetic algorithm," *Electron. Lett.*, vol. 30, no. 3, pp. 174–176, 1994.

[39] D. Marcano, M. Jimenez, and O. Chang, "Synthesis of linear array using Schelkunoff's method and genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Baltimore, MD, July 1996, pp. 814–817.

[40] E. Michielssen, A. Boag, J. M. Sajer, and R. Mittra, "Design of electrically loaded wire antennas using massively parallel genetic algorithms," in *Proc. URSI Radio Sci. Meet.*, Seattle, WA, June 1994.

[41] D. S. Weile, E. Michielssen, and A. Boag, "Optimization of broad-band loaded wire antennas in real environments using genetic algorithms," in *Proc. 12th Ann. Rev. Progress Appl. Computat. Electromagn.*, Monterey, CA, Mar. 1996, pp. 726–733.

[42] Z. Altman, R. Mittra, J. Philo, and S. Dey, "New designs of ultrabroadband antennas using genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Baltimore, MD, July 1996, pp. 2054–2057.

[43] M. Bahr, A. Boag, E. Michielssen, and R. Mittra, "Design of ultrabroadband loaded monopoles," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Seattle, WA, June 1994, pp. 1290–1293.

[44] A. Boag, A. Boag, E. Michielssen, and R. Mittra, "Design of electrically loaded wire antennas using genetic algorithms," *IEEE Trans. Antennas Propagat.*, vol. 44, pp. 687–695, May 1996.

[45] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithms in electromagnetics," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Baltimore, MD, July 1996, pp. 1480–1483.

[46] ———, "Genetic algorithm optimization and its application to antenna design," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Seattle, WA, June 1994, pp. 326–329.

[47] D. S. Linden and E. E. Altschuler, "The design of Yagi antennas using a genetic algorithm," in *Proc. USNC/URSI Radio Sci. Meet.*, Baltimore, MD, July 1996, p. 283.

[48] ———, "Automating wire design using genetic algorithms," *Microwave J.*, vol. 39, no. 3, p. 74, 1996.

[49] T. Eisenhammer, M. Lazarov, M. Leutbecher, U. Schoffel, and R. Sizmann, "Optimization of interference filters with genetic algorithms applied to silver-based mirrors," *Appl. Opt.*, vol. 32, no. 31, pp. 6310–6315, 1993.

[50] S. Martin, J. Rivory, and M. Shoenauer, "Simulated Darwinian evolution of homogeneous multilayer systems: A new method for optical coatings design," *Opt. Communicat.*, vol. 110, pp. 503–506, 1994.

[51] A. Tennant and B. Chambers, "Adaptive optimization techniques for the design of microwave absorbers," in *Proc. Conf. Adaptive Computing Eng. Design Contr.*, Univ. Plymouth, U.K., Sept. 1994, pp. 44–49.

[52] E. Michielssen, J.-M. Sajer, S. Ranjithan, and R. Mittra, "Design of lightweight, broad-band microwave absorbers using genetic algorithms," *IEEE Trans. Microwave Theory Tech.*, vol. 41, pp. 1024–1031, June/July 1993.

[53] B. Chambers and A. Tennant, "Design of wideband Jaumann radar absorbers with optimum oblique incidence performance," *Electron. Lett.*, vol. 30, no. 18, pp. 1530–1531, 1994.

[54] ———, "Optimized design of Jaumann radar absorbing materials using a genetic algorithm," *Proc. Inst. Elect. Eng. Radar, Sonar, Navigat.*, vol. 143, no. 1, pp. 23–30, 1996.

[55] E. Michielssen, J. M. Sajer, and R. Mittra, "Design of multilayered FSS and waveguide filters using genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.*, Ann Arbor, MI, June 1993, pp. 1936–1939.

[56] E. Michielssen, A. Boag, J. M. Sajer, and R. Mittra, "Design of frequency selective surfaces using massively parallel genetic algorithms," in *Proc. URSI Radio Sci. Meet.*, Seattle, WA, June 1994, p. 441.

[57] K. Aygun, D. S. Weile, and E. Michielssen, "Design of multilayered strip gratings by genetic algorithms," *Microwave Opt. Tech. Lett.*, vol. 42, pp. 81–85, Feb. 1997.

[58] G. F. Uler, O. A. Mohammed, and C.-S. Koh, "Utilizing genetic algorithms for the optimal design of electromagnetic devices," *IEEE Trans. Magn.*, vol. 30, pp. 4296–4298, Nov. 1994.

[59] J. A. Vasconcelos, L. Karahenbuhl, L. Nicholas, and A. Nicholas, "Design optimization using the BEM coupled with genetic algorithm," in *Proc. 2nd Int. Conf. Computat. Electromagn.*, London, U.K., Apr. 1994, pp. 60–63.

[60] O. A. Mohammed and G. F. Uler, "Genetic algorithms for the optimal design of electromagnetic devices," in *Proc. 11th Annu. Rev. Progress Appl. Computat. Electromagn.*, Monterey, CA, Mar. 1995, pp. 384–392.

[61] A. A. Arkadan, T. Sareen, and S. Subramaniam, "Genetic algorithms for nondestructive testing in crack evaluation," *IEEE Trans. Magn.*, vol. 30, pp. 4320–4322, Nov. 1994.

[62] M. Enokizono and Y. Akinari, "Estimation of current distribution by a hybrid of genetic algorithm and sampled pattern matching method," *IEEE Trans. Magn.*, vol. 31, pp. 2012–2015, May 1995.

[63] D. McNay, E. Michielssen, R. L. Rogers, F. A. Taylor, M. Akhtari, and W. W. Sutherling, "Multiple source localization using genetic algorithms," *J. Neurosci. Methods.*, vol. 64, pp. 163–172, Feb. 1996.

[64] F. Thollon and N. Burais, "Geometrical optimization of sensors for eddy currents—Nondestructive testing and evaluation," *IEEE Trans. Magn.*, vol. 31, pp. 2026–2031, May 1995.

[65] J. M. Johnson and Y. Rahmat-Samii, "Genetic algorithm optimization of wireless communication networks," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.,* Newport Beach, CA, June 1995, pp. 1964–1967.

[66] E. Betensky, "Postmodern lens design," *Opt. Eng.,* vol. 38, no. 2, pp. 1750–1756, 1993.

[67] J. Kim, H.-B. Lee, C. Cheon, H.-S. Kim, H. K. Jung, and S.-Y. Hahn, "Numerical design technique for waveguide T-junctions in H-plane," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.,* Newport Beach, CA, June 1995, pp. 1562–1565.

[68] E. G. Johnson and M. A. G. Abushagur, "Microgenetic algorithm optimization methods applied to dielectric gratings," *J. Opt. Soc. Amer. A,* vol. 12, no. 5, pp. 1152–1160, 1995.

[69] N. Lu, J. Jin, E. Michielssen, and R. L. Magin, "Optimization of RF coil design using genetic algorithm and simulated annealing method," in *Proc. 3rd Meet. Soc. Magn. Resonance,* Nice, France, Aug. 19–25, 1995, p. 1002.

[70] C.-C. Chiu and P.-T. Lu, "Image reconstruction of a perfectly conducting cylinder by the genetic algorithm," *Proc. Inst. Elect. Eng. Microwaves, Antennas Propagat.,* vol. 143, no. 3, pp. 249–253, June 1996.

[71] P. Ilavarasan, E. J. Rothwell, R. Bebermeyer, K. M. Chen, and D. P. Nyquist, "Natural resonance extraction from multiple data sets using a genetic algorithm," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.,* Seattle, WA, June 1994, pp. 576–579.

[72] P. Ilavarasan, E. J. Rothwell, K.-M. Chen, and D. P. Nyquist, "Natural resonance extraction from multiple data sets using a genetic algorithm," *IEEE Trans. Antennas Propagat.,* vol. 43, pp. 900–904, Aug. 1995.

[73] G. Turhan-Sayan, K. Leblebicioglu, and S. Inan, "The use of genetic algorithms in input signal shaping for target identification," in *Proc. USNC/URSI Radio Sci. Meet.,* Baltimore, MD, July 1996, p. 21.

[74] Q. Li, E. J. Rothwell, K.-M. Chen, and D. P. Nyquist, "Scattering center analysis of radar targets using fitting scheme and genetic algorithm," *IEEE Trans. Antennas Propagat.,* vol. 44, pp. 198–207, Feb. 1996.

[75] E. Michielssen and D. Brady, "Control of spatial excitation patterns in two-level systems using time domain fields," *Opt. Lett.,* pp. 1931–1933, Dec. 1994.

[76] E. Michielssen, W. C. Chew, and D. S. Weile. "Genetic algorithm optimized perfectly matched layers for finite difference frequency domain applications," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.,* Baltimore, MD, July 1996, pp. 2106–2109.

[77] D. S. Weile, "Genetic algorithm applications in electromagnetics," Master's thesis, Univ. Illinois, Urbana-Champaign, Aug. 1995.

[78] D. S. Weile, E. Michielssen, and D. E. Goldberg, "Multiobjective synthesis of electromagnetic devices using genetic algorithms," in *Proc. IEEE Antennas Propagat. Soc. Int. Symp.,* Baltimore, MD, July 1996, pp. 592–596.

[79] ——, "Genetic algorithm design of Pareto optimal broad band microwave absorbers," *IEEE Trans. Electromagn. Compat.,* vol. 38, pp. 518–524, Aug. 1996.

[80] E. Michielssen, J. M. Sajer, and R. Mittra, "Pareto-optimal design of broadband microwave absorbers using genetic algorithms," in *Proc. IEEE Antennas Propagat. Int. Symp.,* Ann Arbor, MI, June 1993, pp. 1167–1170.

[81] N. Srinivas and K. Deb, "Multiobjective optimization using nondominated sorting in genetic algortihms," *Evolutionary Computat.,* vol. 2, no. 3, pp. 221–248, 1995.

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

[83] C. K. Oei, D. E. Goldberg, and S. J. Chang, "Tournament selection, niching, and the preservation of diversity," Univ. Illinois, Urbana-Champaign, IlliGAL Tech. Rep. 91011, 1991.

[84] D. S. Weile and E. Michielssen, "Integer coded Pareto genetic algorithm design of antenna arrays," *Electron. Lett.,* vol. 19, pp. 1744–1745, Sept. 1996.

[85] D. S. Weile, E. Michielssen, and A. Boag, "Community-based evolutionary optimization of frequency selective surfaces," in *Proc. USNC/URSI Radio Sci. Meet.,* Baltimore, MD, July 1996, p. 345.

[86] R. Mittra, C. H. Chan, and T. Cwik, "Techniques for analyzing frequency selective surfaces—A review," *Proc. IEEE,* vol. 76, pp. 1593–1615, Dec. 1988.

[87] A. E. Eiben, E. H. L. Aarts, and K. M. V. Hee, "Global convergence of genetic algorithms: On infinite Markov chain analysis," in *Proc. 1st Int. Conf. Parallel Problem Solving Nature,* Dortmund, Germany, Oct. 1991, pp. 4–12.

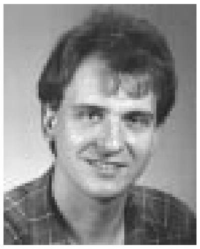
[88] J. Bosworth, N. Foo, and B. P. Zeigler, "Comparison of genetic algorithms with conjugate gradient methods," NASA-CR-2093; NTIS N72–30526, 1972.

[89] D. E. Goldberg, K. Deb, and B. Korb, "Messy genetic algorithms revisited: Studies in mixed size and scale," *Complex Syst.*, vol. 4, pp. 415–444, 1990.

Daniel S. Weile received the B.S. degree in electrical engineering and the B.S. degree in mathematics from the University of Maryland, College Park, in 1994, and the M.S. degree in electrical engineering from the University of Illinois at Urbana-Champaign, in 1995. He is currently working toward the Ph.D. degree at the University of Illinois at Urbana-Champaign.

From 1990 to 1994, he was a Technical Aide with the Army Space Technology and Research Office (ASTRO), Gaithersburg, MD, where he was officially commended three times for his work on reflection from inhomogeneous media. During the summer of 1994, he worked at the Institute of Plasma Research, University of Maryland, College Park, working on developing interactive software for the design of depressed collectors for gyrotron beams. Currently, he is a National Science Foundation Fellow and Teaching Assistant at the University of Illinois at Urbana-Champaign.

Mr. Weile is a member of Eta Kappa Nu, Tau Beta Pi, and Phi Beta Kappa.



Eric Michielssen (M'95) received the M.S. degree in electrical engineering from Katholieke Universiteit, Leuven, Belgium, in 1987, and the Ph.D. degree from the University of Illinois at Urbana-Champaign, in 1992.

He is an Assistant Professor of electrical and computer engineering at the University of Illinois at Urbana-Champaign and the Associate Director of its Center for Computational Electromagnetics. From 1987 to 1988, he was a Research Assistant in the Microwaves and Lasers Laboratory at Katholieke Universiteit, Leuven, Belgium (KUL). He joined the faculty of the Department of Electrical and Computer Engineering at the University of Illinois as a Visiting Assistant Professor in 1992, and was appointed an Assistant Professor of electrical and computer engineering in 1993. He serves as an Associate Editor for *Radio Science*. His research interests include all aspects of computational electromagnetics, with a focus on fast multilevel algorithms, the application of combinatorial stochastic optimization techniques to the design of electromagnetic and optical components, and computational photonics.

Dr. Michielssen serves as the Technical Chair for the 13th Annual Review of Progress in Applied Computational Electromagnetics, Monterey, CA. He was appointed Belgian American Educational Foundation Fellow in 1988 and is a 1995 NSF CAREER Award recipient.