Univerzitet u Beogradu
Publikacije Elektrotehničkog Fakulteta
Serija Tehnička fizika (1998), 47–58

Global Optimization Algorithms II Genetic Algorithms - Application to Calculation of Optical Constants of SiO_2

Aleksandra B. Djurišić

Dept. of EEE, University of Hong Kong, Pokfulam Road, Hong Kong

The genetic algorithms are described in detail. Special attention is given to the genetic algorithms designed for solving continuous optimization problems. Elite genetic algorithm with adaptive mutations, which was developed for model parameter estimation, is shown to be clearly superior over its classical counterpart on two families of test functions for 20, 50 and 100 variables. This algorithm is applied to determine parameters of the modified Lorentz model of optical constants of crystalline and amorphous silicon dioxide.

I Introduction

Genetic algorithms (GAs) [1] are stochastic global search methods that mimic the concept of natural evolution. Due to the nature of the algorithm, their successful application was mostly restricted to optimization problems whose solution can be conveniently represented in binary form. However, there is a rising interest in applying genetic algorithms to continuous optimization problems, especially since there is no need for initial estimates, which is important advantage of GAs over other stochastic search methods like, for instance, simulated annealing [2]. For that reason, various modifications of original GAs have been reported [3, 4, 5, 6, 7].

GAs search for optimal solution by employing mechanisms of natural evolution: selection, mating and mutation, which are applied to the set of possible problem solutions, called population. Each element of the population, called string or chromosome, is represented by vector of variables. Each element of the chromosome is, by analogy, termed a gene. Chromosomes are characterized with performance with respect to some objective function, called fitness. Highly fit individuals have a high probability of being selected for reproduction, i.e. they may survive or give offspring in the next generation. Strings with lower fitness have correspondingly lower probability for transferring their genes to following generations. This is illustrated on Fig 1, where shade of gray represents fitness - the lighter shade, the better gene.

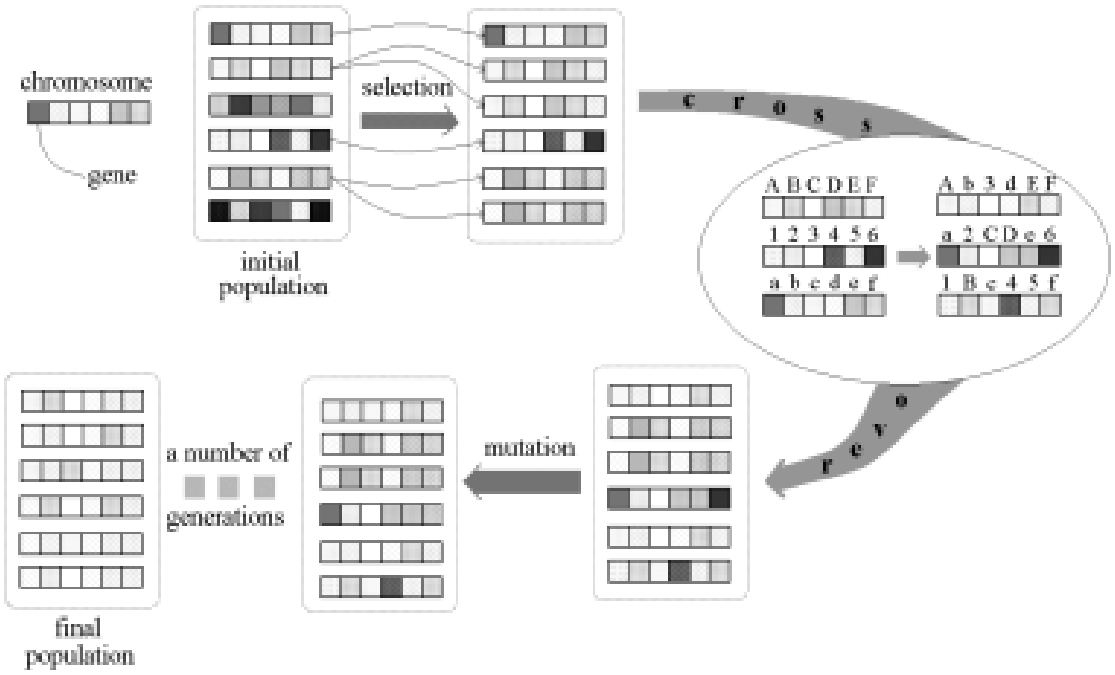

FIG. 1. Illustration of GA: shade of gray represents fitness: the lighter shade, the more fitted gene.

There exist many variations of GA [8], differing in chromosome representation, selection, reproduction and mutation, existence of fitness scaling etc. It was shown that floating-point number representation, instead of binary string representation common for classical GAs, is more convenient for continuous optimization applications [5, 6, 9]. By representing variable values with real numbers, length of the chromosome is equal to the number of variables, thus being much smaller than in the case of binary coding. Also, conversion of decimal numbers into binary ones and *vice versa* is avoided by using floating-point representation. What's more, important feature of floating-point representation is that variable values can not be altered or destroyed during crossover operation, while in the case of binary coding such undesired changes of variable values can lead to loss or deterioration of genetic information. Another important advantage of real-coded GAs is absence of Hamming cliff problem inherent to binary coded GAs [1, 9, 10]. Such problems could be overcome by the use of Gray coding, but that leads to higher nonlinearities with respect to recombination. Also, using real-coding reduces the dimensionality of the problem thus reducing the opportunity for deception, but possibilities for new obstacles to convergence are introduced. The fact that real-coded GAs can in certain cases be blocked from further progress has been recognized and discussed [9].

There has been much work in modifying the real-coded GAs in order to make them as successful in solving continuous optimization problems as they binary counterparts are in solving discrete optimization problems [3, 4, 10, 11]. Main shortcoming in continuous optimization applications of the GAs appears to be the discrete sampling of the solution space, which results in the fact that global minimum can be located only roughly. For obtaining the location of global minimum more precisely a huge number of the chromosomes in the population is required. Several methods have been proposed to overcome this difficulty. Obviously, if new values could be introduced during the optimization procedure, that would reduce the necessary number of chromosomes in the population for finding satisfactory solution of the problem. Traditional role of the mutation operators to introduce new values in order to prevent premature convergence to a local minimum.

However, mutation probabilities are usually very small, because otherwise algorithm might not converge at all. This is the main reason why mutation is usually considered to be of less importance than selection and crossover operators [1, 9]. Consequently, the work in development of real-coded GAs suitable for continuous optimization was concentrated on devising crossover operators suitable for real numbers. Various crossover operators for continuous variables, like intermediate or line recombination [4] or crossover by different continuous mixing functions [3], have been introduced. Deb and Agrawal [10] have suggested simulated binary crossover (SBX), which has similar search power to single-point crossover used in binary GAs. Main advantage of this crossover was demonstrated for cases when initial population was generated in bounds that do not bracket optimum variable value, since SBX operator enables search outside initial boundaries.

However, this feature represents important shortcoming for model parameter estimation, since there is no guarantee that obtained solution would be physically meaningful . Eshelman and Schaffer [11] suggested blend crossover operator BLX-α which randomly picks a point in the range $(p_1 - \alpha(p_2 - p_1), p_1 + \alpha(p_2 - p_1))$ and best results were reported for $\alpha = 0.5$. Common feature of these algorithms is that new values are introduced in the process of crossover, while mutation operator only slightly perturbs a parameter around its current value. These crossover operators were compared to conventional ones designed for binary-coded GAs , but only on test functions for up to 20 variables [4, 10]. However, for large number of variables, the strategy of introducing new values in the process of crossover fails to give good results. Djuričić *et al* [7] have suggested different approach in solving the problem, introducing the concept of parameter space size adjustment. The main idea was to narrow the solution space by improving the guess of the global minimum position. This is achieved by introducing one additional loop. Initial population is generated, classical GA is performed, and boundaries for each parameter are narrowed towards the obtained average value for that parameter in the inner loop final population. New initial population is generated in the new boundaries and GA is performed again. Outer loop is executed for a specified number of iterations n_{max} which means that required time for performing this algorithm is n_{max} times larger than for corresponding classical GA. Elite genetic algorithm with adaptive mutations (EGAAM) is also based on the similar concept of narrowing the parameter boundaries, but there is no observable increase of the CPU time requirements. In other words, executional time of EGAAM and GA are approximately the same, while the precision of EGAAM is significantly better.

Performance of the EGAAM is compared to classical GA on two families of multiminima test functions. After proving that EGAAM is reliable tool for solving continuous optimization problems, its ability to determine model parameter values on the basis of known experimental results is demonstrated. Model parameter estimation is performed by minimizing the objective function, usually in the form of sum of squared absolute or relative differences between calculated and experimental values, which often has multiple minima. In many areas, frequently arises the problem of difficulty or inability of providing initial estimates of parameter values. Sometimes not even order of magnitude of model parameters can be correctly guessed. In such cases, application of global optimization algorithm is imperative.

Let us discuss in more details model parameter estimation problem and the employed model. One of the frequently employed models for modeling the optical properties of solids is Lorentz oscillator model (LOM) or Lorentz-Drude model for metallic solids, where free-carrier absorption should be taken into account [7, 12, 13, 14, 15]. LOM assigns oscillators to major critical points (CPs) in joint density of states to model direct interband transitions, with some additional oscillators to model absorption between CPs. Each oscillator is characterized with oscillator strength, frequency and damping constant. In the model employed in this paper, additional parameter per oscillator is introduced in order to improce flexibility of the model, as will be described in detail. Initial values for oscillator frequencies can be estimated from band structure calculation, while for oscillator strengths and damping constants even an order of magnitude is unknown. Therefore, classical optimization techniques (simplex, Levenberg-Marqardt etc.) require initial estimates close to final values to obtain meaningful solution, cannot be employed for solving this problem.

Applications of global optimization methods like Metropolis algorithm and simulated annealing

to model parameter estimation have been reported recently [14, 15, 16, 17, 18, 19]. In part I of the article, published in the previous issue of this publication [20], simulated annealing approach to model parameter estimation was discussed. However, simulated annealing algorithm shows certain dependence on initial values in practice, which can be reduced but not completely eliminated by careful choice of cooling schedule and move generation procedure. The main advantage in applying GAs to model parameter estimation is the fact that initial estimates are not required. However, problem of achieving satisfactory precision was encountered - to calculate accurately optical properties with LD model, at least two significant digits in estimated model parameter values are required, and for some of the parameters only *a priori* available information is that they can have a value between 10^{-4} and 10. Independency on initial values together with arbitrary precision in locating the global optimum can be achieved with combinations of simulated annealing and GAs, but CPU time requirements of such algorithms are immense [21]. Another way of achieving improvement in the precision of locating the minimum is by introducing adaptive parameter space size in mutation process, which is incorporated in the EGAAM algorithm described here. In this work it is shown that EGAAM algorithm obtains parameter values of optical constants model that give theoretical data exhibiting good agreement with experimental values on the example of crystalline and amorphous silicon dioxide.

In the following section investigated algorithms, GA and EGAAM, are described. Section III is devoted to the comparison of performance of these algorithms applied on nine multiminima test functions. In section IV a short description of the applied model for the optical constants is given and EGAAM was used to estimate its parameters for crystalline and ampurphous SiO_2.

II Description of the algorithm

In implementation of GA must be defined representation of chromosomes, generation of initial population and genetic operators: selection, reproduction and mutation. In the following are presented the chromosome representation and genetic operators employed in investigated algorithms.

II.1 Representation of chromosomes and population generation

Continuous variables can be handled either directly, through real-valued (floating-point) representation and appropriate genetic operators, or by standard binary representation schemes and standard genetic operators. In case of binary representation, real values are approximated to the necessary degree with a fixed-point binary scheme, or the logarithm of the variable is encoded, thus reducing the required number of bits. However, since floating point representation [5, 9] proved to be more convenient for continuous optimization problems, it was applied in this paper. In floating-point chromosome representation, each gene has the value of the corresponding variable $p(k), k = 1, n_{\mathrm{v}}$, where n_{v} is the number of variables. Values $p(k)$ in chromosomes of the initial population are generated according to the formula

$$p(k) = p_{\mathrm{l}}(k) + (p_{\mathrm{u}}(k) - p_{\mathrm{l}}(k)) \cdot r, \tag{1}$$

where r is a random number $r \in [0,1]$, and $p_{\mathrm{l}}(k)$ and $p_{\mathrm{u}}(k)$ are initially set boundaries. In such a manner, confinement of variables in the specified domain is achieved insuring that all variables have physical interpretation, so that, for instance, in case of model parameter determination we can not get a negative value for frequency.

II.2 Selection and reproduction

Many different existing selection methods can be divided in two categories: random selection methods and selection methods based on the fitness measure. The former methods, like roulette wheel [6] and similar methods [22] introduce stochastic errors. These errors are reduced in the latter

ones by taking into account the fitness of an individual, like in binary tournament method [23], "stochastic remainder sampling without replacement" [24], or any other selection method incorporating following concept - the more fitted chromosome, the higher chance of being selected as a parent [3, 4, 7, 25, 26, 27]. For reducing the bias of selection method to highly fit individuals and thus preventing the premature convergence, fitness scaling can be employed [24, 27]. In this paper, elitist selection mechanism [25, 26, 28] is employed. In elitist selection, P_{s} percent of the new generation is produced by selection, and P_{c} percent is produced by crossover. $N_{\mathrm{s}} = N * P_{\mathrm{s}}$ strings with the best fitness, where N is the number of strings in the population which enter directly the next generation. The $N_{\mathrm{c}} = N * P_{\mathrm{c}}$ strings in the new population are generated by crossover among the parent strings which were chosen fitness proportionally between all the strings in the current population. The probability that a string will have offspring in the next generation is inversely proportional to

$$F(i) = \frac{f(i)}{\sum_{i=1}^{N} f(i)} \tag{2}$$

where $f(i)$ is the fitness value of the i-th string.

II.3 Crossover

Operation of crossover exchanges subsets of elements between two parent chromosomes. If the subset consists of adjacent elements, it is an "ordered combination" crossover, while in "uniform combination" crossover each element is randomly chosen [22]. Ordered combination crossover can be one-point or two-point (points of crossover are randomly selected, and elements between them are swapped). In algorithms investigated here, uniform combination crossover is employed, providing that parameter values are not altered or destroyed during the crossover, while new values are introduced in the process of mutation. Crossover is performed by generating a random integer $N_1 \in [n_{\min}, n_{\mathrm{v}}]$, where n_{v} is a number of variables, *i.e.* number of elements in strings, and $n_{\min}$ is the minimal number of elements exchanged in the crossover. Best results were obtained for $n_{\min} = n_{\mathrm{v}}/2$. When number of elements to be exchanged is determined, random integers $n_i \in [1, n_{\mathrm{par}}]$, $i = 1, N_1$ are generated and elements at positions n_i are swapped.

II.4 Mutation, concept of adaptive mutations

Mutation is necessary for maintaining certain diversity in the population, thus preventing the quick convergence to a local minimum. Mutation is usually performed by randomly altering individual genes with probability P_{m}. Real coded GAs usually perturb the solution a little around current value, which can be done with uniformly or normally distributed step, or with specifically designed manner of choosing random step [4], while binary GAs usually adopt bitwise complement operator. The mutation step can be constant, or vary with number of generations [29] or with number of successful mutations [30]. If the mutation probability is too large, mutation no longer improves performance of the population, because it enables losses of genetic information which could cause poor convergence [31]. Although P_{m} is much smaller than P_{c}, and ergo mutation is usually considered a background operator [1, 9] that ensures that probability of searching any particular subspace is never equal zero, replacing random mutations with adaptive mutations substantially improves performance of the algorithm, due to introducing the new values generated in the region containing the global minimum.

The introducing of new values is performed as follows. In the current generation, average value $\hat{\mu}(k)$ of parameter $p(k)$ is computed, and P_{m} percent of the chromosomes in the next generation are formed by generating their genes in the same manner as during the creation of initial population, but in the narrowed boundaries. New boundaries for each parameter are determined according to

$$p_{\mathrm{new-u}}(k) = p_{\mathrm{old-u}}(k) - c \cdot (p_{\mathrm{old-u}}(k) - \hat{\mu}(k)) \tag{3}$$

$$p_{\text{new-l}}(k) = p_{\text{old-l}}(k) + c \cdot (\hat{\mu}(k) - p_{\text{old-l}}(k)) \tag{4}$$

where $\hat{\mu}(k)$ is the average value of the parameter $p(k)$ in the current population, and c is a predetermined positive number $0 < c < 1$. In such a manner, a specified number ($N_{\text{m}} = N * P_{\text{m}}$) of new chromosomes is introduced in every generation. During the evolution, while $\hat{\mu}(k)$ changes towards the optimal value, parameter values in new chromosomes are more and more concentrated around $\hat{\mu}(k)$, providing finer structure and more significant digits in obtained variable values. For preventing excessive narrowing of the boundaries, their extreme values are set to $p_{\text{max-l}}(k) = \hat{\mu}(k) \cdot (1 - d)$ and $p_{\text{min-u}}(k) = \hat{\mu}(k) \cdot (1 + d)$, where d is real number having a value between 0 and 1. Concept of such adaptive mutations is incorporated in EGAAM. To investigate clearly the influence of adaptive mutations to the performance of the algorithm, both EGAAM and classical GA have the same selection and crossover mechanism, they differ only in the mutations procedure. In classical GA investigated here, uniform distributed mutation is performed by changing the parameter value with probability P_{m}. New parameter value is given by

$$p_{mut}(k) = p(k) + sgn * \Delta p(k) \tag{5}$$

where $p_{mut}(k)$, $p(k)$ are values of parameter k after and before mutations, respectively, and sgn is a random number in interval [-1,1], while $\Delta p(k)$ is the step value for parameter k.

III Test of the GA and EGAAM

To compare performance of EGAAM and GA, experiments on two families of test functions for 20, 50 and 100 variables were performed. It is well known that performance of GAs depends strongly on the parameters of the algorithm: N, P_{c}, P_{s} and P_{m}, and in the case of EGAAM, two additional parameters c and d. In results presented here influence of the number of chromosomes in the population N was investigated, since it is only one of the above parameters that significantly influences the required computational time. Before that, test runs on all test functions were performed to determine the optimal values for other algorithm parameters. The best results for classical GA were obtained for $P_{\text{c}} = 0.2$, P_{s}=0.8 and P_{m}=0.01, while optimal values for EGAAM were $P_{\text{c}} = 0.35$, P_{s}=0.6 and P_{m}=0.05. Parameter d was set to value 0.05, while $c = cc/n_{\max}$, where $n_{\max}$ is the maximal number of generations. Results given here correspond to $n_{\max} = 200$ and $cc = 3$.

The first family of multiminima functions was employed by Aluffi-Pentini *et al.* [32] and Dekkers and Aarts [33]. It is given by

$$g(\mathbf{x}) = \frac{\pi}{n}\left[k_1 \sin^2 \pi y_1 + \sum_{i=1}^{n-1}(y_i - k_2)^2(1 + k_1 \sin^2 \pi y_{i+1}) + (y_n - k_2)^2\right], \tag{6}$$

where $y_i = 1 + 0.25(x_i + 1)$, $k_1 = 10$, $k_2 = 1$. This function, shown for two variables on Fig 2, has roughly 5^n local minima for $x_i \in [-10, 10], i = 1, n$. In order to determine influence of the number of chromosomes in the population to the final objective function values, calculations for 2000, 1000, and 250 chromosomes were carried out. Obtained results for 20, 50 and 100 variables are presented in Fig. 3. It can be observed that EGAAM in all cases obtains values lower for several orders of magnitude, compared to results of GA, and its performance considerably less depends on the number of chromosomes in the population. In other words, EGAAM gives satisfactory results even for 250 chromosomes in the population when GA fails to locate the minimum.

The second family of multiminima functions, also investigated by Aluffi-Pentini *et al.* [32] and Dekkers and Aarts [33], is given by

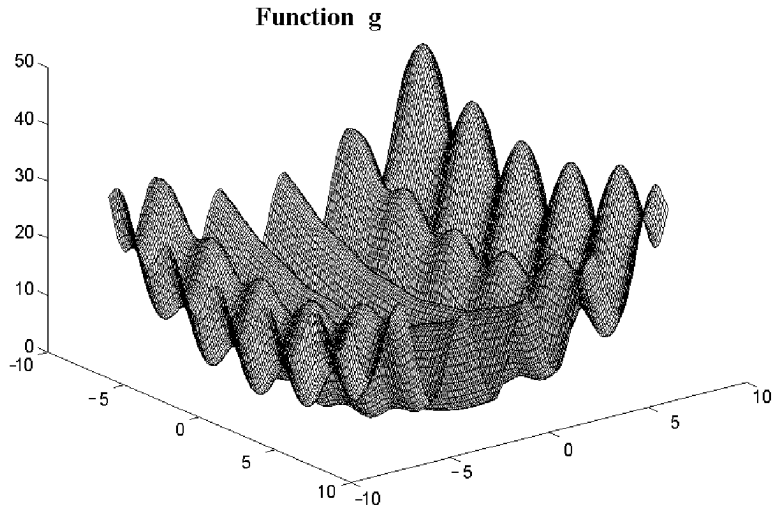

FIG. 2. Function g for two variables

$$h(\mathbf{x}) = k_3 \left\{ \sin^2(\pi k_4 x_1) + \sum_{i=1}^{n-1} (x_i - k_5)^2 [1 + k_6 \sin^2(\pi k_4 x_{i+1})] + (x_n - k_5)^2 [1 + k_6 \sin^2(\pi k_7 x_n)] \right\} \quad (7)$$

where $k_3 = 0.1, k_4 = 3, k_5 = 1, k_6 = 1$and $k_7 = 2$. In [33] this function, whose two-dimensional segment is shown on Fig 4, was investigated for five variables $x_i \in [-5, 5], i = 1..5$, *i.e.* in the area where there are roughly 15^5 minima. In this paper, family of test functions $h(\mathbf{x})$ was investigated for 20, 50 and 100 variables $x_i \in [-10, 10], i = 1, n$. Obtained results are shown in Fig. 5for $N = 2000$ chromosomes in the population only, for the sake of comprehensibility of presentation since the dependence of final solution on the number of chromosomes in the population is similar for all investigated functions. For this function EGAAM also achieves lower objective function values, for about three orders of magnitude.

IV Application to silicon dioxide

In this section, the modified Lorentz model is described. Difficulties of LOM in modeling accurately optical properties in the vicinity of strong absorption lines have already been recognized [34, 17], since maximal changes in refractive index described by LOM are comparable to the maximal value of extinction coefficient. One principal reason for the inaccuracy of the LOM is that Lorentzian shape of the spectral line is characterized with wide wings, leading to higher absorption and higher values of imaginary part of dielectric function ϵ_2. Therefore, it is often the case that there is a good agreement with experiment for real part of dielectric function ϵ_1 (or imaginary part ϵ_2), while there are significant discrepancies for the other part ϵ_2 (ϵ_1). Naturally, this causes existence of disagreement with experiment in all other optical constants, like refractive index, reflectance etc. This feature can have two causes: bad quality of experimental data, or inadequacy of the used model. For LOM, casuality, linearity, reality and Kramers-Kronig requirements are automatically satisfied, so that the agreement with experiment should be equally good for real and imaginary part of dielectric function, and the refractive index values calculated from dielectric function given by LOM should also agree well with the experimental data. Modification of the LOM employed here introduces one additional parameter which enables varying the shape of the line, and, therefore, greater flexibility of the model.

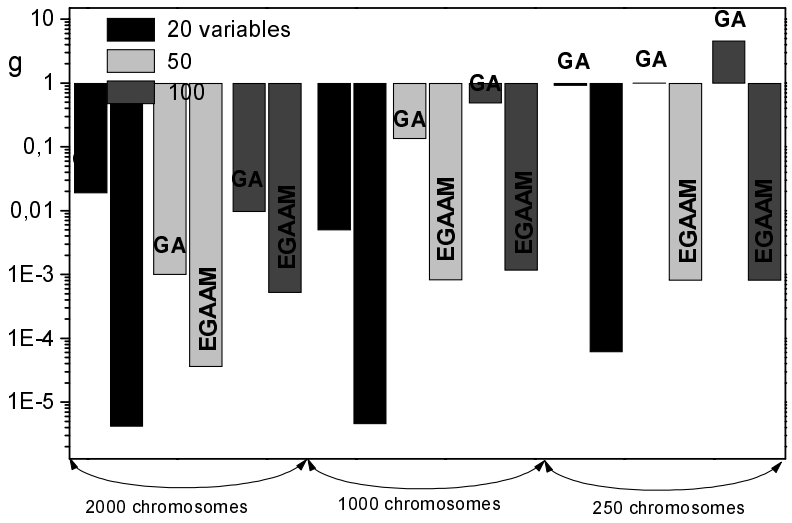

FIG. 3. Comparison of the algorithms for the function g of $x_i \in [-10, 10], i = 1, 20; 50; 100$, for 2000,1000, and 250 strings in the population.

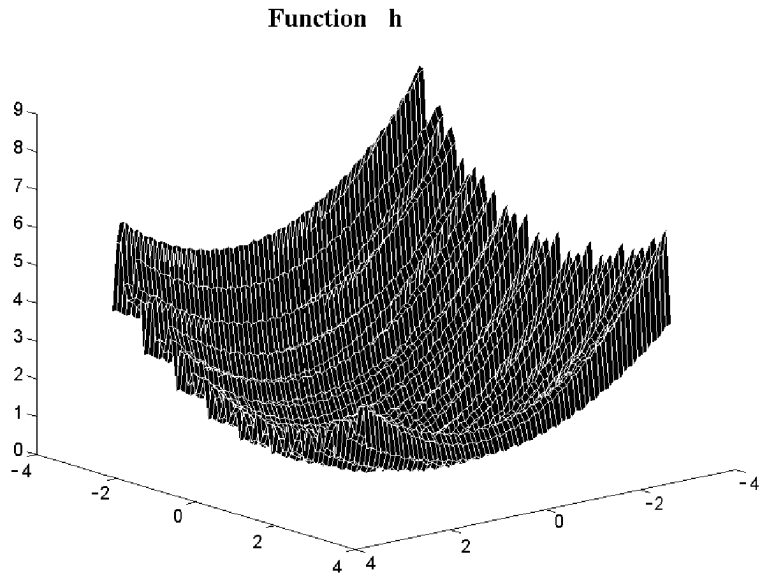

FIG. 4. Function h for two variables

The real and imaginary part of complex dielectric function $\epsilon(\omega) = \epsilon_1(\omega) + \mathbf{i}\epsilon_2(\omega)$ are expressed in the following form

$$\epsilon_1(\omega) = \epsilon_\infty + \sum_{j=1}^{k} \frac{F_j(\omega_j^2 - \omega^2)}{(\omega^2 - \omega_j^2)^2 + (\omega, {}_j)^2}, \tag{8}$$

$$\epsilon_2(\omega) = \sum_{j=1}^{k} \frac{F_j \omega, {}_j}{(\omega^2 - \omega_j^2)^2 + (\omega, {}_j)^2}, \tag{9}$$

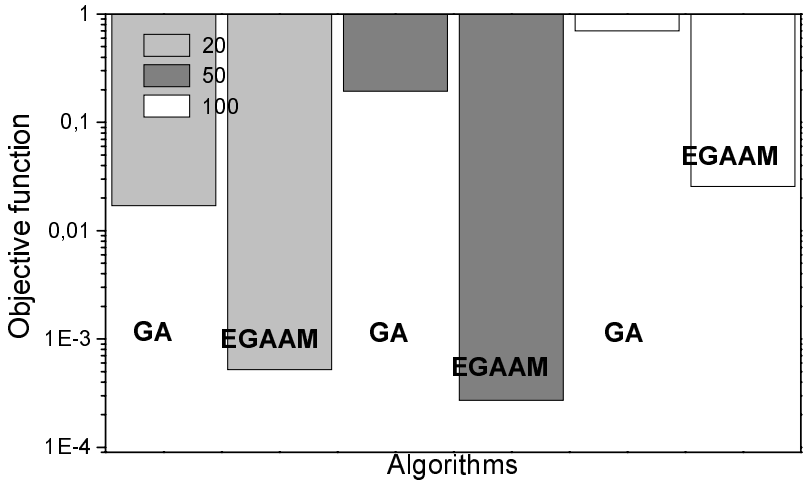

FIG. 5. Comparison of the algorithms for the function h of $x_i \in [-10, 10], i = 1, n; n = 20, 50$ *and 100, for 2000 strings in the population.*

where k is the number of interband transitions with frequency ω_j and lifetime $1/, _j$, while $F_j = f_j\omega_j^2$ is parameter associated with oscillator strength f_j.

When the dielectric function is determined, refractive index is calculated from the expression

$$n = \sqrt{1/2\left(\epsilon_1 + \sqrt{\epsilon_1^2 + \epsilon_2^2}\right)}. \tag{10}$$

In the work of Kim *et al* [35] it was shown that the shape of dielectric function that closely mimics one with assumed Gaussian broadening (which can not be calculated analytically) can be obtained for suitable values of parameter α if the damping constants are replaced with frequency dependent expression

$$, _j' = , _j \exp\left(-\alpha_j \left(\frac{\hbar\omega - E_j}{, _j}\right)^2\right) \tag{11}$$

Recently, Rakić *et al* [18] have shown that better agreement with experimental data for GaAs/AlAs can be obtained by including the above expression in Adachi's model of optical properties of semiconductors [36]. In this work the same frequency dependant damping is introduced in LOM model. In such a manner, the line shape can vary from purely Lorentzian for low values of $\alpha/, $, for $\alpha/, =0.3$ the line shape is nearly Gaussian, while with the further increase of $\alpha/, $ the wings of the peak in ϵ_2 can be reduced even further. Also, greater values of $\alpha/, $ give more pronounced structure in $\epsilon_1$and refractive index n.

Model parameters were determined by minimizing the following objective function

$$E(p) = \sum_{i=1}^{i=N} \left(\frac{n(\omega_i)}{n^{\exp}(\omega_i)} - 1\right)^2. \tag{12}$$

where the summation is performed over available experimental points at frequency ω_i, while $n^{\exp}(\omega_i)$, $n(\omega_i)$ are the experimental and calculated values of refractive index at point ω_i, respectively.

There have been numerous studies of the room temperature optical properties of amorphous SiO_2 (glass). The data used in this paper, ranging from 0.15 eV to 25 eV, are from the study of Philipp

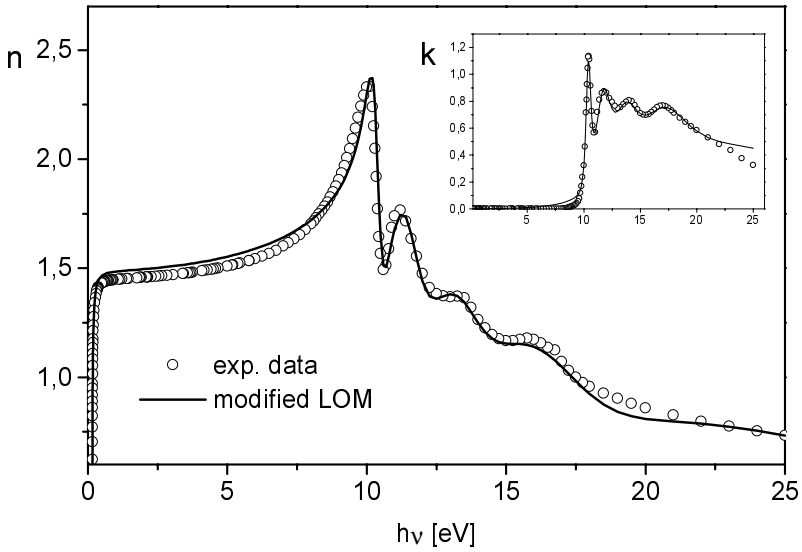

FIG. 6. Refractive index of amorphous SiO_2 vs. *energy; circles - exp. data, solid line - modified LOM. Inset shows imaginary part of the index of refraction $k(\omega)$ as a function of energy.*

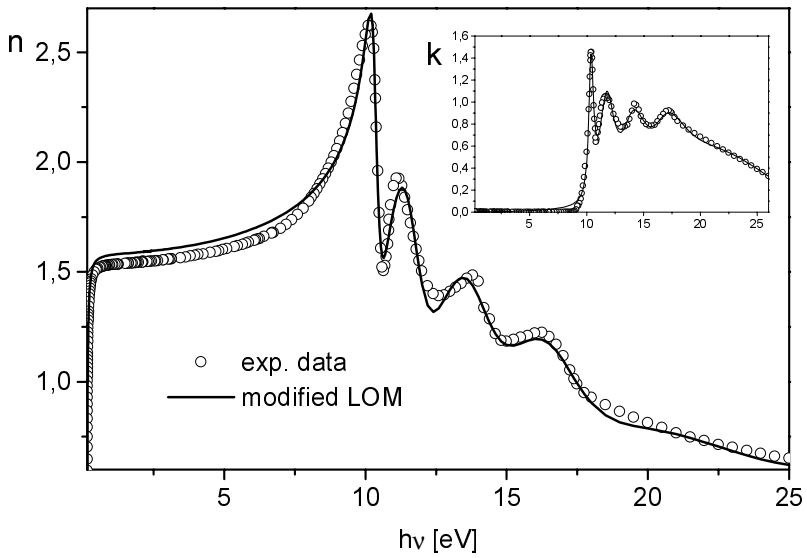

FIG. 7. Refractive index of crystalline SiO_2 vs. *energy; circles - exp. data, solid line - modified LOM. Inset shows imaginary part of the index of refraction $k(\omega)$ as a function of energy.*

[37], who collected data from several sources. Fig. 6 shows refractive index of amorphous SiO_2 (glass) *vs.* energy. Open circles represent experimental data, solid line best fit to modified LOM. Inset shows imaginary part of the index of refraction $k(\omega)$ as a function of energy. In modeling the index of refraction of crystalline silicon dioxide, the data tabulated by Philipp in *Handbook of Optical Constants of Solids* [38] are employed, in the range 0.15 eV to 25 eV. Obtained results are shown on Fig. 7 depicting real part of the index of refraction $n(\omega)$ as a function of energy, while inset shows imaginary part of the index of refraction $k(\omega)$ as a function of energy. For both materials, excellent agreement with the experimental data is evident.

V Conclusion

Genetic algorithms as a tool for solving continuous optimization problems are described. Performance of the elite genetic algorithm with adaptive mutations, which was developed for solving model parameter estimation problem, is compared to the performance of conventional genetic algorithm on two families of multiminima test functions for 20, 50 and 100 variables. After verifying the superiority of this algorithm over its classical counterpart, parameters of the modified Lorentz oscillator model have been determined for amorphous and crystalline silicon dioxide. The employed model differs from the conventional Lorentz oscillator model in terms of the assumed broadening type, i.e. each oscillator is characterized with four instead of three parameters, in order to enable varying of the broadening function from Lorentzian over Gaussian to even narrower lineshapes. Excellent agreement with the experimental data is obtained for both investigated materials.

References

[1] D. E. Goldberg, *Genetic algorithms in search, optimization, and machine learning*, (Addison-Wesley, Reading, 1989).

[2] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, Science **220**, 671 (1983).

[3] M. W. Gutowski, J. Phys. A, Math. Gen., **27**, 7893, (1994).

[4] H. Mühlenbein and D. Schlierkamp-Voosen, Evolutionary Computation, **1**, 25, (1993).

[5] K. P. Wong and Y. W. Wong, in *Proc. ANZIIS-93, Perth, Western Australia*, pp. 512-516, 1993.

[6] K. P. Wong and Y. W. Wong, IEE Proc. Gen. Transm. Distrib., **141**, 507, 1994.

[7] A. B. Djurišić, J. M. Elazar and A. D. Rakić, Opt.Comm., **134**, 407, (1997).

[8] A. Chipperfield and R. Fleming, Control and Computers, **23**, 88, (1995).

[9] D. E. Goldberg, *Complex Systems*, **5**, 139, (1991).

[10] K. Deb and R. B. Agrawal, *Complex Systems*, **9**, 115, (1995).

[11] J. L. Eshelman and J. D. Schaffer, in *Proc. of Foundations of GA Workshop*, pp.187-202, (1992).

[12] A. D. Rakić. Appl. Opt., **34**, 4755, (1995).

[13] C. J. Powell, *J. Opt. Soc. Am.*, **60**,78, (1970).

[14] A. B. Djurišić,A. D. Rakić and J. M. Elazar, Phys. Rev. E, **55**, 4797, (1997).

[15] A. D. Rakić, J. M. Elazar, and A. B. Djurišić. Phys. Rev. E, **52**, 6862, (1995).

[16] T. H. Han and W.W.Chang, *Elec. Lett.*, **32**, 1256, (1996).

[17] A. Franke, A. Stendal, O. Stenzel and C. von Borczyskowski, *Pure Appl. Opt.*, **5**, 845, (1996).

[18] A. D. Rakić and M. L. Majewski, *J. Appl. Phys*, **80**, 5909, (1996).

[19] M. K. Vai, S. Prasad, N. C. Li and F. Kai, *IEEE Trans. on Electron. Devices*, **36**, 761, (1989).

[20] A. B. A. B. Djurišić, Publications of the Faculty of Electrical Engineering, University of Belgrade, Series: Engineering Physics, pp. 43-57, 1997.

[21] A. B. Djurišić,A. D. Rakić and J. M. Elazar, Appl. Opt, **36**, ???, (1997).

[22] F. Curatelli, Int. J. Electronics, **78**, 435, (1995).

[23] S. E. Cienawski, J. W. Ehart and S. Ranjithan, Water Resources Research, **31**, 399, (1995).

[24] Q. Li, E. J. Rotwell, K.-M. Chen and D. P. Nyquist, IEEE Trans. on Antennas and Propagation, **44**, 198, (1996).

[25] R. Vemuri and R. Vemuri, Elec. Lett., **30**, 1270, (1994).

[26] S. H. Clearwater and T. Hogg, Artificial intelligence, **81**, 327, (1996).

[27] C. B. Lucasius, M. L. M. Beckers and G. Kateman, Analytica Chimica Acta, **286**, 135, (1994).

[28] R. R. Brooks, S. S. Iyengar and J. Chen, Artificial intelligence, **81**, 327, (1996).

[29] U. D. Hanebeck and G. K.Schmidt, *Fuzzy Sets and Systems*, **79**, 59, 1996.

[30] T. Bäck and H. P. Schwefell, in *Genetic algorithms in engineering and computer science*, John Wiley & Sons Ltd., Chichester, pp.111-140, (1995).

[31] D. Raynolds and J. Gonatann, Artificial intelligence, **82**, 303, (1996).

[32] F. Aluffi-Pentini, V. Parisi and F. Zirilli, Journal of optimization theory and applications, **47**, 1, (1985).

[33] A. Dekkers and E. Aarts, Mathematical programming, **50**, 367, (1991).

[34] O. Stenzel, R. Petrich, W. Scharff, A. Tikhonravov and V. Hopfe, *Thin Solid Films* **207**, 324, (1992).

[35] C. C. Kim, J. W. Garland, H. Abad and P. M. Raccah, *Phys. Rev. B* **45**, 11749, (1992).

[36] S. Ozaki and S. Adachi, *J. Appl Phys.* **78**, 3380, (1995).

[37] H. R. Philipp, in *Handbook of Optical Constants of Solids*, edited by E. D. Palik (Academic Press Inc., Orlando, Fl.), pp. 749-764, (1985).

[38] H. R. Philipp, in *Handbook of Optical Constants of Solids*, edited by E. D. Palik (Academic Press Inc., Orlando, Fl.), pp. 719-748 (1985).

(Received July 1998.)