GENERALIZED EXTREMAL OPTIMIZATION:
A NEW STOCHASTIC ALGORITHM FOR SOLVING COMPLEX OPTIMAL DESIGN PROBLEMS

Fabiano Luis de Sousa
Instituto Nacional de Pesquisas Espaciais – INPE/DMC
Av. dos Astronautas, 1758, São José dos Campos, SP 12227-010, Brazil
fabiano@dem.inpe.br
Valeri Vlassov
Instituto Nacional de Pesquisas Espaciais – INPE/DMC
Av. dos Astronautas, 1758, São José dos Campos, SP 12227-010, Brazil
vlassov@dem.inpe.br
Fernando Manuel Ramos
Instituto Nacional de Pesquisas Espaciais – INPE/DMC
Av. dos Astronautas, 1758, São José dos Campos, SP 12227-010, Brazil
fernando@lac.inpe.br

Abstract. *In this paper the Generalized Extremal Optimization algorithm is presented. Recently proposed as a tool to be applied on complex optimal design problems, it is of easy implementation, does not make use of derivatives and can be applied to unconstrained or constrained problems, non-convex or disjoint design spaces, with any combination of continuous, discrete or integer variables. It is a global search meta-heuristic, as the Genetic Algorithm and the Simulated Annealing, but with the a priori advantage of having only one free parameter to adjust. Here, the efficacy of the Generalized Extremal Optimization method to deal with a complex design space, with non-linear equality and/or inequality constraints, is illustrated by a numerical example with a test function and the optimal design of a heat pipe for a space application.*

Keywords: *Optimal design, Optimization, Stochastic algorithm, heat pipe.*

1. INTRODUCTION

Recently, Boettcher and Percus (2001) proposed a new optimization algorithm that, as the Genetic Algorithm (GA), is based on the principles of natural selection, but that does not uses the GA's framework of population reproduction. The backbone of their algorithm is built over a simplified model of natural selection, developed to show the emergence of Self-Organized Criticality (SOC) in ecosystems . Evolution in this model is driven by a process where the weakest species in the population, together with its nearest neighbors, is always forced to mutate. The dynamics of this extremal process showed characteristics of SOC, such as punctuated equilibrium, that is also observed in natural ecosystems (Bak & Sneppen, 1993).

Boettcher and Percus (2001) have adapted the evolutionary model of Bak and Sneppel (1993) to tackle hard problems in combinatorial optimization, calling their algorithm Extremal Optimization (EO). To improve performance and avoid a search based only on the forced mutation of the weakest species, they modified the basic EO algorithm introducing an adjustable parameter so that the search could escape local optima. This variation of the EO algorithm was called τ-EO and showed superior performance over the EO even in the cases where the later worked well. However, a drawback of τ-EO (and also EO) is that for each new optimization problem assessed, a new way to define the fitness of the design variables has to be created (Boettcher & Percus, 2001). Moreover, to our knowledge it has been applied so far to combinatorial problems with no implementation to continuous functions.

In order to make the EO method easily applicable to a broad class of design optimization problems, Sousa and Ramos (2002) have proposed a generalization of the EO method called Generalized Extremal Optimization (GEO). As the Simulated Annealing (SA) and the GA, it is a stochastic method, does not make use of derivatives and can be applied to nonconvex or disjoint problems. It can also deal with any kind of variable, either continuous, discrete or integer. In fact, The GEO method has proved to be competitive to the SA and the GA (Sousa & Ramos, 2002; Sousa et al).

In this paper, the efficacy of the GEO method to deal with a constrained design space, which may contain non-linear equally and inequality constraints, is illustrated by a numerical example and a real design application. In the numerical example, the performance of the GEO is compared to a standard GA and the Progressive Genetic Algorithm (PGA) (Guan & Aral, 1999) in finding the solution of a constrained test function. In the optimal design problem, the method is applied to the optimization of a heat pipe (HP) for a space application. This problem pose difficulties to the GEO such as an objective function that presents design variables with strong non-linear interactions, subject to multiple constraints, being considered unsuitable to be solved by traditional gradient based optimization methods (Rajesh & Ravindran, 1997). The HP is optimized in regard to its total mass, given a desirable heat transfer rate and boundary conditions on the condenser. A total of 18 constraints are taken into account, which include operational, dimensional and structural ones. Temperature dependent fluid properties are considered and the calculations are done for steady state conditions. Several runs were performed under different values of heat transfer rate and temperature at the condenser. Integral optimal characteristics were obtained.

2. THE GENERALIZED EXTREMAL OPTIMIZATION ALGORITHM

Self-organized criticality (SOC) has been used to explain the behavior of complex systems in such different areas as geology, economy and biology (Bak, 1996). The theory of SOC states that large interactive systems evolves naturally to a critical state where a single change in one of its elements generates "avalanches" that can reach any number of elements on the system. The probability distribution of the sizes "s" of these avalanches is described by a power law in the form $P(s) \sim s^{-\tau}$, where τ is a positive parameter. That is, smaller avalanches are more likely to occur than big ones, but even avalanches as big as the whole system may occur with a non-negligible probability. To show that SOC could explain features of systems like the natural evolution, Bak and Sneepen (1993) developed a simplified model of an ecosystem in which species are placed side by side on a line with periodic boundary conditions. To each species, a fitness number is assigned randomly, with uniform distribution, in the range [0,1]. The least adapted species, the one with the least fitness, is then forced to mutate, and a new random number assigned to it. The change in the fitness of the least adapted species alters the fitness landscape of their neighbors and, to cope with that, new random numbers are also assigned to them, even if they are well adapted. After some iterations, the system evolves to a critical state where all species have fitness above a critical threshold. However, the dynamics of the system eventually causes a number of species to fall below the critical threshold in avalanches that can be as big as the whole system. An optimization heuristic based on the Bak-Sneppen model would evolve solutions quickly, systematically mutating the worst individuals, whereas at the same time preserving throughout the search process, the possibility of probing different regions of the design space (via avalanches). This would enable the algorithm to escape local optima.

Inspired by the Bak-Sneppen model, Boettcher and Percus (2001) devised the EO method, where each variable is considered as one species and to each of them is assigned a fitness number that represents the level of adaptability of that species. In the τ-EO implementation of the method, the ecosystem of N variables "evolves" (that is, search the design space for the global optimum), as one variable is mutated with probability $P_k \propto k^{-\tau}$. Where, k is the rank of the variable. The rank varies from 1, to the variable with least fitness, to n, for the variable with the best fitness. Hence, while privileging the variable with least rank to mutate, the algorithm makes all variables accessible to mutate, making it able to "jump" out of regions of local minimum. In a practical implementation of the τ-EO method, making $\tau \to 0$ the algorithm becomes a random walk, while for $\tau \to \infty$, we have a deterministic search.

Although already been successfully applied to hard combinatorial problems, as pointed out by Boettcher and Percus (2001), referring collectively to both the basic EO and the τ-EO, "a drawback to EO is that a general definition of fitness for the individual variables may prove ambiguous or even impossible" (Boettcher & Percus, 2001). What means that for each new optimization problem assessed, a new way to define the fitness of the design variables has to be created. Moreover, to our knowledge it has been applied so far to combinatorial problems with no implementation to continuous variables. In order to make the EO method applicable to a broad class of design optimization problems, without concern to how the fitness of the design variables are assigned, and capable of tackle continuous, discrete or integer variables, a generalization of the EO method was devised (Sousa and Ramos, 2002). In this new algorithm, named Generalized Extremal Optimization, the fitness assignment is not done directly to the design variables, but to a "population of species" that encodes the variables. Each species receives its fitness, and eventually mutates, following general rules.

In the GEO algorithm each species is represented by a bit in a string, that represents the entire ecosystem of species. The variables are encoded in this string that is similar to a chromosome in the canonical GA. But different from the GA, in the GEO there is not a population of strings (or solutions), but a population of bits represented by one string (see Figure 1). To each bit of this string is assigned a fitness number that indicates the level of adaptability of that bit on the population, according to the gain or loss that one has on the value of the objective function, when the bit is mutated (flipped).

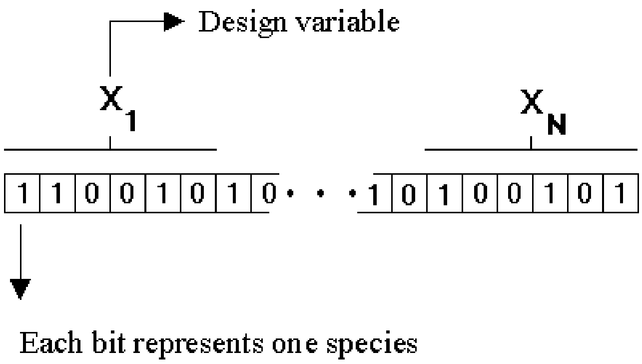

Figure 1 - The population of bits. In this example each design variable is encoded in 6 bits.

The GEO algorithm is written as:

1. Initialize randomly a binary string of length L that encodes N design variables of bit length l_j (j = 1, N). For the initial configuration C of bits, calculate the objective function value V and set C_{best} = C and V_{best} = V.
2. For each bit i of the string, at a given iteration:
   a) flip the bit (from 0 to 1 or 1 to 0) and calculate the objective function value V_i of the string configuration C_i,
   b) set the bit fitness as $\Delta V_i = (V_i - V_{best})$. It indicates the relative gain (or loss) that one has in mutating the bit, compared to the best objective function value found so far.
   c) return the bit to its original value.
3) Rank the bits according to their fitness values, from k = 1 for the least adapted bit to k = L for the best adapted. In a minimization problem, higher values of ΔV_i will have higher ranking, and otherwise for maximization problems. If two or more bits have the same fitness, rank them randomly with uniform distribution.
4) Choose with equal probability a candidate bit i to mutate. Generate a random number RAN with uniform distribution in the range [0,1]. If the mutating probability $P_i(k) = k^{-\tau}$ of the chosen bit is equal or greater than RAN the bit is confirmed to mutate. Otherwise, the process is repeated until a bit is confirmed to mutate.
5) For the mutated bit set C = C_i and V = V_i.
6) If $V < V_{best}$ ($V > V_{best}$, for a maximization problem) then set V_{best} = V and C_{best} = C.
7) Repeat steps 2 to 6 until a given stopping criteria is reached.
8) Return C_{best} and V_{best}.

Note that in step 4 any bit can be chosen to mutate, but the probability of a given chosen bit be confirmed to mutate is dependent on its rank position. The ones more adapted (with higher rank values) are less prone to have its mutation confirmed and only the least adapted bit (rank = 1) is always confirmed to mutate, if chosen. The probability of mutating the chosen bit is regulated by the adjustable parameter τ. The higher the value of τ, the smaller the chance of a bit (with rank greater than 1) be mutated. The possibility of making moves that do not improve the value of the objective function is what allows the algorithm to escape from local optima.

Equality and inequality constraints are taken into account simply setting a high (for a minimization problem) or low (for a maximization problem) fitness value to the bit that, when flipped, leads the configuration to an unfeasible region of the design space. Side constraints are directly applied through the encoding of the design variables. Note that the move to an infeasible region is not prohibited, since any bit has a chance to mutate according to the P(k) distribution. Moreover, no special condition is posed for the beginning of the search process, which can even start from an infeasible region. It must be pointed out here, that other ways to take into account constraints in GEO may also be implemented, such as a penalty function approach (Barbosa & Lemonge, 2002). However, the approach described above is very simple to apply and does not introduce any new adjustable parameter in the algorithm.

A slightly different implementation of the GEO algorithm can be obtained, changing the way the bits are ranked and mutated. Instead of ranking all the bits according to step 3, we rank them separately for each variable. In this way the bits of each variable will have a rank ranging from 1 to l_j. In step 4 one bit of each variable is chosen and mutated in the same way of GEO, but the process is done independently for each variable. Hence, at each iteration, N bits are mutated in this new implementation of GEO, one per variable, while in its canonical form only one bit is mutated per iteration. We named this variation of the GEO algorithm

GEO_{var}. The idea behind GEO_{var} is to try to improve all variables simultaneously, at each algorithm iteration, as an attempt to speed up the process of searching the global minimum.

3. TEST FUNCTION EXAMPLE

In order to illustrate the performance of the GEO algorithm in a constrained optimization problem, we compared it to a standard GA and the PGA, in a test function subject to non-linear equality and inequality constraints. The optimization problem is stated as follows:

Minimize

$$F(X) = x_1^3 + 2x_2^3 x_3 + 2x_3$$

Subject to

$$x_1^2 + x_2 + x_3^2 = 4\,;\; x_1^2 - x_2 + 2x_3 \leq 2\,;\; 0 \leq x_2 \leq 4;\; 0 \leq x_3 \leq 2;\; x_1 \geq 0\,.$$

This test function has a global minimum at X = {0,4,0}, where the objective function value is zero. It was used by Guan and Aral (1999) to assess the performance of the PGA, which was specially developed to tackle problems with non-linear equality and inequality constraints. In the PGA the search process is divided in two phases. First, the non-linear constraints are linearized and then, the simplified problem is solved by a conventional GA. The GA works on sub-domains of the design variables. The idea behind the PGA approach is to transform a complex optimization problem with nonlinear constraints in a series of simplified problems that are solved by a GA. In the PGA the first derivatives of the constraints have to be calculated, while the derivatives of the objective function may also be used to increase the performance of the algorithm, if available.

Guan and Aral (1999) compared the performance of the PGA to find the optimum of the test function with the one of a standard GA. The standard GA used a population of 50 individuals and the search for the optimum stopped after 50 generations, what resulted in a value of 2500 for the number of function evaluations (*nfe*) at the end of each search performed with the standard GA. On the other hand, no information could be retrieved from the paper of Guan and Aral (1999) that could lead to an estimation of *nfe* for the PGA runs. Hence, for comparison purposes, we used as the stopping criteria for the GEOs the same one used for the standard GA, that is, the GEOs stopped when $nfe \geq 2500$. Since the performance of the GEOs are influenced by the value of τ, a search in the interval [0.25, 3.00] was performed to find out the value of τ that provided the best results in the search for the optimum of the objective function. For each GEO, at a given value of τ, were performed 50 independent runs. As can be seen on Figure 2, better results were obtained for GEO_{var} at τ = 1.75, where the exact solution was found seven times out of 50 runs, with an average value for the minimum value of F(X) of 0.0053. For the GEO, the exact solution was found in all runs performed for τ in the range [1.00, 3.00].

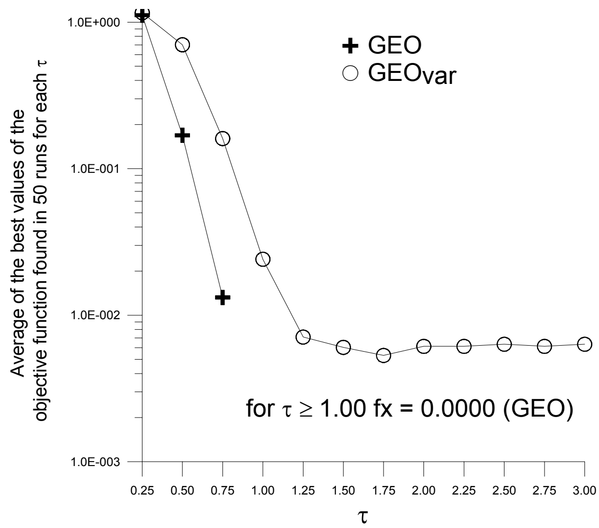

Figure 2 – Average of the best values of the objective function as a function of τ.

In Table 1, the results obtained by Guam and Aral for the standard GA and the PGA are shown.

Table 1. Guam and Aral results for the test function[a].

Variable	Exact solution	Standard GA with penalty function[a]		PGA[a] $\gamma = 0.1$	
		$\beta = 10$	$\beta = 100$	Search start at {2,0,0}	Search start at {1.4,1,1}
F(X)	0.0000	0.18035	0.00264	0.0000	0.0000

[a]From Guan and Aral (1999) ; β = penalty coefficient ; γ = interval contraction coefficient.

From the results of Figure 2 and Table 1, it can be seen that the GEO algorithm performed clearly better than the GEO_{var} and the standard GA in the range $\tau \in [1.0,3.0]$. A direct comparison between the performance of the GEO algorithms and the PGA in terms of *nfe* was not possible since no information was given concerning the *nfe* necessary for the later to converge. Nevertheless, it is mentioned by Guam and Aral (1999) that the GA seemed to have required fewer computations than the PGA to find the solutions, what indicates that the GEO, that used the same *nfe* that the GA as stopping criteria, may also have performed better than the PGA. Moreover, as opposed to the PGA, which requires the constraint functions to be first-order continuously differentiable, both GEO and GEO_{var} algorithms can tackle the constraints regardless of their functional characteristics.

4. OPTIMIZATION OF A HEAT PIPE FOR A SPACE APPLICATION

Heat Pipes (HPs) are thermal devices used to transfer high amounts of heat over long distances with a minimum temperature gradient. Although their concept had been devised in the late 1940's, it was from the mid-1960's that they started to be developed and used in engineering applications, that now range from oil ducts to spacecraft (Peterson, 1994). In its basic form, the HP is a hermetically sealed tube-type container with a porous structure placed on its internal walls and filled with a working fluid. Vapor occupies the center of the tube (vapor core) whereas liquid fills the porous structure (wick). When operating, liquid at the evaporator side of the HP evaporates, and vapor moves through the center of the tube to the condenser, where it condenses. At the same time, liquid flows through the wick from the condenser to the evaporator due to the action of capillary forces. This heat-mass transfer mechanism can transport great amounts of heat, from the evaporator to the condenser with little temperature drop between the two parts. In Figure 3 a drawing of the HP concept is shown.

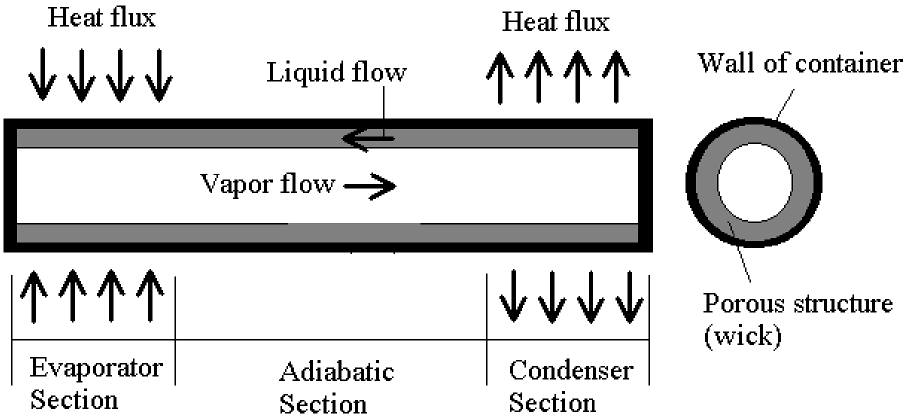

Figure 3 – Concept of operation of a conventional heat pipe.

The materials of the container's wall and wick, as well as the working fluid are chosen depending on the application for the HP and their compatibility. There exists also a variety of wick types available for usage. Details of HP operating, materials compatibility, testing and technologies can be found elsewhere (for example, in Peterson, 1994; or Faghri, 1995).

In any space application, one of the main concerns is to keep the total mass of the space platform as low as possible. Hence, the design of HPs for this kind of application focus on a desired performance while keeping its mass as low as possible. In the problem tackled here, the objective function to be minimized is the total mass of the HP (M_{total}) for a desired heat transfer rate (Q), given a constant temperature at the outside surface of the condenser section (T_{si}). The design variables are the wick's mesh number (N), the diameter of the vapor core (d_v), the diameter of the wick's wire (d), the thickness of wick (t_w), the thickness of the container's wall (t_t), the length of the evaporator section (L_e) and the length of the condenser section (L_c). The length of the adiabatic section (L_a) is dependent on the application and here was fixed equal to 0.5 m.

A total of 18 constraints are applied to the HP, which can be divided in dimensional, operating and structural. The dimensional constraints are mainly concerned with practical aspects of manufacture and installation of the HP, such as defining feasible minimum lengths for the evaporator and condenser section. Operational constraints are posed to assure that the HP will operate properly while keeping the temperature of the equipment from which heat is being removed within a specified operational range. Finally, since the HP is essentially a pressurized system, structural constraints are applied so that the burst of the container is prevented. From these 18 constraints, 7 are side constraints, applied to the design variables, and 11 are inequality constraints.

In this paper, we considered three different working fluids: ethanol, methanol and ammonia. Stainless Steel (SS 304) was used as the material of the container since it is compatible with all these fluids. The wick is of the mesh type and also made of SS 304. Results were obtained for a heat transfer rate (Q) and a condenser temperature (T_{si}) ranging from 25.0 W to 100.0 W and from $-15^{o}C$ to $30^{o}C$, respectively. The maximum temperature allowed on the external surface of the evaporator was fixed in 45.0 ^{o}C. Data from Dunn and Reay (1976) were used to obtain interpolation curves for the temperature dependent fluid properties.

The first decision to be made on the utilization of GEO and GEO_{var} is the number of bits used to each design variable. These values depend on the precision one desires for each variable. For the present problem, the design parameter that required the bigger number of bits to encode its value within the desired precision was L_c. It required 14 bits and, for the sake of simplicity and considering that the computational cost of estimating the objective function was small, that was the number also used for the other design variables.

Since the performances of the GEO and GEO_{var} are dependent on the parameter τ, we first made a study to determine their best value, for each algorithm. We set T_{si} = 0.0 °C and Q = 25.0 W and run GEO and GEO_{var} for 10^5 function evaluations. Fifty independent runs were made for each algorithm, and for various values of τ, in the range [0.25,3.00], in steps of 0.25. The initialization of the string of bits at each run was done randomly.

From these numerical tests, since the GEO algorithm performed worse than the GEO_{var}, we decided to use only the later in the remaining simulations, for all operational conditions. The values of τ used on these runs were τ = 2.75 for the ethanol, τ = 2.00 for the methanol and τ = 1.75 for the ammonia. The stopping criteria was modified so that the search stopped after the number of function evaluations reached 10^6.

In Figure 4, the minimum values found for the HP total mass are plotted, for all operational conditions.

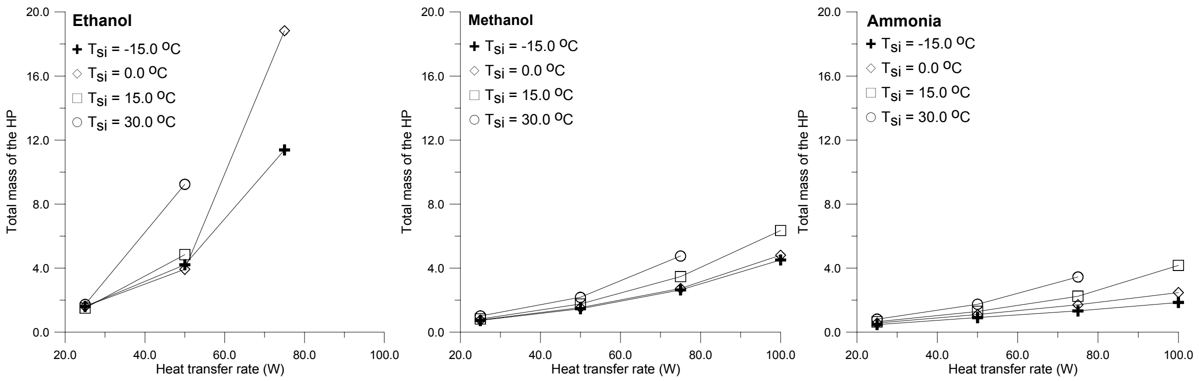

Figure 4 – Minimum HP mass found for ethanol, methanol and ammonia, at different operational conditions.

It can be seen from these results that for moderate heat flux rates (up to 50 W), the ammonia and methanol HPs display similar results in terms of optimal mass, while for high heat transport rates (as for Q = 100 W), the HP filled with ammonia shows considerably better performance. In practice this means that, for applications which require the transport of moderate heat flow rates, cheaper methanol HPs can be used, whereas at higher heat transport rates, the ammonia HP should be utilized.

We also note that the higher the heat to be transferred, the higher the HP total mass. Although this is an expected result, the apparent non-linearity of the HP mass with Q means that for some applications there is a theoretical possibility that the use of two HPs of a given heat transfer capability can yield a better performance, in terms of mass optimization, than the use of an unique HP with double capability. From the curves in Figure 3, this non-linearity is more pronounced as the temperature on the external surface of the condenser is increased.

Finally, it is noteworthy that the apparent non-linearity of the optimal characteristics has an important significance in design practice and, thus, should be further investigated.

5. CONCLUSIONS

In this paper the Generalized Extremal Optimization algorithm was presented and its efficacy to tackle complex design optimization problems illustrated in a numerical experiment with a constrained test function, and in an application to a heat pipe optimization problem. Inspired by the Bak-Sneppen model of natural evolution, it is a global search stochastic algorithm devised for handling problems that presents such features as nonconvex design domains or presence of different kinds of design variables. Having only one free parameter to adjust gives the GEO, and its variation GEO_{var}, an “a priori” advantage over other popular stochastic algorithms, such as the GA or the SA, since the time spent on fine tuning the algorithm to its best performance would be greatly reduced.

Summarizing, from the results shown on this paper and previously studies (Sousa and Ramos, 2002; Sousa et al), it can be said that the GEO method is a good candidate to be incorporated to the designer’s tools suitcase. Further work is under the way on the implementation of hybrids, parallelization of the algorithm and other practical applications.

Acknowledgements

F. M. Ramos acknowledges the support of CNPq-Brazil through the research grant 300171/97-8. V. V. Vlassov acknowledges the support of CNPq-Brazil through the research grant 300034/96-2/2002.

References

Bak, P., 1996, How Nature Works. Copernicus, Springer-Verlag.

Bak, P. and Sneppen, K., 1993, Punctuated equilibrium and criticality in a simple model of evolution, Physical Review Letters, Vol. 71, Number 24, 4083-4086.

Boettcher, S. and Percus, A. G., 2001, Optimization with extremal dynamics, Physical Review Letters, Vol. 86, 5211-5214.

Barbosa, H.J.C and Lemonge, A.C.C., 2002, An Adaptive penalty scheme in genetic algorithms for constrained optimization problems. GECCO 2002: Proceedings of the Genetic and Evolutionary Computation Conference. Morgan Kaufmann, San Francisco, 287-294.

Dunn, P. and Reay, D.A., 1976, Heat Pipes. Pergamon Press.

Faghri, A., 1995, Heat Pipe Science and Technology, Taylor & Francis.

Guan, J. and Aral, M.M., 1999, Progressive Genetic Algorithm for Solution of Optimization Problems With Nonlinear Equality and Inequality Constraints, Applied Mathematical Modelling, Vol. 23, 329-343.

Rajesh, V.G. and Ravindran K.P., 1997, Optimum heat pipe design: A nonlinear programming approach. International Communications in Heat and Mass Transfer, Vol. 24, No. 3, 371-380.

Peterson, G.P., 1994, An Introduction to Heat Pipes. Modeling, Testing, and Applications. John Wiley & Sons.

Sousa, F.L. and Ramos, F.M., 2002, Function optimization using extremal dynamics. Proceedings of the 4th International Conference on Inverse Problems in Engineering, Rio de Janeiro, Brazil.

Sousa, F.L., Ramos, F.M. Paglione P., Girardi R. M., A new stochastic algorithm for design optimization, AIAA Journal (in press).