Journal of Global Optimization **13:** 433–444, 1998.

Stochastic Methods for Practical Global Optimization

ZELDA B. ZABINSKY
Industrial Engineering, University of Washington, Seattle, Washington, USA

(Accepted in final form 30 June 1998)

Abstract. Engineering design problems often involve global optimization of functions that are supplied as 'black box' functions. These functions may be nonconvex, nondifferentiable and even discontinuous. In addition, the decision variables may be a combination of discrete and continuous variables. The functions are usually computationally expensive, and may involve finite element methods. An engineering example of this type of problem is to minimize the weight of a structure, while limiting strain to be below a certain threshold. This type of global optimization problem is very difficult to solve, yet design engineers must find some solution to their problem – even if it is a suboptimal one. Sometimes the most difficult part of the problem is finding any feasible solution. Stochastic methods, including sequential random search and simulated annealing, are finding many applications to this type of practical global optimization problem. Improving Hit-and-Run (IHR) is a sequential random search method that has been successfully used in several engineering design applications, such as the optimal design of composite structures. A motivation to IHR is discussed as well as several enhancements. The enhancements include allowing both continuous and discrete variables in the problem formulation. This has many practical advantages, because design variables often involve a mixture of continuous and discrete values. IHR and several variations have been applied to the composites design problem. Some of this practical experience is discussed.

Key words: Adaptive search, Global optimization, Multi-disciplinary optimization, Simulated annealing

1. Introduction

Global optimization problems are prevalent in engineering design. Often times the physical nature of engineering problems have multiple optima, however engineers are just beginning to make use of techniques available in global optimization, as opposed to local optimization techniques. Engineering design problems, in particular structural design, have used traditional gradient-based local optimization methods for over 30 years [31]. How can global optimization be used for engineering design taking practical considerations into account, and how should we develop optimization tools so they will be useful to engineers? This paper shall address this issue. The introduction describes the environment that practicing engineers operate under, and includes a brief description of a structural design area that I have worked in for several years. After the introduction, this paper describes several sequential random search methods for global optimization, and summarizes some theoretical results. This motivates the development of a class of Hit-and-Run algorithms. Finally, numerical experience applying these algorithms to structural design is discussed.

1.1. ENGINEERING ENVIRONMENT

Engineers always strive to optimize their designs – after all, who wouldn't want the strongest, cheapest, and overall best design? Yet in practice, engineers tend to compare a very few number of designs. There is a real need for practical optimization techniques for the engineering community. Even engineers that use gradient-based local optimization methods have commented that their problems tend to have many local optima [5, 34].

Very few engineers have considered using global optimization techniques, partly perhaps because the methods are still relatively new. The most common global optimization method found in practice is multi-start [5]. However newer methods are quite powerful, and with the advances in computer technology, are becoming very practical. In addition, many practicing engineers do not formulate their design problems as formal optimization problems. Hopefully, this paper and others like it will help bridge the gap between the advances in global optimization and the practical needs of the engineers.

To understand how global optimization can be used in practice, it is important to consider the design engineer's point of view. Many times the practicing engineer has very tight deadlines and must respond very quickly to changes in design. Consequently, they are primarily interested in any feasible design, and secondarily interested in an optimal design. In addition, the problem definition changes frequently so, for example, one day the optimization issue may be to minimize weight subject to buckling constraints and the next day may be to maximize stiffness under weight constraints. The optimization process may be most effective early in the design process, when decisions are made that determine a large percentage of the final cost of the product. This adds to the need for solving a lot of different problems and developing insight into the design space. Contrast this type of environment with a scheduling problem which may be solved repeatedly a hundred times a day but with slightly different data. The environment is very different and calls for a different approach to optimization. Instead of fine tuning a large scale integer linear programming algorithm, we must develop a flexible approach that be used for a variety of functions.

The functions used in the objective function and constraints are often termed 'black-box' functions, because the functions can be evaluated through a computer subroutine, but are difficult to express in a one-line mathematical formula. Characteristics of black-box functions, such as the gradient, and Hessian matrix, can be numerically approximated, but may often be numerically unstable. Black-box functions are very appealing to engineers because they can try out different existing subroutines and see the effect on the optimal solution. For example, someone who is trying to maximize strength might want to see how the design changes when different approximating equations are used for the strength calculation.

These black-box functions may be highly non-linear functions, and may involve many local optima. The functions may be nondifferentiable or even discontinuous. The design variables may be continuous or discrete, or a mixture of the two. For example, a 10-bar truss is a typical structural optimization problem. The design variables are the diameters of the bars. It is interesting to compare the optimal solutions for the problem when the diameters are assumed to be continuous variables, versus when the diameters are restricted to discrete values that are commonly available. This is an example of how manufacturing considerations can motivate discrete variables instead of continuous variables.

Global optimization methods that are practical for this type of environment cannot rely strongly on the structure of the functions, since the structure (convexity, continuity, etc.) changes so frequently. In addition, there is no time to tailor the algorithm when the functions change. So we are seeking flexible methods that can be used for black-box, mixed continuous/discrete, global optimization problems. Of course trade-offs must be made. In Section 2, trade-offs between the structure of the problem and the quality of solution will be discussed. Engineers in practice may be more satisfied with a quick solution that is feasible but sub-optimal, than with a globally optimal solution that takes months to obtain. This is demonstrated in the next example.

1.2. EXAMPLE – DESIGN OF A COMPOSITE PANEL

For several years I have worked with several colleagues to develop optimization software for the design of large composite structures, specifically laminated composite panels to be used in aircraft fuselage [3, 4, 33]. Through the collective efforts of NASA, Boeing, the University of Washington and others, a preliminary design software package called COSTADE (Cost/Composite Optimization Software for Transport Aircraft Design Evaluation) was developed [12, 13].

To simplify the problem for this paper, consider as an example of an engineering design problem, the design of a stiffened panel made from advanced composite materials, such as graphite epoxy, as illustrated in Figure 1. The design variables include the fiber angles associated with each ply (θ_i, for $i = 1, \ldots, n$), and the geometry variables for the stiffener (width of cap, width of flange, height of web, angle of web, and stiffener spacing), and the number of plies in the skin and in the stiffener (n). In this example, the binary variables (t_i, for $i = 1, \ldots, n$) indicate whether ply i exists in the laminate, and n is viewed as an upper bound on the number of plies. Some of these variables may be continuous and others may be discrete. The most interesting are the fiber angles, which may take on continuous values between $+90$ and -90 degrees, or, for practical purposes the fiber angles are often restricted to discrete values, such as 0, ± 45, or 90 degrees.

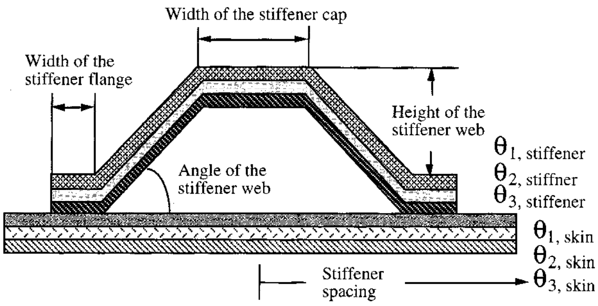

Figure 1. Design variables in a composite stiffened panel.

The functions to be used as constraints and/or the objective function can be described as black-box functions where the functions often are only available in the form of a computer subroutine. For example, stiffness of a composite laminate can be calculated using classical lamination theory [7] or it could involve a finite element analysis. The calculations in COSTADE are based on classical lamination theory, although recent work at Boeing and the UW has focused on incorporating finite element analyses into the global optimization design framework [16, 18].

To illustrate the global nature of these equations, a plot of the in-plane stiffness of a four ply, symmetric laminate, $[\theta_1, \theta_2, \theta_2, \theta_1]$, using classical lamination theory, is shown in Figure 2. The greatest in-plane stiffness occurs when the fiber angles are all 0 degrees, as makes sense intuitively. The plateau in the graph represents infeasible designs, with a stiffness that is less than a prescribed critical value. If stiffness is used as an objective function, it can be seen to be nonlinear and nonconvex. If stiffness is used in the constraints to allow only those designs above the threshhold of critical stiffness, the feasible region itself is nonconvex and even has holes in it. This indicates what a difficult problem even attaining feasibility can be.

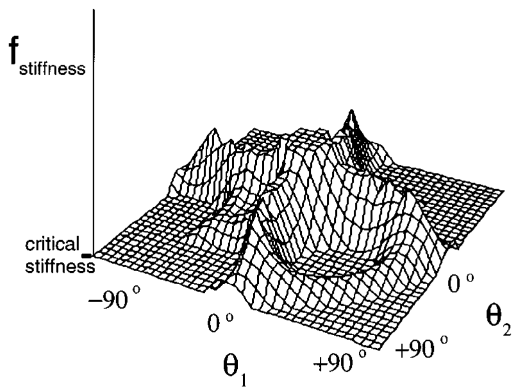

Figure 2. Graph of in-plane stiffness for a four ply symmetric laminate, $[\theta_1, \theta_2, \theta_2, \theta_1]$.

The basic formulation for the composites design problem can be written as,

$$(P)\quad \begin{array}{lll} \text{minimize} & f(x) & \\ \text{subject to} & g_j(x) \geq 0 & \text{for } j = 1, \ldots, m \\ & x \in X & \\ & L_i \leq x_i \leq U_i & \text{for } i = 1, \ldots, n \end{array}$$

where the objective function $f(x)$ may be the weight or cost of the structure, the m constraints ensure that margins of safety $g_j(x)$ based on mechanics calculations are positive, while allowing the design variables to be restricted to a discrete set of values, X. There are also upper and lower bounds on the variables. The variables include fiber angles, geometry variables, and binary thickness variables. There have been many extensions to this basic formulation.

A major extension has been to extend this formulation to cover an entire panel. As stated in (P), this problem prescribes a design at a specific point experiencing the loads as included in the margin of safety constraints. It is not realistic that the design of the fuselage is the same for the entire length of the aircraft. According to design engineers, different locations in the aircraft experience different loads, which calls for varying designs. For example, the number of plies in the keel near the mid-point of the aircraft just behind the opening for the landing gear may be one to two hundred, while the number of plies in the keel near the tail section is only twenty to thirty. Manufacturing considerations allow ‘ply drops’ to tailor the design at several points along the length of the aircraft. The COSTADE software designs a ‘blended panel’ with specific rules for manufacturing incorporated into an extended formulation [9, 27, 33, 35]. One effect of extending the formulation to cover an entire panel is that it increases the number of variables by an order of magnitude; from 20 variables to almost 200 variables.

Other extensions to the basic formulation include multiple objectives, multiple load cases, penalty function approaches, and incorporating finite element analysis into the calculation for margins of safety. The details of these extensions are left to other papers, the next topic to discuss is methods suitable to solve this type of engineering design problem.

2. Methods to solve these problems

Algorithms for global optimization are often categorized into deterministic methods and stochastic methods. An excellent overview of global optimization methods can be found in [6]. At the onset of our experience with the composites design problem, we tried several deterministic methods, including a branch and bound scheme coupled with gradient search local optimization methods. Our experience was that the numerical approximations for the Hessian matrix were very unstable, often leading to numerical errors. Also, the bounds seemed so loose that the branch and bound scheme produced enormous lists, and we were not able to solve problems with more than 10 design variables, due to both memory limitations and computation time. Another limitation to the deterministic approaches we tried was the inclusion of discrete variables. The need to incorporate mixed continuous/discrete variables, coupled with the attempt to solve black-box functions in a hundred variables led us to consider stochastic methods for global optimization.

The question remains whether random search algorithms are well-suited to this type of global optimization problem. Several features are important: (i) the quality of the solution and how close it is to the global optimum; (ii) the computation involved in the algorithm and the effect the number of variables has on the computation; and (iii) the ability to handle continuous and discrete variables and to easily interchange black-box functions.

2.1. SEQUENTIAL RANDOM SEARCH ALGORITHMS

Sequential random search algorithms at first glance are very simplistic methods, but they have certain advantages. They can be readily adapted to mixed continuous and discrete variables, and their simplicity makes them well suited to black-box functions. The basic form of a sequential random search algorithm is to have a method to generate a candidate point based on the current point and possibility several previous points, and then a test to either accept or reject the candidate point.

Rastrigin [20] originally proposed a family of step size algorithms where the candidate point is specifed by moving in direction D_k a certain step length S_k:

$$X_{k+1} = \begin{cases} X_k + S_k D_k & \text{if candidate point is accepted} \\ X_k & \text{if candidate point is rejected} \end{cases}$$

where X_k is the current point on the k^{th} iteration, and $X_k + S_k D_k$ is the candidate point. This form of algorithm was described by Rastrigin [15, 20, 21], and extended by several others [10, 24–26].

Typically the candidate point is accepted if it improves the objective function, and is rejected otherwise. In simulated annealing [8], the candidate point may be accepted even if it is not improving, and the acceptance criteria may be probabilistic and based on a temperature with a cooling schedule. The method of generating the candidate point may be described using a neighborhood rule, but simulated annealing still fits into the category of sequential random search. In general, a sequential random search algorithm can be described by

Generate a candidate point, W_k
Update the new point,

$$X_{k+1} = \begin{cases} W_k & \text{if candidate point is accepted} \\ X_k & \text{if candidate point is rejected} \end{cases}$$

In 1981, Solis and Wets [29] provided a general proof with generic conditions under which a sequential random search algorithm will asymptotically converge with probability one to the global optimum. To interpret the conditions of convergence quite loosely (see [29] for technical details), the method of generating and accepting candidate points must not consistently exclude any region in the design space. It can be shown [2] that simulated annealing converges in probability to the global optimum as long as the cooling schedule within the acceptance criteria is sufficiently slow, and generation method eventually includes the whole domain. These conditions are relaxed in [11], where simulated annealing is shown to converge in probability, even when the next candidate point is sampled from a distribution whose support is not the whole feasible set. See [11] for details. This addresses the issue of whether sequential random search methods will find the global optimum, the next issue is how long it will take.

Several papers on step size algorithms reported surprising performance for such simplistic sequential random search methods. In fact, there were several reports [24, 26, 29] that the number of function evaluations was linear in the number of variables. Most of these early reports were performed on a hyperspherical test function. Schumer and Steiglitz [26] concluded that ‘despite its simplicity, adaptive random search is an attractive technique for problems with large numbers of dimensions.’

An attempt to provide a general explanation for the reported linear performance of sequential random search was based on a theoretical analysis of pure adaptive search (PAS) [19, 36]. Pure adaptive search is a sequential random search method that consistently improves the objective function on each iteration. For a detailed definition, see [36], but to interpret loosely, each candidate point is generated according to a uniform distribution on the improving level set (set of points with improving objective function values). Although this step may be practically impossible to achieve with one function evaluation, suppose for the moment that it were possible. In [36] we proved that the expected number of iterations is a linear function of the dimension of the problem. This linearity result holds for a convex function and for a global optimization (non-convex) problem on a continuous domain.

In [38] the theoretical linearity result was extended to finite discrete global optimization problems as well. Current results [1, 32] extend performance results to a more general form of PAS, named hesitant adaptive search (HAS). Hesistant adaptive search relaxes the assumption that each iteration produces an improving point, and allows the algorithm to ‘hesitate’ before improving. An analytical expression for the number of function evaluations for HAS is given in terms of the underlying generating distribution and hesitation parameters. These theoretical results combined with numerical experience are encouraging that sequential random search algorithms are appropriate for the type of black-box functions in large dimensions that are characteristic of the engineering design environment. We now turn to an implementation of this concept.

2.2. HIT AND RUN BASED ALGORITHMS

The question remains how we can implement an algorithm with performance close to that of PAS on global optimization problems with black box functions and mixed continuous/discrete variables. In 1984, Smith [28] showed that a hit-and-run method can be used to generate points that are asymptotically uniformly distributed. The Improving Hit-and-Run (IHR) [37] algorithm couples the idea of PAS with the hit-and-run generator to produce a sequential random search algorithm that is easily implemented. The concept is that if hit-and-run could generate an approximately uniform point, then PAS predicts that we need only a linear number of such points. Since the point generated by just one iteration of hit-and-run is far from uniform, the number of function evaluations may not be linear, but in [37], we showed that the expected number of function evaluations for IHR on the class of elliptical programs is polynomial in dimension, $O(n^{5/2})$.

The main step of the IHR algorithm is (see [37] for details)

$$X_{k+1} = \begin{cases} X_k + \lambda_k D_k & \text{if } f(X_k + \lambda_k D_k) < f(X_k) \\ X_k & \text{otherwise} \end{cases}$$

where the direction vector D_k is generated as in hit-and-run (uniformly distributed on a hypersphere) and the step size λ_k is uniform over all step lengths that maintain feasibility. If the feasible region is described by linear equations, especially upper and lower bounds, then this is very easy to compute. Because this basic IHR algorithm exhibited good performance, both theoretically and computationally, several variations have been explored.

One variation to IHR was to add the idea of a simulated annealing algorithm and use an acceptance probability with a cooling schedule to the hit-and-run generator [22]. The pros and cons of only accepting improving points (as in IHR) versus accepting non-improving points according to an acceptance probability (as in simulated annealing) is a topic of current research [9,16].

A second variation to the basic IHR algorithm was to incorporate a penalty function approach for highly nonlinear constraints embedded in box constraints. A variable penalty factor was introduced, and the sequential updates of the penalty factor may be linked to the cooling schedule if desired. Some numerical experiments on attaining feasibility and maintaining feasibility are reported in [16].

A third variation to IHR was to extend the hit-and-run generator to be applicable to discrete domains [17, 23]. The hit-and-run as described so far was defined on a continuous domain. An extension to a discrete domain was accomplished by super-imposing the discrete domain onto a continuous real number system. It was motivated by design variables such as fiber angles in a composite laminate, or diameters in a 10-bar truss, where the discrete variables have a natural continuous analog. Two slightly different schemes are being tested.

In [23], the candidate points are generated using hit-and-run on the expanded continuous domain, but the objective function of a non-discrete point is equal to the objective function evaluated at its nearest discrete value. In this way, the modified algorithm is just IHR operating on a continuous domain where the objective function is a multi-dimensional step function, with plateaus surrounding the discrete

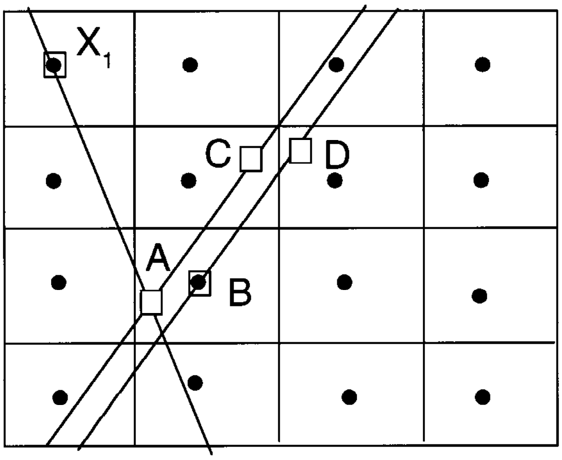

Figure 3. Two schemes to modify hit-and-run to discrete domains.

points. This modification still converges with probability 1 to the global optimum, as discussed in [23].

The diagram in Figure 3 illustrates the method. Starting from point X_1, hit-and-run on the continuous domain generates a candidate point such as A. The objective function at A is set equal to that of its nearest discrete point B, forcing $f(A) = f(B)$. If the candidate point is accepted, then $X_2 = A$, and another candidate point (shown as C) is generated.

A second scheme to modify IHR to operate on discrete domains is to similarly generate a point on a continuous domain, and then round the generated point to its nearest discrete point in the domain on each iteration [17]. Again starting from point X_1 in Figure 3, suppose A is generated. In this version, the candidate point is taken as the nearest discrete neighbor, in this example B. The objective function is evaluated at B, $f(B)$, and if the point is accepted, then $X_2 = B$. The difference in this variation is illustrated by noting that the next candidate point is generated from B instead of from A, see point D in the figure. Also note that only discrete points are maintained.

Other modifications to the basic hit-and-run generator include ways to generate the direction vector, and ways to generate the step length. Experiments with using only coordinate directions as compared to hyperspherical directions have been performed [34]. Experimentally, coordinate directions outperform hyperspherical directions on specific problems where the optimum is properly aligned, however other problems are easily constructed where it can be shown that just using coordinate directions will never converge [9]. The step length can also be generated in several ways. Instead of generating the point uniformly on the whole line segment, the line segment can be restricted to a fixed length, or adaptively modified. In COSTADE, this is referred to as full-line length, restricted line length, or adaptive step size. See [16] for a more detailed discussion of these modifications.

3. Numerical experience

Numerical experience with IHR and many of its variations has led us to believe that it is a viable approach for the type of engineering design problems described here. Details are left to other papers, but a summary of much of this experience is included.

Several experiments have been performed on test functions, where the global optimum is known. A sinusoidal function has been easy to use [9, 16, 23, 34]. It is easily adapted to multiple dimensions (e.g. 50), the global optimum is analytically described, it has a large number of local minima and it can also be easily adapted to discrete domains.

Other numerical experience is based on solving engineering design problems where the optimal solution is not known. In these cases, the engineers determine the maximum number of iterations, and then examine the solution generated. The design often provides insight to the problem at hand. During the development of COSTADE, several practical problems have been solved with different modifications to IHR. Originally we applied the basic IHR algorithm to optimize a flat plate with all continuous variables [3]. This was so successful we next optimized a skin-stringer composite crown panel [30], reducing the weight of the panel by over 30 percent. Cost was also reduced, and the effect of stringer spacing on cost emerged as a significant factor. Several points along the crown panel with different loads were optimized. We did not have an automated way to include blending into the optimization, so some manual blending was done. After that, the algorithm with several modifications was applied to the design of a sandwich keel panel [12]. This was a ‘blended panel’ and involved 160 variables, both continuous and discrete. The optimization software COSTADE was also used to evaluate the design of a window belt [14].

The sequential random search algorithm had been used separately to design a crown panel, keel and window belt. These independent panels still needed to be resolved into a full design. A remaining challenge was to expand the formulation to optimize the full barrel of an aircraft fuselage. Finally, in [16], we optimized the full barrel design of an aircraft fuselage using mixed continuous/discrete variables and some coarse finite element capability. At each level of success, the number of variables has grown, the computational time for the function evaluations has grown, and the additional insight into feasible practical designs has grown. This indicates success, because engineers have an optimization tool that is flexible enough to meet their needs.

Acknowledgments

The author is extremely grateful to Mark Tuttle for his patience and support in collaboratively developing the composites design application. The author would also like to thank Robert L. Smith for his insights and collaboration in the theoretical work on the stochastic algorithms. This work has been supported in part by NSF Grant DMI-9622433, Boeing and NASA.

References

1. Bulger, D.W. and Wood, G.R. (1998), Hesitant adaptive search for global optimisation, *Mathematical Programming* 81(1): 89–102.
2. Dekkers, A. and Aarts, E. (1991), Global optimization and simulated annealing, *Mathematical Programming* 50: 367–393.
3. Graesser, D.L., Zabinsky, Z.B., Tuttle, M.E. and Kim, G.I. (1991), Designing laminated composites using random search techniques, *Composite Structures* 18: 311–325.
4. Graesser, D.L., Zabinsky, Z.B., Tuttle, M.E. and Kim, G.I. (1993), Optimal design of composite structures, *Composite Structures* 24: 273–281.
5. Haftka, R.T. and Gürdal, Z. (1992), *Elements of Structural Optimization*, Kluwer Academic Publishers, Dordrecht/Boston/London.
6. Horst, R. and Pardalos, P.M. (1995), *Handbook of Global Optimization*, Kluwer Academic Publishers, Dordrecht/Boston/London.
7. Jones, R.M. (1975), *Mechanics of Composite Materials*, Scripta Book, Washington, D.C.
8. Kirkpatrick, S., Gelatt, C.D. Jr. and Vecchi, M.P. (1983), Optimization by simulated annealing, *Science* 220(4598): 671–679.
9. Kristinsdottir, B.P. (1997), *Analysis and Development of Random Search Algorithms*, Ph.D. Dissertation, University of Washington, Seattle.
10. Lawrence, J.P. III and Steiglitz, K. (1972), Randomized pattern search, *IEEE Transactions on Computers* C-21: 382–385.
11. Locatelli, M. (1996), Convergence properties of simulated annealing for continuous global optimization, *Journal of Applied Probability* 33: 1127–1140.
12. Mabson, G.E., Flynn, B.W., Ilcewicz, L.B. and Graesser, D.L. (1994), The use of COSTADE in developing composite commercial aircraft fuselage structures, *Proceedings of the 35th AIAA/ASME/ASCE/AHS/ASC Structure, Structural Dynamics, and Materials Conference*, Hilton Head, SC.
13. Mabson, G.E., Fredrikson, H.G., Graesser, D.L., Metschan, S.L., Proctor, M.R., Stogin, D.C., Tervo, D.K., Zabinsky, Z.B., Tuttle, M.E. and Gutowski, T.G. (1995), Cost optimization software for transport aircraft design evaluation, *Proceedings of the 6th NASA/DOD/ARPA Advanced Composite Technology Conference*.
14. Metschan, S.L., Graesser, D.L., Mabson, G.E., Proctor, M.R., Tervo, D.K. and Ilcewicz, L.B. (1994), Manufacturing data for costade analysis of composite fuselage panels, *Proceedings of the 5th NASA ACT Conference*.
15. Mutseniyeks, V.A. and Rastrigin, L.A. (1964), Extremal control of continuous multi-parameter systems by the method of random search, *Engineering Cybernetics* 1: 82–90.
16. Neogi, S. (1997), *Design of Large Composite Structures using Global Optimization and Finite Element Analysis*, Ph.D. Dissertation, University of Washington, Seattle.
17. Neogi, S., Zabinsky, Z.B. and Tuttle, M.E. (1994), Optimal design of composites using mixed discrete and continuous variables, *Proceedings of the ASME Winter Annual Meeting, Symposium on Processing, Design and Performance of Composite Materials*, MD-Vol. 52, pp. 91–107.
18. Neogi, S., Zabinsky, Z.B., Tuttle, M.E. and Kristinsdottir, B.P. (1995), Load redistribution issues in optimal design of composites airplane structures, *Proceedings of the ICCE/2 Conference*, New Orleans.
19. Patel, N.R., Smith, R.L. and Zabinsky, Z.B. (1988), Pure adaptive search in Monte Carlo optimization, *Mathematical Programming* 4: 317–328.
20. Rastrigin, L.A. (1960), Extremal control by the method of random scanning, *Automation and Remote Control* 21: 891–896.
21. Rastrigin, L.A. (1963), The convergence of the random search method in the extremal control of a many-parameter system, *Automation and Remote Control* 24: 1337–1342.
22. Romeijn, H.E. and Smith, R.L. (1994), Simulated annealing for constrained global optimization, *Journal of Global Optimization* 5: 101–126.

23. Romeijn, H.E., Zabinsky, Z.B., Graesser, D.L. and Neogi, S. (1995), Simulated annealing for mixed integer/continuous global optimization, technical report Erasmus University. Forthcoming in *Journal of Optimization: Theory and Applications* 101(1) (April 1999).
24. Schrack, G. and Borowski, N. (1972), An experimental comparison of three random searches, in F. Lootsma (ed.), *Numerical Methods for Nonlinear Optimization*, pp. 137-147, Academic Press, London.
25. Schrack, G. and Choit, M. (1976), Optimized relative step size random searches, *Mathematical Programming* 10: 230–244.
26. Schumer, M.A. and Steiglitz, K. (1968), Adaptive step size random search, *IEEE Transactions on Automatic Control* AC-13: 270–276.
27. Seifer, J.D. (1995) Incorporating realistic engineering considerations into the optimal design of composite structures, Master's Thesis, University of Washington, Seattle.
28. Smith, R.L. (1984), Efficient Monte Carlo procedures for generating points uniformly distributed over bounded regions, *Operations Research* 32: 1296–1308.
29. Solis, F.J. and Wets, R.J.-B. (1981), Minimization by random search techniques, *Mathematics of Operations Research* 6: 19–30.
30. Swanson, G.D., Ilcewicz, L.B., Walker, T., Graesser, D.L., Tuttle, M.E. and Zabinsky, Z.B. (1991), Local design optimization for transport fuselage crown panels, *Proceedings of the 9th DoD/NASA/FAA Conference on Fibrous Composites in Structural Design*, Lake Tahoe, Nevada.
31. Vanderplaats, G.N. (1993) Thirty years of modern structural optimization, *Advances in Engineering Software* 16: 81–88.
32. Wood, G.R., Zabinsky, Z.B. and Kristinsdottir, B.P. (1997), Hesitant adaptive search: the distribution of the number of iterations to convergence, technical report, Central Queensland University.
33. Zabinsky, Z.B. (1994), Global optimization for composite structural design, *Proceedings of the 35th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Material Conference*, Hilton Head, SC.
34. Zabinsky, Z.B., Graesser, D.L., Tuttle, M.E. and Kim, G.I. (1992), Global optimization of composite laminate using Improving Hit and Run, in C.A. Floudas and P.M. Pardalos (eds.), *Recent Advances in Global Optimization*, pp. 343–365, Princeton University Press, Princeton.
35. Zabinsky, Z.B., Seifer, J.D., Tuttle, M.E., Kristinsdottir, B.P. and Neogi, N. (1995), Optimal design of composite panels with varying loads, *Proceedings of the ICCE/2 Conference*, New Orleans.
36. Zabinsky, Z.B. and Smith, R.L. (1992), Pure adaptive search in global optimization, *Mathematical Programming* 53: 323–338.
37. Zabinsky, Z.B., Smith, R.L., McDonald, J.F., Romeijn, H.E. and Kaufman, D.E. (1993), Improving Hit and Run for global optimization, *Journal of Global Optimization* 3: 171–192.
38. Zabinsky, Z.B., Wood, G.R., Steel, M.A. and Baritompa, W.P. (1995), Finite pure adaptive search, *Mathematical Programming* 69: 443–448.