ORNL/TM-13737

ornl

OAK RIDGE
NATIONAL
LABORATORY

LOCKHEED MARTIN

STP: A STOCHASTIC TUNNELING ALGORITHM FOR GLOBAL OPTIMIZATION

E. M. OBLOW

MANAGED AND OPERATED BY
LOCKHEED MARTIN ENERGY RESEARCH CORPORATION
FOR THE UNITED STATES
DEPARTMENT OF ENERGY

ORNL-27 (3-96)

ORNL/TM-13737

SPT: A Stochastic Tunneling Algorithm for Global Optimization

E. M. Oblow
Computer Science and Mathematics Division

DATE PUBLISHED — April 1999

Research sponsored by the
Basic Energy Sciences Program of the
U.S. Department of Energy

Prepared by the
OAK RIDGE NATIONAL LABORATORY
Oak Ridge, Tennessee 37831
managed by
LOCKHEED MARTIN ENERGY RESEARCH CORP.
for the
U.S. DEPARTMENT OF ENERGY
under Contract No. DE-AC05-96OR22464

Contents

List of Figures v

List of Tables vii

Acknowledgements ix

Abstract xi

1 Introduction 1

2 Problem Formulation 2

3 A Stochastic Pijavskij Algorithm 3

4 The General SPT Algorithm 5

5 Lower Bound Approximations 7

6 Pseudo-Lipschitz Approximations 8

7 One-Dimensional Approximations 9

8 Science Benchmark Results 11

9 Large-Scale Application 11

10 Conclusions 15

List of Figures

1 Pijavskij scheme 4
2 SPT scheme iterations 6
3 Two-dimensional exclusion regions 10
4 Unoptimized subsurface seismic map 14
5 Optimized subsurface seismic map 15

List of Tables

1 Benchmark Results 12

Acknowledgements

This research was performed at the Center for Engineering Science Advanced Research (CESAR), Computer Science and Mathematics Division, Oak Ridge National Laboratory. Funding was provided by the Engineering Research Program of the Office of Basic Energy Sciences and by the Advanced Computing Technology Initiative Program of the U.S. Department of Energy under contract DE-AC05-96OR22464 with Lockheed Martin Energy Research Corp.

Abstract

A stochastic approach to solving continuous function global optimization problems is presented. It builds on the tunneling approach to deterministic optimization presented by Barhen et al. [1,2] by combining a series of local descents with stochastic searches. The method uses a rejection-based stochastic procedure to locate new local minima descent regions and a fixed Lipschitz-like constant to reject unpromising regions in the search space, thereby increasing the efficiency of the tunneling process. The algorithm is easily implemented in low-dimensional problems and scales easily to large problems. It is less effective without further heuristics in these latter cases, however. Several improvements to the basic algorithm which make use of approximate estimates of the algorithms parameters for implementation in high-dimensional problems are also discussed. Benchmark results are presented, which show that the algorithm is competitive with the best previously reported global optimization techniques. A successful application of the approach to a large-scale seismology problem of substantial computational complexity using a low-dimensional approximation scheme is also reported.

Keywords and Phrases: Global optimization, tunneling, stochastic, Pijavskij

1 Introduction

In two recent papers, Barhen et al. [1,2] reported a significant improvement in solving low-dimensional global optimization benchmark problems. They developed a code called TRUST, which employs a descent-tunneling methodology characterized by non-Lipschitzian descent and one-dimensional (1-D) search. The essence of this approach is to break the solution of an optimization problem down into two distinct phases: 1) local descent and 2) tunneling (using 1-D search). The descent phase makes use of local derivative information and the wealth of algorithmic experience in finding local minima in continuous functions. The search phase utilizes a generalization of the recent concept of tunneling [3] to find a new region in which to begin to descend again. Instead of searching for zeros of a polynomial as in [3], however, a one-dimensional search scheme is used to tunnel to the basin of attraction of a new local minimum. What makes this combined approach novel and efficient is that by descending into a local minimum before beginning to search for a new descent region, the method avoids repeated descents to the same local minimum. The final solution is thus, in effect, a single descent interspersed with 1-D searches to find the descent basins of new local minima. A key characteristic of this method, using 1-D search to tunnel to new basins, is an attractive feature because in general the ratio of a basin's perimeter to its volume favors a 1-D search over multi-dimension (volume) search for anything but a hyperspherical basin [4]. Both of these characteristics seem to explain the improvements Barhen et al. reported in benchmark testing using the TRUST methodology for a paper in *Science* [2].

Starting with the Barhen et al. descent-tunneling methodology as the basis for our research efforts, it is clear that for higher dimensional problems the search phase of the algorithm needs improvement. In high-dimensional spaces, search is much more costly in terms of function evaluations than descent. In order to move from benchmarks to realistic high-dimensional problems, therefore, further improvements must be made in TRUST's search methodology. The most limiting characteristic of search, however, is that *local* derivative information is of little (or, in principle, no) use in locating a new basin of attraction different from those already found. We must, therefore, explore what other information can be used to help in this search process. Most recent research indicates that branch and bound approaches address this problem theoretically but are complex and costly to implement in practice (see for example [5,6]). The bounding process allows certain regions of the search space to be eliminated from further function evaluations.

It is also widely understood that practical global optimization problem cannot be solved in general without imposing resolution or probabilistic convergence constraints on its solution [7]. Without such constraints it is impossible to know how to halt an algorithm and declare that the lowest-to-date minimum is the global minimum. Only an exhaustive bounded search is guaranteed to find a global minimum since no local information can ever be used to characterize the current best answer as definitively being the global minimum. From a practical standpoint, we found that probabilistic constraints are easier to implement and have, therefore, concentrated our efforts in this area. In this approach only a probabilistic guarantee can be given that the best local minimum found is the global minimum. There is always a chance that another lower minimum will be found, albeit within a user specified probability constraint.

The comments above lead us to believe that a stochastic approach to tunneling-search coupled with some limited knowledge of function bounds and Lipschitz-like constants can be made the basis of a powerful yet simple algorithm to improve Barhen's approach. To make maximum use of such bounding information, a low (or one) dimensional stochastic tunneling approach to covering a high-dimensional space seems to be necessary. Since we are committed to a *stochastic* reformulation of Barhen's method, we reiterate that only a probabilistic guarantee can be made on deciding whether a global minimum has been found. A halting algorithm which stochastically characterizes the solution is thus our aim. We will use a simple PAC (Probabilistically Approximately Correct) criterion successfully developed to halt machine learning algorithms [8]. Without stochastic characterization the only other alternative would be to solve the problem within prescribed resolution bounds using a fixed grid and an associated fixed number of function evaluations.

We propose, therefore, to only make use of global bounding information in a low-dimensional stochastic algorithm to significantly improve the tunneling phase of Barhen's TRUST algorithm. In doing so we utilize the already powerful breakup of the problem into local descent and tunneling. To accomplish this, we adapt the classic Pijavskij one-dimensional bounding procedure [9] to higher dimensional problems. This extension eliminates many of the problems encountered in the past with this bounding approach and produce a stochastic version of Pijavskij's method useful exclusively in the search phase. The new approach is called SPT: (S)tochastic (P)ijavskij (T)unneling, suitable for the tunneling-search phase of Barhen's algorithm. This methodology is closely allied with similar extensions of bounding-covering algorithms published by Shubert [10], Evtushenko [11], and Mladineo [12].

We apply the stochastic improvements developed directly to the combined descent and tunneling TRUST algorithm. This allows us to revisit the benchmark calculations Barhen et al. reported in *Science* [2], to see what improvements can be made in these already good results. We will also solve a new high-dimensional seismic problem similar to the one reported in the *Science* paper to demonstrate the method's scalability.

2 Problem Formulation

The generic global optimization problem considered in this paper is defined as follows. Let $f(\mathbf{x}) : \mathcal{D} \rightarrow \mathcal{R}$ be a bounded piecewise-continuous function and $\mathbf{x}$ be an n-dimensional state vector. The function $f(\mathbf{x})$ is referred to as the objective function, and $\mathcal{D}$ its domain. The optimization goal is to find the state vector $\mathbf{x}^*$ which minimizes $f(\mathbf{x})$ in $\mathcal{D}$. Without loss of generality, we shall take $\mathcal{D}$ to be the hyperparallelopiped $\mathcal{D} = \{x_i \mid \beta_i^- \leq x_i \leq \beta_i^+ \; ; \; i = 1, 2, \ldots, n\}$,where β_i^- and β_i^+ denote, respectively, the lower and upper bound of the i-th state variable.

We address this unconstrained global optimization problem by breaking its solution into two phases. The first, is a standard local descent phase in which local derivative information is used to converge a descent algorithm to a local minimum. Any efficient descent algorithm can be chosen for this phase of the work. Many finite difference methods, as well as Jacobian or Hessian based schemes, are available for algorithmic implementation [13-19]. Once a local minimum has be found, the second phase will employ a stochastic search to find a new local minimum basin of attraction to descend into. This procedure is employed to avoid costly repeated descents into already identified local minima, a characteristic weakness of multiple random start algorithms [13]. It should be noted that this step represents an algorithmic decision that might not be optimal for all problem classes. Specifically, this approach performs less efficiently in problems characterized by the existence of a hierarchy of local minima that has increasing numbers of narrower and narrower minima as the function values decrease. New minima become more difficult to find because of decreasing basin sizes in such cases.

To improve the search phase of Barhen's algorithm we use a new stochastic version of Pijavskij's one-dimensional bounding algorithm. The deterministic version of this algorithm [9] has been widely studied in global optimization work in the past. Its geometric complexity in higher dimensions and its slow convergence even in one-dimensional cases have often been cited as the reasons for its limited usefulness in most practical problems [13]. By developing a stochastic version of this traditional algorithm and using it only in the search phase of our methodology, we hope to eliminate these problems entirely. The newly developed SPT (Stochastic Pijavskij Tunneling) algorithm is easily implemented in one-dimension and scales well to higher dimensional problems, although its usefulness decreases exponentially in this latter case.

3 A Stochastic Pijavskij Algorithm

A brief review of Pijavskij's one-dimensional deterministic algorithm is in order here so that the improvements afforded by a new stochastic implementation can be understood more clearly. The principle used by Pijavskij in his original paper [9] and also employed by Shubert [10], was to utilize the Lipschitz constant to determine a lower bound for the global minimum of a one-dimensional function. We assume the function f is continuously differentiable and define the Lipschitz constant as: $L = (|f(x) - f(y)|/|x - y|)_{max}, \forall x, y \in \mathcal{D}, x \neq y$. In practice, it specifies the largest rate of descent a function can have over a region of interest. If the function is known for two values of x, then drawing lines with slope L (called L-lines for future reference) from both of them generates a lower bound f_L for the function in the region between the two given values of x. Using a zeroth-order estimate of this lower bound (i.e. f_L^0) and the point at which it occurs x_{min}^0, Pijavskij's method prescribes that a guess for the position of the global minimum x^* should be taken at $x^* = x_{min}^0$, the intersection of the two L-lines. That is, $f(x_{min}^0)$ should be evaluated to make a new estimate of the global minimum of the function with the guarantee that $f^* \geq f_L^0$. This procedure is illustrated in Figure 1 using two regions in $[a, b]$ generated by the arbitrary point x_1 (i.e. $x \in [a, x_1]$ and $x \in [x_1, b]$).

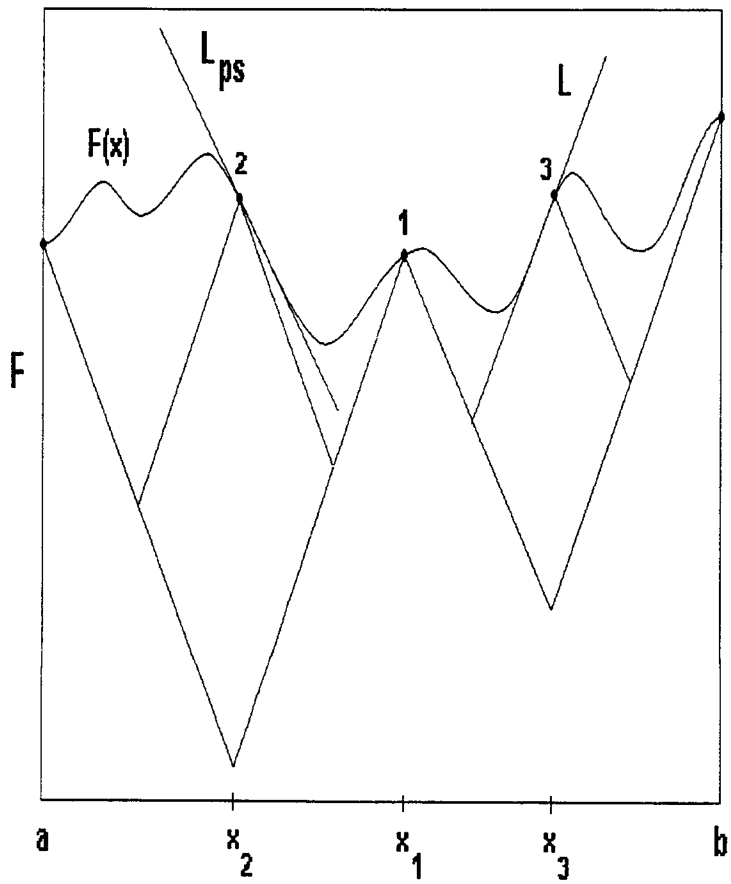

Figure 1: Pijavskij scheme

The Pijavskij approach is iterative in nature, employing a repetitive sequence of subinterval bounding steps similar to the one described above. Evaluation of the function at each new point leaves two new intervals in which the algorithm can be applied to find lower regional bounds. By prescribing the next evaluation point to be the x value of the lowest of these two newly generated bounds (or the lowest of all the others that might exist from previous bounded intervals) an iterative saw-toothed lower bounding function is developed for the function under investigation. The results of this continued sequence is also shown in the Figure 1 with steps for generating x_2 and x_3.

From this description of Pijavskij's scheme it should be clear that because no descent is performed this approach becomes increasingly inefficient for flatter regions of the search space (i.e. as the global minimum is approached). Local descent making use of local derivative information is clearly much more efficient in such regions. In addition, the extension of the method to higher dimensions makes finding new points to evaluate much more difficult. Multidimensional surface intersections are needed in these cases to find the lower bound points which prescribe the next evaluation point. Several approaches have been used to derive more useful alternatives to this difficult multidimensional task [10-13] but they are all more complex and inefficient in the large problems of practical interest. These later solutions are characterized by using multidimensional rectangular or conical elements. We will refer to these two approaches as the a P-cone and P-cube approaches in the rest of this paper. Clearly an approach that combines descent and bounded search will produce a more efficient scheme than the extensions to Pijavskij's method published to date.

To address this point, we propose to use the key characteristics of Pijavskij's algorithm, the L-lines (and, as needed in higher dimensions, P-cones or P-cubes), in just the search phase of an optimization algorithm to prescribe the next evaluation point. In Figure 2 we show a sequence of five randomly selected points (labelled 1-5) which implement this new scheme. Looking this figure we note that if point x^0_{min} was the first local minimum already found by a descent algorithm, the regions from $[a_{intersect}, x^1_{intersect}]$ and $[x^0_{min}, b_{intersect}]$ in which the L-lines intersect the value f^0_{min} represent regions in which the function could have a value lower than f^0_{min}. Pijavskij's deterministic algorithm chooses the lowest intersection point of the L-lines (f^0_L) as the next point to evaluate the function. Conversely, the interval $[x^1_{intersect}, x^0_{min}]$ represents a region in which f must be at least as large as f_{min} and is thus a region which should be excluded from further consideration in finding values of f lower than f^0_{min}, the current lowest minimum value found. This reverse reasoning defines the concept of exclusion regions and provides the basis for developing a stochastic version of Pijavskij's basic scheme for finding new basins of attraction for a descent algorithm to work in.

To produce a basic stochastic algorithm then, we can use a *region-of-possibility* for finding a point lower than the current lowest local minimum to define a sampling region for selecting a new point for evaluation. Pijavskij's algorithm essentially takes the midpoint of the lowest subregion of this region. A stochastic algorithm with a uniform distribution over the whole subregion would have Pijavskij's point as its average value. Sampling over all possible subregions is simpler, however, and such a procedure should, on average, follow a sequence of Pijavskij-prescribed evaluations.

A complementary algorithm using a rejection technique can also be used to meet this end. This approach would randomly sample over the whole x-domain and reject sample points which lie outside the *region-of-possibility* (i.e. those in the excluded region) until an acceptable point is found. Both techniques have their advantages and disadvantages. They both require search and storage operations to determine intervals of possibility or exclusion and they are the complements of each other in an algorithmic sense. Because of the difficulties entailed in extending a direct sampling method to higher dimensions and the natural availability of a halting test based on the probability of finding a point lower than any one found before, we chose to implement the rejection technique for the SPT algorithm.

4 The General SPT Algorithm

The general stochastic multi-dimensional SPT algorithm can therefore be described as follows:

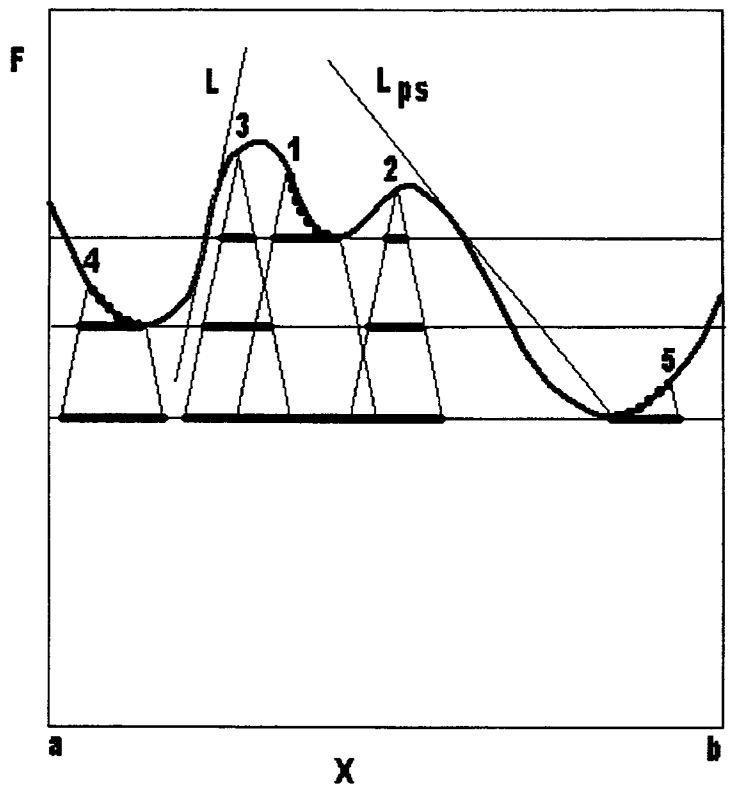

Figure 2: SPT scheme iterations

- 1. Select a random starting point $\mathbf{x} = \mathbf{x^1}$ and set $f^* = f^1_{min} = f(\mathbf{x^1})$.
- 2. Use a descent algorithm to converge to local minimum evaluating $N - 1$ lower function values in the process. Set $f^* = f^N_{min} = f(\mathbf{x^N})$
- 3. Compute Pijavskij P-cone radii r_i for all N points evaluated so far as follows: $r_i = (f^i - f^*)/L, \forall \mathbf{x}^i$ with $i \in N$.
- 4. Select new random points from the region outside of all P-cones intersecting f^*, searching for a value of f less than f^*. Halt this procedure if more than N_{max} points are selected without finding a value of f lower than f^*, where $N_{max} = \log\delta/\log\epsilon$. This constitutes a stochastic rejection technique to get $\mathbf{x^{N_{max}}}$ points at which to evaluate for values potentially lower than f^*.
- 5. Go to [2.] if item [4.] is successful in finding an $f < f^*$, otherwise halt and report $f^* = f_{globalmin}$.

In general this algorithm attempts to make a stochastic ϵ-cover of the $\mathbf{x}$-domain with confidence δ (see [8]) using the only information on the Lipschitz constant to exclude certain unnecessary function evaluations. The larger a function value is compared to the lowest value found to date, the larger the exclusion region is for finding new function evaluation possibilities. The exclusion process is thus aided by large function values and lower minimum values. At some point before N_{max} is reached it is possible that the exclusion region will cover the whole $\mathbf{x}$-space and the algorithm will halt having found the global minimum within the prescribed stochastic confidence limit δ for the chosen ϵ-cover.

Based on the description above, this new procedure can best be described as an importance sampling technique for stochastically finding a value of $f(\mathbf{x})$ lower than an existing estimate of f^*. The bias of the sampling is toward regions of the phase space that have lower function values than the current estimate of the global minimum f^*. Used in conjunction with a local descent algorithm, it goes from local minimum to local minimum to try to find the global minimum within prescribed stochastic confidence limits.

5 Lower Bound Approximations

A striking feature of the algorithm given in the last section is the fact that the larger the function values are compared to the current minimum value, the more x-space is excluded from future function evaluation. Thus, larger function values make the method more efficient. This feature can be used to great advantage by adding a single additional piece of information to the algorithm to improve its efficiency even more. This piece of information is an estimate of the lowest value the function can attain in the domain of interest f_M (i.e. an estimate of the global minimum). Instead of letting the algorithm develop local minima estimates of the global minimum as it proceeds, a low estimate can be input right from the start. This added parameter will immediately increase the efficiency of the algorithm by increasing the exclusion region of potential function evaluations. This efficiency increase is always present until a local minimum is found which has a value lower than the estimate.

This estimation process can be made more formal and adaptive by simply using the Lipschitz constant already being used. L can be used directly to get a lower bound for this new parameter f_M from $f_M = max(f(x_0), f(x_1))L$ and then its value can be adaptively changed during the course of the algorithm. If too low a value is used the algorithm will in general halt due to a complete exclusion of the x-domain. At this point a halving scheme can be used to reduce the estimate to halfway between the lowest value of f found to date f^n_{min} and the current estimate for the global minimum f_M. This new estimate will still be lower than f^n_{min} and therefore will improve the algorithms efficiency accordingly. The new estimate, moreover, requires no new function evaluations and simply changes the values of the current set of P-cones in time linear with N. No additional function evaluations performed and at worst it is at least as efficient as a scheme which uses the current local minimum for an estimate of f_M.

6 Pseudo-Lipschitz Approximations

As stated in section 2 the general algorithm requires an estimate of the Lipschitz constant as the only additional piece of information used to increase search efficiency. Since analytic computer evaluated functions are our focus, the availability of a good Lipschitz constant is assumed to exist. A closer examination of the current algorithm however, reveals that a true Lipschitz constant is not really required for this algorithm. What is required is a pseudo-Lipschitz constant, defined as the largest slope of any line drawn from the global minimum tangent to the walls of the basin of attraction of that global minimum. This constant called L_{ps} guarantees that no function evaluation will ever cause the global minimum point to be excluded from potential function evaluation. The tangent point with the maximum slope is the point at which the global minimum is finally excluded from evaluation by a P-cone drawn from the tangent point. It should be noted that even a square-well with infinite Lipschitz constant still has a finite L_{ps} provided only a single global minimum is to be found out of a potential infinity of possible values. This concept is thus very powerful in practice if a way can be found to estimate L_{ps} rather than L.

The difficulty with this new concept is, however, precisely its estimation. In principle this cannot be done without knowing a great deal of information about the size and shape of the global minimum basin, the very thing we presumably are trying to find. In practice however we can use an adaptive procedure to estimate this parameter since we are ultimately willing to accept only a probabilistic answer to the global optimization problem anyway. A possible procedure for estimating this constant is to use a real Lipschitz constant as the basis for a first cut estimate of L_{ps}. Using L itself will either cause the algorithm to halt on the exclusion condition or the N_{max} limit. If the latter limit is achieved nothing further can be done. But, if the algorithm halts on an exclusion limit then a halving (of the Lipschitz constant) approach can be used to begin estimating L_{ps}. This philosophically means that the solution has not yet reached the N_{max} limit so acceptable solutions have not been exhausted even though no more sampling is possible because the exclusion limit has been reached. Since the real $L_{ps} \leq L$, we still have a chance of opening up further sampling regions by decreasing the estimate of L_{ps} below that of its upper limit of L. Therefore, if we adopt a halving scheme for estimating L_{ps}, we can continue to sample until the N_{max} limit is reached and our original problem solution condition is met. Thus we set $L_{ps} \approx L/2$ and then reevaluate all P-cones using the new estimate of L_{ps} without any further function evaluations. The algorithm can then be reinitialized with this new smaller value of L_{ps}. The halving continues as each new exclusion limit is reached until the N_{max} limit is reached and the problem is solved to our prescribed probabilistic confidence.

It should be noted here that this adaptive algorithm will always halt on the N_{max} limit being reached. But this does not mean that N_{max} functions have been evaluated. The real efficiency of the algorithm will depend on the size of the final exclusion region after halting takes place. In practice orders of magnitude less function evaluations might be needed to achieve a probabilistic confidence equivalent to a fixed grid of $O(N_{max})$. For a relatively flat (or even constant) function, N_{max} function evaluations will probably be approached, but few other algorithms can be conceived of that will do much better than this on such a problem.

7 One-Dimensional Approximations

In practice this algorithm suffers from one limiting characteristic: its efficiency compared to pure random search diminishes as the dimension of the x-space increases. It becomes increasingly difficult to cover higher dimensional spaces with P-cones based on a single Lipschitz-like constant. In practice the exclusion fraction approaches zero in high dimensional problems and the algorithm is effectively reduced to a pure random search. We expect the method to be useful in low-dimensional problems (as will be demonstrated in the next benchmark testing section) but practically only random search is achieved in very large-dimensional problems.

A little further analysis reveals, however, that the maximum efficiency of the method can be extended to higher dimensional problems by accepting a very useful approximation to basic SPT approach. This approximation is based on the construction of a multidimensional Lipschitz constant from the partial derivative information if available. This construction is illustrated here by assuming that in each individual x-direction a maximum Lipschitz constant $L_x, \forall x \in \mathcal{D}$ can be estimated. The real Lipschitz constant in d-dimensions is thus: $LO(\sqrt{d}L_x)$. A graphical picture of this process and the resulting P-cone in two-dimensions is shown in Fig. 3. It is clear that L is always larger than any of the individual $L_x's$ and thus the P-cone is smaller than that derived from any individual L_x. If we take the real shape of the excluded region based on the true maximum directional derivative in all directions we get a starfish shaped region which is tangent to the P-cones in d-dimensions and the L-lines in each individual one-dimensional direction. As larger dimensional problems are considered L scales up as $\sqrt{d}$ and the P-cone exclusion region becomes decreasingly smaller.

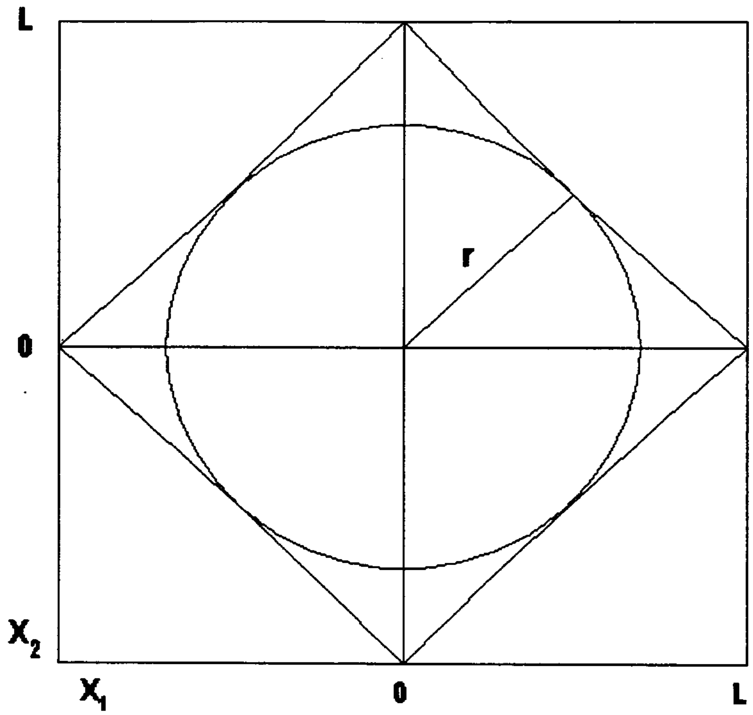

Figure 3: Two-dimensional exclusion regions

The starfish effect becomes, in the limit of very high dimensions, almost a collection of one-dimensional exclusion problems with no measurable exclusion volume in d-dimensions. This means in practice that the algorithm would be no better than pure random search, since the P-cones would have almost negligible measure and the algorithm will halt when the N_{max} limit is reached (rather than on any exclusion region halt).

The above facts cannot be avoided in principle, but if the algorithm is recast into a search of each individual direction starting from a random point in d-dimensions, then d very efficient one-dimensional searches can be used to solve the larger problem with some significant efficiency advantages accruing because of availability of L_x information. Thus, in practice the maximum efficiency of one-dimensional problem exclusions can be transferred to higher dimensional problems with only minimal approximations being used. A d-dimensional problem is thereby reduced to a collection of d uncoupled one-dimensional problems using the most efficient use of the Lipschitz or pseudo-Lipschitz information.

The one-dimensional implementation of this approach is thus the recommended one for higher dimensional problems. It effectively produces a random line cover of a d-dimensional space with an underlying grid of density N_{min}^d points. Of these points, only a fraction of them are required for function evaluation depending on the 1-D exclusion regions that develop iteratively as more function evaluations are made. In this sense it is guaranteed to be at least as efficient as a fixed N_{min}^d grid evaluation of a d-dimensional function. The simplicity and speed of the algorithm should give it a competitive advantage over most existing approaches to global optimization in high dimensional problems. This advantage is demonstrated well in the 314-dimensional seismic benchmark we present later in this paper.

8 Science Benchmark Results

This section presents results of benchmarks carried out to assess the new SPT algorithm using several standard multidimensional test functions taken from the literature. A description of each test function can be found in [1,20]. In Tables 1-2, the performance of SPT is compared to the best competing global optimization methods including the previous version of TRUST published in *Science* [2]. Here the term "best" indicates the best widely reported reproducible results the authors could find for the particular test function. The criterion for comparison is the number of function evaluations. All SPT results were converged to a convergence criteria of $\epsilon = 10^{-5}$.

In Table 1, the benchmark labels BR (Branin), CA (Camelback), GP (Goldstein-Price), RA (Rastrigin), SH (Shubert), and H3 (Hartman), refer to the test functions considered. The following abbreviations are also used: SDE is the stochastic method of Aluffi-Pentini [21]; EA denotes the evolution algorithms of Yong, Lishan, and Evans [22], or Schneider [23]; MLSL is the multiple level single linkage method of Kan and Timmer [24]; IA is the interval arithmetic technique of Ratschek and Rokne [25]; TUN is the tunneling method of Levy and Montalvo [3]; and TS refers to the Taboo Search scheme of Cvijovic and Klinowski [20] and TRUST refers to Barhen's TRUST code [2]. The results demonstrate that TRUST is substantially faster than these state-of-the-art method, albeit needing a number of parameters to be optimized for each benchmark. The SPT results are comparable to the TRUST numbers and use only a single adjustable parameter and are averaged over 100 random (vs 2^d deterministic) runs.

9 Large-Scale Application

To assess the performance of SPT for a large-scale practical application, the high-dimensional problem of residual statics corrections for seismic data was chosen. In many geophysical tasks seismic energy is detected by receivers that are regularly spaced along a grid that covers the domain being explored. A source is positioned at some grid location to produce a shot. Time series data is collected from the detectors for each shot, then the source is moved to another grid node for the next shot.

A major degradation of seismic signals usually arises from near-surface geologic irregularities [26-28]. These include uneven soil densities, topography, and significant lateral variations in the velocity of seismic waves. The most important consequence of such irregularities is a distorted image of the subsurface structure, due to misalignment of signals caused by unpredictable delays in recorded travel times of seismic waves in a vertical neighborhood of every source and receiver. To improve the quality of the seismic analysis, timing adjustments (i.e., "statics corrections") must be

Table 1. Number of function evaluations required by different methods to reach a global minimum of several Standard Test Functions.

Method	BR	CA	GP	RA	SH	H3
SDE	2700	10822	5439	–	241215	3416
EA	430	–	460	2048	–	–
MLSL	206	–	148	–	–	197
IA	1354	326	–	–	7424	–
TUN	–	1469	–	–	12160	–
TS	492	–	486	540	727	508
TRUST	55	31	103	59	72	58
SPT	67	26	123	140	150	75

Table 1: Benchmark Results

performed. This problem has generally been formulated in terms of global optimization, and, to date, Monte-Carlo techniques [29], (e.g., simulated annealing, genetic algorithms) have provided the primary tools for seeking a potential solution. Such an approach is extremely expensive, and a major need has existed for better methods which would allow the accurate and efficient solution of large-scale problems.

The statics correction optimization problem can be summarized as follows. Acoustic signals are shot from N_s source locations and received by N_r sensors. Each signal reflects at a midpoint $k, k = 1, ..., N_k$. A trace t corresponds to seismic energy traveling from a source s_t to a receiver r_t via a midpoint k_t. We denote by D_{ft} the (complex) Fourier coefficient of frequency f $(f = 1, .., N_f)$ for trace t, $t = 1, ..., N_t \leq N_r N_s$. Ideally, after they have been corrected for normal moveout [27], all traces corresponding to the same midpoint carry coherent information . If there were no need for statics corrections, all signals, stacked by their common midpoint, should be in phase and yield a maximum for the total power

$$E(\mathbf{S}, \mathbf{R}) = \sum_k \sum_f | \sum_t \exp[2\pi i\, f\, (S_{s_t} + R_{r_t})] D_{ft} \delta_{kk_t}|^2 \ . \tag{1}$$

In (1), the statics corrections $\mathbf{S} = (S_1, .., S_{N_s})$, and $\mathbf{R} = (R, .., R_{N_r})$ are now considered independent variables. Their optimum values are found by maximizing the power E. This expression highlights the multimodal nature of E which, even for relatively low dimensional $\mathbf{S}$ and $\mathbf{R}$, has been found to exhibit a very large number of local minima.

A smaller problem in a 154-dimensional space has already been reported in the Barhen *Science* paper. To assess the performance improvements of the SPT algorithm, we considered a much larger problem involving $N_s = 100$ shots and $N_r = 216$ receivers yielding a 316-dimensional space. A data set consisting of $N_t = 4776$ synthetic seismic traces folded over $N_k = 423$ common midpoint gathers was obtained from CogniSeis

Corporation [30]. It uses $N_f = 118$ Fourier components for data representation. This set is somewhat typical of collections obtained during seismic surveys by the oil industry and is thus representative of the complexity underlying generic residual statics problems.

To derive a quantitative estimate of SPT's impact, the next two figures show the original subsurface map without statics corrections and the same map derived after solving the statics optimization problem. The SPT results illustrated in Figure 5 show significant improvement over the uncorrected results shown in Figure 4. These results also show considerable improvement over an attempt to solve this same problem with industry standard methods (see discussion in [31]). The optimization run used an adaptive method for estimating the pseudo-Lipschitz constant which resulted in a value of $L_{ps} = 50$. A one-dimensional grid of $N_{max} = 100,000$ was used to converge each individual one-dimensional ray. The approximate global minimum found by SPT was $f_{min} = -2442$ compared with an industry best result of $f_{min} = -2230$. The industry method required close to 40 hours of computer time to converge compared with less than one hour for SPT on comparable computing hardware. TRUST itself, was unable to solve the problem in its full dimensional entirety. A stack decomposition method similar to that used to solve the original *Science* paper statics 154-dimensional problem was needed to achieve a result of $f_{min} = -2320$.

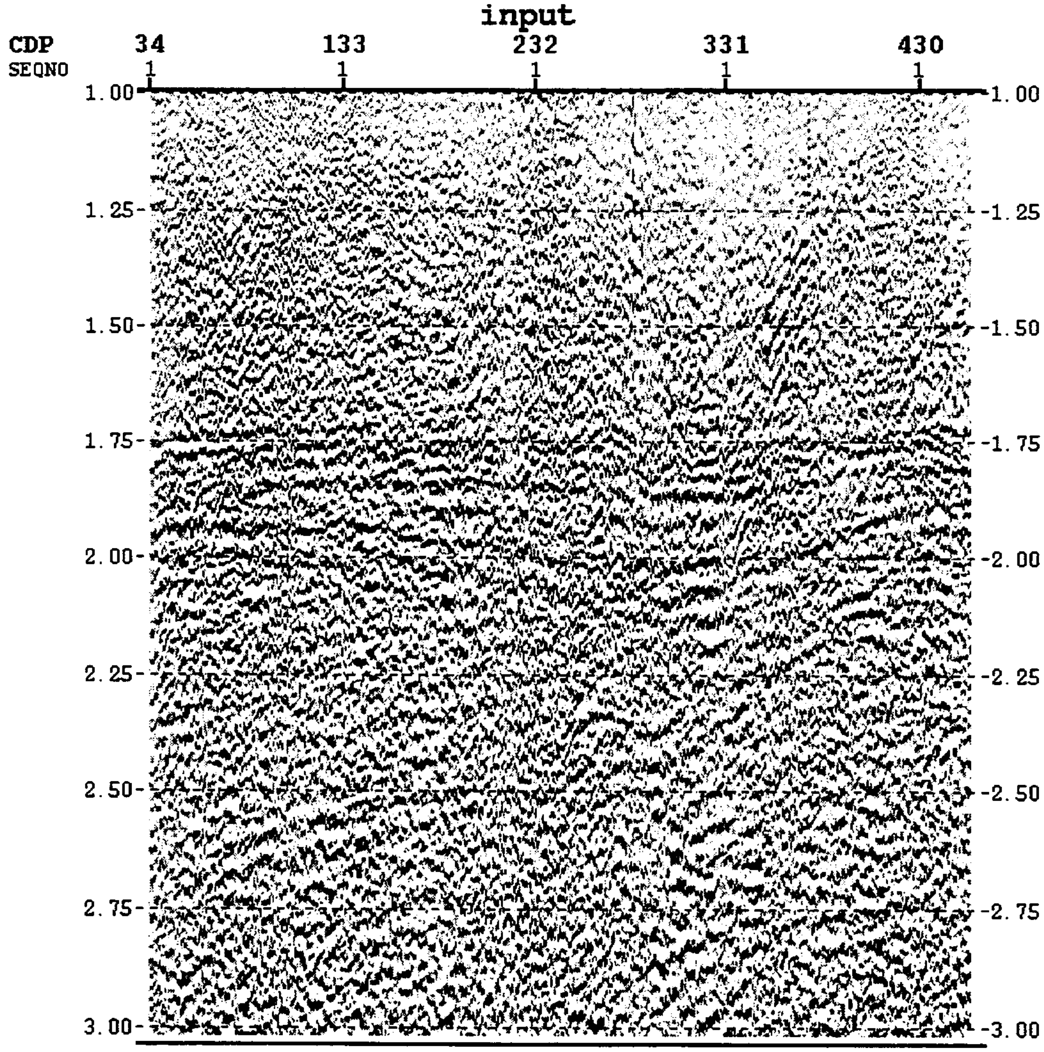

Figure 4: Unoptimized subsurface seismic map

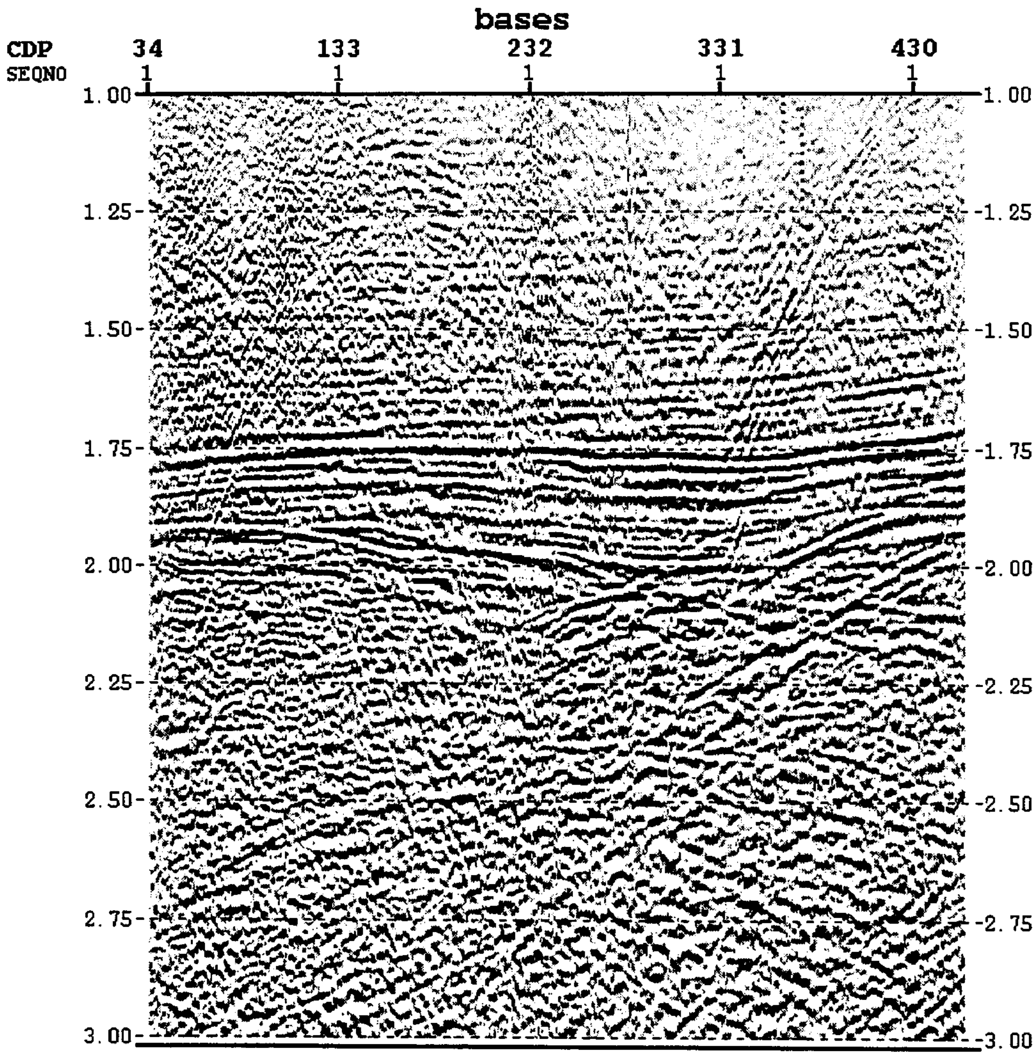

Figure 5: Optimized subsurface seismic map

10 Conclusions

From the results presented, one can conclude that a stochastic approach to tunneling makes a considerable improvement in Barhen's TRUST approach to solving continuous-function global optimization problems. This new approach builds on the deterministic tunneling-descent methodology presented by Barhen et al. [1,2], as one of the best benchmark methods to date, but uses a rejection-based stochastic procedure to locate new local minima descent regions. The method employs a series of local descents interspersed with stochastic search to find new local minima, using a rejection-based stochastic procedure to prune the tunneling search space traversed in searching for new local minima descent regions. It uses a fixed Lipschitz-like constant to reject unpromising regions in this search space thereby increasing the efficiency of the tunneling process. The algorithm is most easily implemented in low-dimensional problems, which allows it to be used efficiently as a heuristic in large scale problems.

Several offshoots of the basic algorithm should allow for further improvements in efficiency by allowing approximations to be made to estimating the algorithms parameters. The benchmark results presented, show that the algorithm is competitive with the best previously reported global optimization techniques. A successful application of the approach (using the one-dimensional approximation search scheme) to a large-scale seismology problem of substantial computational complexity bears out the practical applicability of the approach.

References

[1] J. Barhen and V. Protopopescu, "Generalized TRUST Algorithm for Global Optimization," *State of the Art in Global Optimization*, C.A. Floudas and P.M. Pardalos eds., pp. 163-180 (Kluwer, 1996).

[2] J. Barhen, V. Protopopescu, and D. Reister "TRUST: A Deterministic Algorithm for Global Optimization," ,,**276**, 1094-1097 (1997).

[3] A. Levy and A. Montalvo, "The Tunneling Algorithm for the Global Minimization of Functions," *SIAM Journal on Scientific and Statistical Computing*, **6**, 15 (1985).

[4] A. Törn and A. Zilinskas, *Global Optimization* (Springer-Verlag, 1989), pp. 66-67.

[5] R. Horst and H. Tuy, *Global Optimization*, 2nd ed. (Springer-Verlag, 1993).

[6] A. Törn and A. Zilinskas, *Global Optimization* (Springer-Verlag, 1989), pp. 30-38.

[7] A. Törn and A. Zilinskas, *Global Optimization* (Springer-Verlag, 1989), pp. 4-10.

[8] E. M. Oblow, "Implementing Valiant's Learnability Theory Using Random Sets," *Machine Learning*, **9**, 45-73 (1992).

[9] S. Pijavskij, "An Algorithm for Finding the Absolute Extremum of a Function," *USSR Computational Mathematics and Mathematical Physics*, **12**, 57-67 (1972).

[10] B. O. Shubert, "A Sequential Method Seeking the Global Extremum of a Function," *SIAM Journal of Numerical Analysis*, **9**, 379-388 (1972).

[11] Yu. G. Evtushenko, "Numerical Optimization Techniques," *Optimization Software Inc., New York*, p. 561 (1985).

[12] R. H. Mladineo, "An Algorithm for Finding the Global Maximum of a Multimodal, Multivariate Function," *Mathematical Programming*, **34**, 188-200 (1986).

[13] H. Ratschek, and J. Rokne, *New Computer Methods for Global Optimization* (Ellis Horwood, 1988).

[14] A. Törn and A. Zilinskas, *Global Optimization* (Springer-Verlag, 1989).

[15] R. Horst and H. Tuy, *Global Optimization*, 2nd ed. (Springer-Verlag, 1993).

[16] B. Cetin, J. Barhen, and J. Burdick, "Terminal Repeller Unconstrained Subenergy Tunneling (TRUST) for Fast Global Optimization," *Journal of Optimization Theory and Applications*, **77**, 97 (1993).

[17] R. Horst and P. M. Pardalos, *Handbook of Global Optimization* (Kluwer, 1995)

[18] R. Horst, P. M. Pardalos, and N. V. Thoai, *Introduction to Global Optimization* (Kluwer, 1996).

[19] C. A. Floudas and P. M. Pardalos, *State of the Art in Global Optimization: Computational Methods and Applications* (Kluwer, 1996).

[20] D. Cvijovic and J. Klinowski, "Taboo Search: An Approach to the Multiple Minima Problem," *Science*, **267**, 664 (1995).

[21] F. Aluffi-Pentini, V. Parisi, and F. Zirilli, "Global Optimization and Stochastic Differential Equations," *Journal of Optimization Theory and Applications*, **47**, 1 (1985).

[22] L. Yong, K. Lishan, and D.J. Evans, "The Annealing Evolution Algorithm as Function Optimizer," *Parallel Computing*, **21**, 389 (1995).

[23] G. Schneider, J. Schuchhardt, and P. Wrede, "Evolutionary Optimization in Multimodal Search Space," *Biological Cybernetics*, **74**, 203 (1996).

[24] A. H. G. R. Kan and G. T. Timmer, "A Stochastic Approach to Global Optimization," in *Numerical Optimization*, eds. P. T. Boggs, R. H. Byrd, and R. B. Schnabel, pp. 245–262 (SIAM, 1985).

[25] H. Ratschek, and J. Rokne, *New Computer Methods for Global Optimization* (Ellis Horwood, 1988).

[26] D. Rothman, "Nonlinear Inversion, Statistical Mechanics and Residual Statics Estimation," *Geophysics*, **50**, 2784 (1985).

[27] O. Yilmaz, *Seismic Data Processing*, Society of Exploration Geophysicists (Tulsa, 1988).

[28] J. DuBose, "Practical Steps Toward Realizing the Potential of Monte Carlo Automatic Statics," *Geophysics*, **58**, 399 (1993).

[29] M. Sen and P. L. Stoffa, *Global Optimization Methods in Geophysical Inversion* (Elsevier, 1995).

[30] The authors thank J. DuBose for providing the benchmark problem.

[31] J. Barhen, D. Reister, E. Oblow, and J. DuBose, ORNL/TM in preparation.

INTERNAL DISTRIBUTION

1. J. Barhen
2. S. G.Batsell
3. T. Dunigan
4. H. R. Hicks
5. M. A. Kuliasha

6.-25. E. M. Oblow

26. C. E. Oliver
27. L. E. Parker
28. V. Protopopescu
29. N. S. V. Rao
30. D. B. Reister
31. T. Rowan
32. B. A. Worley

33.-34. T. Zacharia

35. Central Research Library
36. Document Reference Section
37. Laboratory Records - RC
38. ORNL Patent Office

EXTERNAL DISTRIBUTION

39-40. Office of Scientific and Technical Information, P.O. Box 62, Oak Ridge, TN 37831.

41. Prof. I. Akyildiz, School of Electrical Engineering, Georgia Institute of Technology, Atlanta, GA 30332
42. Prof. J. Dongarra, Department of Computer Science, 107 Ayres Hall, University of Tennessee, Knoxville, TN 37996-1301
43. Dr. D. Hitchcock, SC-31, U.S. Department of Energy, 19901 Germantown Road, Germantown, MD 20874-1290
44. Dr. F. Howes, SC-31, U.S. Department of Energy, 19901 Germantown Road, Germantown, MD 20874-1290
45. Prof. S. S. Iyengar, Department of Computer Science, Louisiana State University, Baton Rouge, LA 70803
46. Prof. K. Maly, Department of Computer Science, Old Dominion University, Norfolk, VA 23529
47. Prof. N. Manickam, Department of Mathematics, Depauw University, Greencastle, IN 46135
48. Dr. O. Manley, 13212 Wye Oak Dr. Gaithersburg, MD 20878
49. Dr. D. B. Nelson, SC-30, U.S. Department of Energy, 19901 Germantown Road, Germantown, MD 20874-1290
50. Dr. R. Price, SC-15, U. S. Department of Energy Basic Energy Sciences 19901 Germantown Road, Germantown, MD 20874-1290
51. Prof. R. C. Ward, Department of Computer Science, 107 Ayres Hall, University of Tennessee, Knoxville, TN 37996-1301