Proceedings of the 1999 Winter Simulation Conference
P. A. Farrington, H. B. Nembhard, D. T. Sturrock, and G. W. Evans, eds.

POLYNOMIAL ACCELERATION OF MONTE-CARLO GLOBAL SEARCH

James M. Calvin

Department of Computer and
Information Science
New Jersey Institute of Technology
Newark,NJ 07102, U.S.A.

ABSTRACT

In this paper we describe a class of algorithms for approximating the global minimum of a function defined on a subset of d-dimensional Euclidean space. The algorithms are based on adaptively composing a number of simple Monte Carlo searches, and use a memory of a fixed finite number of observations. Within the class of algorithms it is possible to obtain arbitrary polynomial speedup in the asymptotic convergence rate compared with simple Monte Carlo.

1 INTRODUCTION

Let f be a real-valued function defined on a compact set $A \subset \mathcal{R}^d$. We are interested in approximating the global minimum of f over A based on observation of the function value at sequentially selected points, while maintaining a memory of a fixed finite number of function values. In this paper we construct a class of randomized algorithms with the following property: For a broad class of objective functions, and for any integer k, there exists an algorithm for which the probability that the error after n observations exceeds n^{-k} converges to 0. Thus the convergence rate can be made better than any polynomial in the reciprocal of the number of observations.

The algorithms are based on Monte Carlo, or independent uniform sampling over the domain. Simple Monte Carlo global search has several attractive properties as a method for approximating the global minimum of a complicated function. For many objective functions the error, when suitably normalized, converges in distribution. One can use this fact to construct confidence intervals for the minimum based on a sample of independent observations. The convergence rate depends on the function, but is typically a small power of the reciprocal of the number of observations. Another advantage, more important for this paper, is that the limiting distribution of points near the global minimizer can be precisely characterized.

The algorithms described in this paper are generalizations of an algorithm described in Calvin (1997). That algorithm was only for one-dimensional objective functions, and the analysis was only carried out in the setting of a random objective function.

Let $f^* = \min_{t \in A} f(t)$ denote the global minimum of the function. We assume that f attains its global minimum at a unique point t^* in the interior of A. The object of a global minimization method is to approximate the global minimum f^*, and sometimes also the location t^*. We adopt the framework that the approximation is based on observation of the function value at sequentially selected points. That is, the searcher chooses points $t_1, t_2, \ldots \in A$ and forms an approximation (t_n^*, f_n^*) to (t^*, f^*) based on $\{t_i, f(t_i) : i = 1, 2, \ldots, n\}$. A general adaptive algorithm in this setting will choose the $(n+1)$st point t_{n+1} as a function of all the previous observations and some auxiliary randomization; i.e.,

$$t_{n+1} = h_{n+1}(t_1, f(t_1), t_2, f(t_2), \ldots, t_n, f(t_n), Z_n)$$

for some function h_{n+1} and random variable Z_n.

In this paper we are concerned with the case of bounded memory. If a total of M observation pairs are allowed, then to keep a new observation an old one must be discarded (after M are stored), so an algorithm takes the form

$$t_{n+1} = h_{n+1}\left(t_{i_1}, f(t_{i_1}), t_{i_2}, f(t_{i_2}), \ldots, t_{i_M}, f(t_{i_M}), Z_n\right).$$

The questions include how to choose t_{n+1} given the past history, and what information to keep and what to discard. Our ultimate aim is to determine how fast the global minimum can be approximated.

Let

$$\Delta_n = \min_{i \le n} f(t_i) - f(t^*)$$

denote the error after n observations. We are mainly interested in the convergence rate of Δ_n to 0 under various algorithms; that is, we undertake an asymptotic analysis. The obtainable convergence rates depend on the characteristics of the objective function f, as well as on the cardinality of information M.

In the next section we describe the basic assumption that we make on the objective functions and the basic facts about Monte Carlo random search. In Section 3 we describe the adaptive extension. Section 4 presents the results of numerical simulations of the algorithm.

2 SIMPLE MONTE CARLO SEARCH

In this section we review the basic facts about simple Monte Carlo search; a detailed treatment is given in Zhigljavsky (1991). The behavior of the error variables is relatively well understood in the case of uniform independent sampling, and we will exploit that fact in this study.

For $T > 0$ let $B_T = \{x \in \mathcal{R}^d : \|x\| \le T\}$ be the closed ball of radius T in $\mathcal{R}^d$. By rescaling and extending the function if necessary we can take the domain $A = B_1$. Denote the Borel σ-field on B_T by $\mathcal{B}_T$.

Let $\{U_i : i \ge 1\}$ be a sequence of independent random variables, all uniformly distributed on B_1. Let U_n^* be that observation point of the first n with the smallest function value: i.e., $U_n^* = U_j$ for some $1 \le j \le n$ and $f(U_n^*) \le f(U_i), i \le n$, with ties broken arbitrarily. By "simple Monte Carlo" we mean approximating $f(t^*)$ by $f(U_n^*)$ after n function evaluations.

We now describe the basic assumption we make on the objective function and its consequences in the context of independent, uniform sampling. Assume that f is Borel measurable and that for any $\delta \in (0, 1)$,

$$n^{(1-\delta)/d} \|U_n^* - t^*\| \xrightarrow{P} 0 \tag{1}$$

as $n \to \infty$. Here $\xrightarrow{P}$ denotes convergence in probability; that is, for random variables $X_n, n \ge 1$ and X, $X_n \xrightarrow{P} X$ if $P(\|X_n - X\| > \epsilon) \to 0$ as $n \to \infty$ for any $\epsilon > 0$. This assumption is satisfied, for example, if the objective function is continuous and differentiable on a neighborhood of t^* (recall the assumption that the minimizer is unique).

Let $b_1 = |B_1|$ denote the Lebesgue measure of the unit ball in $\mathcal{R}^d$. For $T > 0$, define a sequence of point processes N_n^T on B_T by

$$N_n^T(A) = \sum_{k=1}^{n} I_{\{(n/b_1)^{1/d}(U_k - t^*) \in A\}}, \quad A \in \mathcal{B}_T,$$

where I_A is the indicator function of the set A. Therefore, $N_n^T(A)$ is the number of observations of the first n that land in the set $t^* + (n/b_1)^{-1/d} A$. It is well known (see Kallenberg 1976) that for any $T > 0$, N_n^T converges in distribution to a Poisson point process with unit intensity on B_T as $n \to \infty$; we denote this by $N_n^T \xrightarrow{\mathcal{D}} N^T$, where N^T is a Poisson process on B_T with intensity one. This means that for disjoint $A_1, A_2, \ldots, A_k \in \mathcal{B}_T$, the $N^T(A_i)$, $1 \le i \le k$ are independent random variables, and $N^T(A_i)$ has a Poisson distribution with mean $|A_i|$:

$$P(N^T(A_i) = k) = \frac{|A_i|^k}{k!} \exp(-|A_i|), \quad k \ge 0.$$

Thus, under uniform sampling, the point process of observations near the minimizer t^* (normalized by multiplying the distance from t^* by $(n/b_1)^{1/d}$) looks like a Poisson point process.

How can we exploit the results of the uniform sampling search? Since $U_n^* \xrightarrow{P} t^*$, U_n^* gives an approximation to t^*, which we use to guide a second search which progresses in parallel with the first. On the nth iteration, the second search chooses a point uniformly on a ball of radius $n^{-(1-\delta)/d}$ centered at U_n^*. Since $n^{(1-\delta)/d}\|U_n^* - t^*\| \xrightarrow{P} 0$, the radius tends to be large compared with the distance between U_n^* and t^*, and so the ball is eventually likely to contain t^*. Therefore, the points are raining down uniformly (at an intensity that varies with n) around t^*. It turns out that under appropriate scaling the points of the second search near t^* converge to a Poisson point process. From this we obtain an approximation to t^* from the second search that is used to guide a third search, in an analogous way to the first search guiding the second. We continue on for an arbitrary number of searches. In the next section we introduce the notation needed to describe precisely the extension to M parallel searches.

3 ADAPTIVE MONTE CARLO

The adaptive algorithm is most easily explained in terms of an implementation on $M \ge 2$ processors (M corresponds to the cardinality of information alluded to in the Introduction). The processors communicate in a way that will be described below.

Let $U_{i:j}, 1 \le i \le M, 1 \le j$ be an array of independent random variables, uniformly distributed over B_1. We will define an array of observation points $t_{i:j}, 1 \le i \le M, 1 \le j$, where $t_{i:1}, t_{i:2}, \ldots$ is the sequence of observations made by processor i, $1 \le i \le M$, which will be randomized by the sequence $U_{i:1}, U_{i:2}, \ldots$.

We begin with a discussion of the situation with $M = 2$, since it forms the basis of what is to follow.

Processor 1 makes the jth search at $t_{1:j} = U_{1:j}$ independent and uniformly distributed over B_1. Denote by $t^*_{1:n}$ the best location of the first n; that is, $t^*_{1:n} = t_{1:j}$ for some $j \le n$ and $f(t^*_{1:n}) \le f(t_{1:i})$, $i \le n$. (To connect to the notation of the previous section, $t_{1:i}$ corresponds to U_i and $t^*_{1:i}$ corresponds to U^*_i.)

Processor 2 searches uniformly over a ball of radius $n^{-(1-\delta)/d}$ centered at the best location reported by processor 1; i.e.,

$$t_{2:n} = t^*_{1:n} + n^{-(1-\delta)/d} U_{2:n}, \qquad n \ge 1.$$

Denote by $t^*_{2:n}$ the best location of the first n seen by the second processor, and set $c_{1:n} = (n/b_1)^{1/d}$ and

$$\begin{aligned} c_{2:n} &= \left(\frac{1}{b_1} \sum_{k=1}^{n} k^{(1-\delta)} \right)^{1/d} \\ &= \left(\frac{1}{b_1(2-\delta)} \right)^{1/d} n^{(2-\delta)/d} + O(n^{1/d}), \end{aligned}$$

and so $n^{-1/d} c_{2:n} \to \infty$; i.e., $c_{1:n} = o(c_{2:n})$.

For any $T > 0$, define the sequence of point processes

$$N^T_{2:n}(A) = \sum_{k=1}^{n} I_{\{c_{2:n}(t_{2:k} - t^*) \in A\}}, \qquad A \in \mathcal{B}_T.$$

Let N be a Poisson process with unit intensity on $\mathcal{R}^d$, and for any $T > 0$, let N^T be the restriction of N to B_T. The key to the acceleration procedure is the fact that for any $T > 0$, as $n \to \infty$,

$$N^T_{2:n} \xrightarrow{\mathcal{D}} N^T.$$

This shows that the Poisson nature of the point processes is preserved under the concentration scheme we have described.

We have now established the results needed for the two processor case. Notice that the only interprocess communication is the transmission of $t^*_{1:n}$ from processor 1 to processor 2.

In order to extend the results to three or more processors, we need to determine how closely $t^*_{2:n}$ approximates t^*. So far we have only used the fact that $k^{(1-\delta)/d} \|t^*_{1:k} - t^*\| \xrightarrow{P} 0$. The extension to three processors uses the fact that $c^{(1-\delta)/d}_{2:k} \|t^*_{2:k} - t^*\| \xrightarrow{P} 0$. The situation as seen locally at t^* is essentially as if $c^d_{2:n}$ observations, instead of n, had been made uniformly over the entire interval. Since $c^d_{2:n}/n \to \infty$, there is a speedup.

The remaining processors follow the same pattern, each searching a ball of decreasing radius centered at the best location reported by its predecessor. Processor 3 searches at

$$t_{3:n} = t^*_{2:n} + c^{-(1-\delta)}_{2:n} U_{3:n}.$$

Denote by $t^*_{3:n}$ the best location of the first n and set

$$c_{3:n} = \left(\frac{1}{b_1} \sum_{k=1}^{n} c^{d(1-\delta)}_{2:k} \right)^{1/d}.$$

For any $T > 0$, define

$$N^T_{3:n}(A) = \sum_{k=1}^{n} I_{\{c_{3:n}(t_{3:k} - t^*) \in A\}}, \qquad A \in \mathcal{B}_T,$$

and note that

$$c^{(1-\delta)/d}_{3:n} \|t^*_{3:n} - t^*\| \xrightarrow{P} 0.$$

The remaining processors follow a similar pattern.

In summary, processor 1 searches uniformly over the unit ball, keeping track of the best location, which it communicates to processor 2. Processor 2 makes the nth search uniform over a sub-ball of radius $n^{-(1-\delta)/d}$ centered at the best location transmitted from processor 1, and keeps track of its best location, transmitting it on to processor 3, and so on. At the end of the chain, processor M makes its nth search uniform over a sub-ball of radius $c^{-(1-\delta)}_{M-1:n}$ centered at the best location transmitted from processor $M-1$. The radii are chosen to approach 0 at such a rate that the distance from the center of a sub-ball to t^* is asymptotically negligible compared to the radius of the sub-ball. Each processor stores only one location, and the computational cost grows linearly with the number of iterations.

We emphasize that each processor centers its search on the best location observed by its immediate predecessor in the chain, not the minimum of all its predecessors.

In the two-dimensional ($d = 2$) case, the sampling density of the algorithm (all processors taken together) can be pictured as the side view of a wedding cake with M layers. Thus the density is highest in the smallest central section. Note, however, that the sections need not be concentric, or even overlap (though they typically would).

We are now ready to state our main result, which basically says that the normalized point process of observations corresponding to processor j converges to a Poisson process with unit intensity for each $1 \le j \le M$.

Theorem 1 *Fix an integer $M > 1$, a positive number T, and $\delta \in (0, 1)$. Define*

$$c_{1:n} = (n/b_1)^{1/d}, \qquad t_{1:n} = U_{1:n},$$

and for $1 < j \le M$ *set*

$$c_{j:n} = \left(\frac{1}{b_1} \sum_{k=1}^{n} c_{j-1:k}^{d(1-\delta)} \right)^{1/d},$$

$$t_{j:n} = t^*_{j-1:n} + c_{j-1:n}^{-(1-\delta)} U_{j:n},$$

and for $1 \le j \le M$ *define*

$$N^T_{j:n}(A) = \sum_{k=1}^{n} I_{\{c_{j:n}(t_{j:k} - t^*) \in A\}}, \quad A \in \mathcal{B}_T.$$

Then for each $j \le M$,

$$N^T_{j:n} \xrightarrow{\mathcal{D}} N^T$$

as $n \to \infty$, *where* N^T *is a Poisson point process, with intensity one, on* B_T.

For the proof of the theorem (under a more general set of assumptions) see Calvin (1999).

Denote the superposition of all M point processes (with the Mth scaling) by

$$N^T_n(A) = \sum_{j=1}^{M} \sum_{k=1}^{n} I_{\{c_{M:n}(t_{j:k} - t^*) \in A\}}, \quad A \in \mathcal{B}_T.$$

Since $c_{M:n}$ is the dominant rate ($c_{M-j:n} = o(c_{M:n})$ as $n \to \infty$ for $j > 1$), N^T_n has the same limit distribution as N^T_M. In other words, the results of processors 1 through $M-1$ serve only to guide the searches of the higher-numbered processors, and their approximations to the global minimum are insignificant in the limit.

This is our main result. Under the basic assumption (1), with memory of cardinality M we can make the point process of observations near t^* after n observations look as if we had made $c^d_{M:n}$ observations instead uniformly over B_1.

Theorem 1 holds under the sole assumption that

$$n^{(1-\delta)/d} \| U^*_n - t^* \| \xrightarrow{P} 0$$

as $n \to \infty$. To say more about the error we must know more about the objective function f. For an example, assume that f is twice continuously differentiable at t^*, with nonsingular matrix of second partial derivatives D. Then

$$P\left(n^{2/d} \frac{\Delta_n}{(\det D)^{1/d}} > y \right) \to \exp(-y^{d/2}), \quad y > 0;$$

see de Haan (1981). Then

$$\begin{aligned} c^d_{j:n} &= \left(\frac{n}{b_1}\right)^{[1-(1-\delta)^j]/\delta} \left(\frac{1}{\prod_{i=0}^{j-2} \left([1-(1-\delta)^j]/\delta \right)} + O(1/n) \right) \\ &= a_j n^{j - O(\delta)}, \end{aligned}$$

where a_j is a constant independent of n and $O(\delta) \to 0$ as $\delta \to 0$. The constants $a_j \to 0$ as $j \to \infty$. Thus with M processors, we can obtain an effective intensity of n^{M-1} instead of n, for example. In this case the claim made in the Introduction is borne out; for any $k > 0$, we can define an algorithm (choose M and δ), such that the probability that the error exceeds n^{-k} converges to 0.

4 NUMERICAL EXPERIMENTS

In this section we describe the results of some numerical experiments. The purpose of the experiments is to gain insight into how well the theoretical limit distribution approximates the empirical distribution of the error after a moderate number of iterations.

For the tests we use a standard test function called *Rosenbrock's saddle*. It is defined by

$$F(x_1, \ldots, x_d) = \sum_{i=1}^{d-1} \left((1 - x_i)^2 + 100(x_{i+1} - x_i^2)^2 \right)$$

for $-2.048 \le x_i \le 2.048$, $1 \le i \le d$; see Moré et all (1981). This function has a long curved valley which is only slightly decreasing towards the global minimum of 0 at $x_i = 1$ for $1 \le i \le d$. We modified the algorithm so that instead of a unit ball, a ball large enough to include the specified set was used by the first processor.

Figure 1 plots the empirical distribution functions for the error and the theoretical cumulative distribution function for the Rosenbrock function with $d = 4$ and 1,000 independent replications. Three different experiments were performed, each with $M = 2$ processors, and 10, 50, and 100 thousand observations per processor, respectively. As can be seen, the empirical distributions have fat tails, and approach the theoretical CDF $1 - \exp(-x^2)$ as the number of observations increases.

The explanation for the way the empirical CDFs change with the number of iterations is as follows. If the first processor happens to have made an observation near the global minimizer, then the second processor will be concentrating its effort near the minimizer and the limit distribution will be well-approximated. On the other hand, if the first processor has *not* placed an observation near the global minimizer (an event that has probability converging to 0), then the

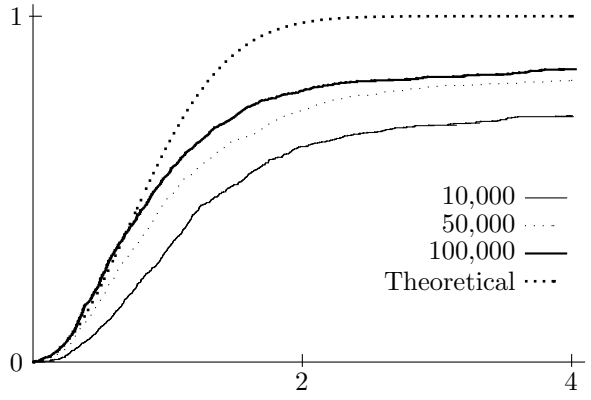

Figure 1: Comparison of Empirical CDFs

second processor will be wasting its effort away from the minimizer. In short, there is a vanishing probability that the best observation point is far from t^*, but if it is far away, then it is likely to be very far away.

ACKNOWLEDGMENTS

This work was supported in part by the National Science Foundation under grant DMI-9500173.

REFERENCES

Calvin, J. M. 1997. Average performance of a class of adaptive algorithms for global optimization. *Annals of Applied Probability* 7: 711–730.

Calvin, J. M. 1999. Adaptive Monte Carlo global search with bounded memory. New Jersey Institute of Technology, Computer and Information Science Report No. 99-5.

de Haan, L. 1981. Estimation of the minimum of a function using order statistics. *Journal of the American Statistical Association* **76** (374): 467–469.

Kallenberg, O. 1976. *Random Measures*. Berlin: Akademie-Verlag.

Moré, J., Garbow, B., and Hillstrom, K. 1981. Testing unconstrained optimization software. *ACM Transactions on Mathematical Software* **7**: 17–41.

Zhigljavsky, A. 1991. *Theory of Global Random Search.* Dordrecht: Kluwer.

AUTHOR BIOGRAPHY

JAMES M. CALVIN is an assistant professor in the Department of Computer and Information Science at the New Jersey Institute of Technology. He received the Ph.D. in Operations Research from Stanford University. His research interests include global optimization and simulation output analysis.