ORSJ Publication Series T-94-1

New Directions in SIMULATION for Manufacturing and Communications

Edited by

MORITO Susumu
Waseda University

SAKASEGAWA Hirotaka
Waseda University

YONEDA Kiyoshi
Toshiba R&D Center

FUSHIMI Masanori
University of Tokyo

NAKANO Kazuo
Kozo Keikaku Engineering

Operations Research Society of Japan

1994

New Directions in Simulation for Manufacturing and Communications
(eds. Morito S., Sakasegawa H., Yoneda K., Fushimi M., and Nakano K.), 1994

RANDOM OPTIMIZATION METHODS

Luc Devroye
School of Computer Science
McGill University
luc@crodo.cs.mcgill.ca
FAX: +1–514-398-3883

Abstract.

Pure random search, adaptive random search, simulated annealing, genetic algorithms, evolution strategies, nonparametric estimation methods, bandit problems, simulation optimization, clustering methods, probabilistic automata and random restart. These are a sample of the random search methods that are reviewed in this paper. The discussion focuses on computational issues as well as behavior in difficult optimization problems.

Books.

This survey starts with the mention of a few good books on the subject matter. These include Zhigljavsky (1991), Törn and Žilinskas (1989), Aarts and Korst (1989), Van Laarhoven and Aarts (1987), Holland (1975), Schwefel (1981), Schwefel and Männer (1991), Ackley (1987), Goldberg (1989), Ermoliev and Wets (1988), and Wasan (1969).

Advantages of random search.

Below are some reasons why you may wish to look at random search algorithms more closely.

A. Ease of programming. Simple easily understood programs, that can be implemented on nearly any computer.

B. Inexpensive realization. Many of the methods require very simple storage and comparison facilities. There is virtually no overhead, so that the cost of a run is virtually borne only by the number of function evaluations.

C. Insensitivity to the criterion function. Convergence of most random search procedures can be guaranteed for any function, regardless of its smoothness properties or its multimodality. The particular shape (granularity, discontinuity, presence of holes or plateaus) of a function has virtually no effect on most random search procedures.

D. Efficiency. In deterministic schemes, a lot of computer time is spent in deciding where the next probe point should be. In random search procedures, this time is saved, at the expense perhaps of a few more probes. According to the minimax criterion, random search is more efficient than any deterministic search. This says that random search is best in the worst possible circumstances. Jarvis (1975) in his survey claims that "it is one of the best methods in the worst situation possible (granularity, plateaus, holes, discontinuities, high dimensionality, multimodality) and perhaps the worst method in the best situation (smoothness, continuity, low dimensionality, unimodality)".

E. Flexibility. Random methods fill the entire gap between pure random search (which totally ignores any previously obtained information) and deterministic methods. In fact, many are geared towards efficient combinations of methods.

F. Information extraction. During the optimization process, the information gathered can be used to guide the search; this is especially useful when global information about the shape of the function has to be extracted.

G. Easily parallelizable. Many random search procedures either totally ignore past information, or proceed with a number of simultaneous searches or moving clouds of points, with only an occasional need for communication between the various components. This lends itself superbly to parallelization.

H. Insensitivity to noise. Function evaluations that are perturbed by noise affect the performance of random search algorithms much less than that of deterministic algorithms. Random search is also ideally suited for multimodal stochastic optimization problems.

I. A simple startpoint selection method. Pure random search and some of its variants can be used as a method for the selection of a suitable startpoint of a local search algorithm. It is

still nearly the only way to select startpoints.

J. A standard for comparisons. In tests and simulations, pure random search can be (and is being) used as a simple benchmark against which we can gauge the goodness of other algorithms. This is especially so since pure random search provides us with enough information to decide how "difficult" the optimization problem is.

Why are we doing random search?

Of course, it is in the human nature to optimize, so optimization is here to stay. But we also want to explore new terrain. And to explore in an unbiased manner, nothing really beats random sampling—look for example at the continued success of (random) opinion polls. Randomness is an ideal and unequaled information gathering device.

Passive versus active algorithms.

In some settings, we have no control over the probe points—they are part of the data given to us. The latter context leads to so-called passive algorithms (Härdle and Nussbaum, 1993): the prototype problem dealt with is the simultaneous estimation and minimization of a regression function.

Issues.

In optimization, it is important to know whether we have a discrete or continuous parameter space, whether we are dealing with a unimodal or multimodal function, and whether we can make use of parallel processors or not. In some contexts it is out of the question to consider parameters at which the function has not been evaluated, and in others, this does not matter. More modern algorithms tend to make better use of storage and memory as they become cheaper and faster.

Noisy problems.

Here is a rather general optimization problem: for each parameter x (living in some space $\mathcal{X}$), we can observe a random process $Y_1, \ldots, Y_n, \ldots$ with $Y_n \to Q(x)$ almost surely, where Q is the function to be minimized. I will refer to this as the noisy optimization problem. Examples:

A. Engineering noise: at x, we can observe independent copies of $Q(x) + \xi$, where ξ is measurement noise satisfying $\mathbf{E}\xi = 0$ and $\mathbf{E}|\xi| < \infty$. This classical setting is no longer important. Averaging may lead to a sequence Y_n with the given convergence property.

B. In simulation optimization, Y_n may represent a simulation run for a system parametrized by x. It is necessary to take n large for accuracy, but taking n too large would be wasteful for optimization. Beautiful compromises are awaiting the analyst.

C. In some cases, $Q(x)$ is known to be the expected value or an integral, as in $Q(x) = \int_A q(x,t)\,dt$ or $Q(x) = \mathbf{E}q(x,T)$ where A is a fixed set and T is a given random variable. In both cases, Y_n may represent a certain Monte Carlo estimate of $Q(x)$, which may be made as accurate as desired by taking n large enough.

Ordinary random search.

The simple ordinary random search algorithm is given below:

$$X_{n+1}^* = \begin{cases} X_{n+1} & \text{if } Q(X_{n+1}) < Q(X_n^*) \\ X_n^* & \text{otherwise.} \end{cases}$$

Here X_n^* is the best estimate of the (global) minimum after n iterations, and X_{n+1} is a random probe point. In pure random search, $X_1, \ldots, X_n$ are i.i.d. with a given fixed probability measure over the parameter space $\mathcal{X}$. In local random search of a discrete space, X_{n+1} usually is a random neighbor of X_n^*, where the definition of a neighborhood depends upon the application. In local random search in a Euclidean space, one might set

$$X_{n+1} = X_n^* + W_n \ ,$$

where W_n is a random perturbation usually centered at zero.

Pure random search.

The properties of pure random search are well documented in nearly all books on the subject since its early introduction by Brooks (1958). The fundamental properties of the method are related to the fact that $F(Q(X_n^*))$ is approximately distributed as E/n, where E is an exponential random variable, and F is the distribution function of $Q(X_1)$: $F(u) \stackrel{\text{def}}{=} \mathbf{P}\{Q(X_1) \le u\}$. This follows from the fact that if F is nonatomic,

$$\mathbf{P}\{F(Q(X_n^*)) > t/n\} = (1-t/n)^n \to e^{-t} \ , \ t > 0 \ .$$

Note first of all the distribution-free character of this statement: its universality is both appealing and limiting. We note in passing here that many papers have been written about how one could decide to stop random search at a certain point. In the case of pure random search, this is nearly always futile. For example, assume that we stop when no improvement of X_n^* is found

in the last 100 iterations. Unfortunately, the value of $Q(X_N^*)$ when we stop at such a time N is independent of N, so information about improvements of X_n^* is useless for stopping rules. As a curiosity related to the theory of records, note that $\mathbf{E}N = \infty$ if we were to stop at time N, where N is the first time we find an improvement on X_1.

Stratified random search.

Brooks (1959) looked at stratified random search, in which we partition $\mathcal{X}$ beforehand into a finite number of sets $A_1, \ldots, A_k$, and generate n/k probe points at random in each of these sets according to a fixed distribution. If the partitions are such that the X_1 of pure random search would fall equally likely in each subset, then one can prove that $Q(X_n^*)$ is stochastically smaller in stratified pure random search than in pure random search. This is not necessarily so if the probabilities of the partitions are unequal.

Minimax theory.

In the hope of pinning down the ultimate algorithm, we may resort to a minimax strategy. Let T be the time needed to reach the global minimum of a function $Q : \{1, \ldots, N\} \to \mathbb{R}$. A pessimist might consider as a measure of the performance of an algorithm the quantity

$$\sup_Q \mathbf{E}\{T\} .$$

It turns out that for any algorithm, the latter quantity is at leat $(N+1)/2$. Consider just three strategies:

A. Pure random search with a uniform distribution. Here $\mathbf{E}\{T\} = N$.

B. Cyclic (brute-force) search. Here too, we have $\mathbf{E}\{T\} = N$.

C. Random permutation search. A simple exercise shows that this is optimal in the sense stated above, as $\mathbf{E}\{T\} = (N+1)/2$.

We might also look at

$$\sup_Q \mathbf{P}\{Q(X_n^*) > Q_m\}$$

where Q_m is the m-th smallest value among $Q(1), \ldots, Q(N)$. Here too, random permutation search is optimal, as the given supremum is $\binom{N-n}{m}/\binom{N}{m}$ when $m + n \le N$.

If we move to functions $Q : \mathbb{R}^d \to \mathbb{R}$, and let F be the distribution function of $Q(X)$, where X is as before, then

$$\inf_{\text{all algorithms}} \sup_{Q:\mathbf{R}^d \to \mathbf{R}} \mathbf{E}\{F(Q(X_n^*))\} = \frac{1}{n+1}$$

and equality is reached for pure random search. Both examples given above describe the optimality of some version of global random search in the toughest of situations, i.e., when no additional knowledge is available about Q except the space of its argument. Let us briefly consider a more restricted class of functions. We are quick to add that most function classes immediately lead to challenging analytical problems. The one that has received some attention in the literature is the class of functions Q that are Lipschitz with a constant C on $[0,1]^d$. We assume that C is known. For mathematical convenience, we suppose that Q is periodically continued and continuous by replicas of the unit hypercube. Also, the minimum of Q is exactly zero. If the X_i's are the probe points, and X_n^* is the best probe point seen thus far, we may measure the performance of an algorithm by

$$P_n \stackrel{\text{def}}{=} n^{1/d} \sup_Q \mathbf{E}\{Q(X_n^*)\} ,$$

where the coefficient is added for easy normalization. We may cover $[0,1]^d$ with circles of radius r at the points at which we have probed Q. If r is the smallest such radius, then we know that $\min Q \ge \min Q(X_i) - Cr$. Thus, P_n may be found by looking at the largest gap between the points left by a certain algorithm. To make the largest gap small, we need to sprinkle the points out very densely but evenly. Here are some strategies.

A. If we place the points in a rectangular grid, then $P_n \to C\sqrt{d/4}$.

B. It is known that rectangular grids do not pack points very well. For example, Voronoi's principal lattice leads to $\lim_{n\to\infty} P_n = C\gamma_d$, where, as $d \to \infty$, $\gamma_d \sim \sqrt{d/12}$. This is the thinnest cover for $d = 2$, the thinnest lattice cover for $d \le 5$, and the thinnest known cover to date for $d \le 23$. This is a remarkable improvement. For a wealth of nice lattices, see Conway and Sloane (1988).

C. Coxeter, Few and Rogers (1959) have results on packing and covering that imply that no matter how we position the probe points,

$$\liminf_{n\to\infty} P_n \ge C\sqrt{(d+o(1))/(2\pi e)} .$$

In 1959, Rogers showed in a nonconstructive way that there is a grid that would lead to

$$\limsup_{n\to\infty} P_n \le C\sqrt{(d+o(1))/(2\pi e)} .$$

Interestingly, the optimal lattice covering is not known except for small d, so this avenue is of no help when one is faced with $d = 50$, say.

D. In pure random search, we have $\lim_{n\to\infty} P_n = C\Gamma(1+1/d)/V_d^{1/d}$, where V_d is the volume of the unit sphere in $\mathbb{R}^d$. The limit may be rewritten as $C\sqrt{(d+o(1))/(2\pi e)}$ and is therefore optimal for large d. Pure random search handily outperforms Voronoi lattice search despite its striking simplicity. Although there are deterministic algorithms that will do as well, no explicit deterministic algorithm is known to date that would beat pure random search in this respect.

Random covering methods.

Lipschitz functions may be dealt with by covering methods. Consider for example the method of Shubert (1972) in one dimension. At the trial points, linear bounds on Q are pinned down, and the next trial point is given by

$$X_{n+1} = \arg\min_{x\in\mathcal{X}} \max_{i\le n} \{Q(X_i) - C\|x - X_i\|\} .$$

where C is the Lipschitz constant. This is a beautiful approach, whose implementation for large d seems very hard. For noisy problems, or when the dimension is large, a random version of this was proposed in Devroye (1978). If $\mathcal{X}$ is compact, X_{n+1}^* is taken uniformly in $\mathcal{X}$ minus the union of the n balls centered at the X_i's $(1 \le i \le n)$ with radius $(Q(X_i) - Q(X_n^*))/C$. If C is unknown, replace it in the formula for the radius by C_n and let $C_n \to \infty$ such that $C_n^d/n \to 0$ and $(C_{n+1}/C_n)^d = 1 + o(1/n)$ (example: $C_n = \exp((\log n)^p)$ for $p \in (0,1)$). Then $Q(X_n) \to \min Q$ almost surely.

Local random search.

Random search may proceed in a local manner, yet find a global minimum. Assume for example that we set

$$X_{n+1} = X_n^* + \sigma_n N_{n+1} ,$$

where $N_1, N_2, \ldots$ are i.i.d. normal random vectors, and $\sigma_n \to 0$ is a given deterministic sequence. The new probe point is not far from the old best point, as if one is trying to mimic local descent algorithms. However, over a compact set, global convergence takes place whenever $\sigma_n\sqrt{\log n} \to \infty$. This is merely due to the fact that $N_1, N_2, \ldots, N_n$ form a cloud that becomes dense in the expanding sphere of radius $\sqrt{2\log n}$. Hence, we will never get stuck in a local minimum. The convergence result does not put any restrictions on Q.

The above result, while theoretically pleasing, is of modest value in practice as σ_n must be adapted to the problem at hand. A key paper in this respect is by Matyas (1965), who suggests making σ_n adaptive and setting

$$X_{n+1} = X_n^* + \sigma_n N_{n+1} + D_{n+1} ,$$

where D_{n+1} is a preferred direction that is made adaptive as well. A rule of thumb, that may be found in several publications (see Devroye, 1972, and more recently, Bäck, Hoffmeister and Schwefel, 1991), is that σ_n should increase after a successful step, and decrease after a failure, and that the parameters should be adjusted to keep the probability of success around 1/5. Schumer and Steiglitz (1968) and others investigate the optimality of similar strategies for local hill-climbing. Alternately, σ_n may be found by a one-dimensional search along the direction given by N_{n+1} (Bremermann, 1968; Gaviano, 1975).

Simulated annealing.

In simulated annealing, one works with random probes as in random search, but instead of letting X_{n+1}^* be the best of X_{n+1} (the probe point) and X_n^* (the old best point), a randomized decision is introduced, that may be reformualted as follows (after Hajek and Sasaki, 1989):

$$X_{n+1}^* = \begin{cases} X_{n+1} & \text{if } Q(X_{n+1}) - Q(X_n^*) \le t_n E_n \\ X_n^* & \text{otherwise.} \end{cases}$$

where t_n is a positive constant depending for now on n only and $E_1, E_2, \ldots$ is an i.i.d. sequence of positive random variables. The best point thus walks around the space at random. If t_n, the temperature, is zero, we obtain ordinary random search. If $t_n = \infty$, $X_1^*, X_2^*, \ldots$ is a random walk over the parameter space. If $t_n > 0$ and E_n is exponentially distributed, then we obtain the Metropolis Markov chain or the Metropolis algorithm (Metropolis et al, 1953; Kirkpatrick, Gelatt and Vecchi, 1983; Meerkov, 1972; Cerny, 1985; Hajek and Sasaki, 1989). Yet another version of simulated annealing has emerged, called the heat bath Markov chain (Geman and Hwang, 1986; Aluffi-Pentini et al, 1985), which proceeds by setting

$$X_{n+1}^* = \begin{cases} X_{n+1} & \text{if } Q(X_{n+1}) + t_n Y_n \\ & \qquad \le Q(X_n^*) + t_n Z_n \\ X_n^* & \text{otherwise,} \end{cases}$$

where now $Y_1, Z_1, Y_2, Z_2, \ldots$ are i.i.d. random variables and t_n is the temperature psarameter. If the Y_i's are distributed as the extreme-value distribution (with distribution function $\exp(e^{-x})$) then we obtain the original version of the heat bath Markov chain. Note that each Y_i is then distributed as $\log\log(1/U)$ where U is uniform $[0,1]$, so that computer simulation is not hampered.

The two schemes are not dramatically different. The heat bath Markov chain as we presented it here has the feature that function evaluations are intentionally corrupted by noise. This clearly reduces the information content and must slow down the algorithm. Most random search algorithms take random steps but do not add noise to measurements; in simulated annealing, one deliberately destroys valuable information. It should be possible to formulate an algorithm that does not corrupt expensive function evaluations with noise (by storing them) and outperforms the simulated annealing algorithm in some sense. One should be careful though and only compare algorithms that occupy equal amounts of storage for the program and the data.
We now turn to the choice of t_n. In view of the representation given above, it is clear that $\mathbf{E}\{Q(X_n^*) - \min Q\}$ is bounded from below by a constant times t_n as t_n is the threshold we allow in steps away from the minimum. Hence the need to make t_n small. This need clashes with the condition of convergence. The condition of convergence depends upon the setting (the space $\mathcal{X}$ and the definition of X_{n+1} given X_n^*). We briefly deal with the specific case of finite-domain simulated annealing in the next section. In continuous spaces, progress has been made by Vanderbilt and Louie (1984), Dekkers and Aarts (1991), Bohachevsky, Johnson and Stein (1986), Gelfand and Mitter (1991), and Haario and Saksman (1991). Other key references on simulated annealing include Aarts and Korst (1989), Van Laarhoven and Aarts (1987), Anily and Federgruen (1987), Gidas (1985), Hajek (1988), and Johnson, Aragon, McGeoch and Schevon (1989).
Further work seems required on an information-theoretic proof of the inadmissibility of simulated annealing and on a unified treatment of multistart and simulated annealing, where multistart is a random search procedure in which one starts at a randomly selected place at given times or whenever one is stuck in a local minimum.

Finite domain simulated annealing.

On a finite connected graph, simulated annealing proceeds by picking a trial point uniformly at random from its neighbors. Assume the graph is regular, i.e., each node has an equal number of neighbors. If we keep the temperature $t > 0$ fixed, then there is a limiting distribution for X_n^*, called the Gibbs distribution or Maxwell-Boltzmann distribution: for the Metropolis algorithm, the asymptotic probability of node i is proportional to $e^{-Q(i)/t}$. Interestingly, this is independent of the structure of the graph. If we now let $t_n \to 0$ then with probability tending to one, X_n^* belongs to the collection of local minima. With probability tending to one, X_n^* belongs to the set of global minima if additionally, $\sum_n e^{-\Delta/t_n} = \infty$ (for example, $t_n = c/\log(n+1)$ for $c \geq \Delta$ will do). Here Δ is the maximum of all depths of strictly local minima (Hajek, 1988). The only condition on the graph is that all connected components of $\{x : Q(x) \leq c\}$ are strongly connected for any c. The slow convergence of t_n puts a severe lower bound on the convergence rate of simulated annealing.

Random walks on graphs.

Simulated annealing may be considered as a random walk on a graph if $\mathcal{X}$ is discrete. It finds the global minimum only because it exhausts the entire space. Therefore, some results from the theory of random walks on graphs are very relevant here. Consider a fixed graph with n nodes on which we perform a random walk by picking a neighbor with equal probability from all neighbors. If Q is one everywhere except at one point, where it is zero, it is immediately clear that simulated annealing must take as long as it takes our random walk to discover the zero-valued node. Related to this (but slightly larger) is the cover time T, i.e., the time needed to visit all nodes. The following results are known: there exists a universal constant $c > 0$ such that for any connected graph, and any n, $\mathbf{E}T \geq cn \log n$ (Aldous, 1989). Note that this grows faster than linear in n and is not competitive with even brute force search, in which all nodes are looked at. This casts a great shadow on any method that must find a global minimum by some sort of random walk, be it local (each node has just a few neighbors) or global (each node has many neighbors). Even for the complete graph, we have $\mathbf{E}T \sim n \log n$. Graphs that have small values of $\mathbf{E}T$ and yet few edges allow rapid travel over the space of n nodes. For example, for the hypercube graph, we have $\mathbf{E}T \sim n \log n$ as well. Cycle graphs have $\mathbf{E}T \approx n^2$ and are thus to be avoided. Kahn et al (1989) showed that for all regular graphs, $\mathbf{E}T = O(n^2)$, and Aleliunas et al (1979) showed that in any case $\mathbf{E}T = O(ne)$, where e is the number of edges. Various bounds based upon probabilistic arguments or the Perron-Frobenius theory for the second largest eigenvalue of a transition matrix may be found in Devroye and Sbihi (1990), Matthews (1988), or Broder and Karlin (1987).

Continuous-space simulated annealing.

Let us consider optimization on a compact of $\mathbb{R}^d$, and let Q be bounded there. If we let $X_{n+1} - X_n^*$ have a fixed density f that is bounded from below by a constant times the indicator of a small unit ball of positive radius, then X_n^* in the Metropolis algorithm converges to the global minimum in probability if $t_n \downarrow 0$, yet $t_n \log n \to \infty$. Bohachevsky, Johnson and Stein (1986) adjust t_n during the search to make the probability of accepting a trial point hover near a constant. Nevertheless, if t_n is taken as above, the rate of convergence to the minimum is bounded from below by $1/\log n$, which is much slower than the polynomial rate we would have if Q were multimodal but Lipschitz (recall that we had $n^{-1/d}$ rates there).

The idea to add noise to help the search is not new. In the so-called heavy ball method, it is used to make gradient descent more robust (Uosaki et al, 1970; Bekey and Ung, 1974; Geman and Hwang, 1986; Kushner, 1987). A typical step there is $X_{n+1} + X_n - \alpha_n Q'(X_n) + \beta_n N_n$, where N_n is a normal noise vector, Q' is the vector gradient, and α_n, β_n are positive sequences.

Novel ideas in random search.

Several ideas deserve more attention as they lead to potentially efficient algorithms. These are listed here in arbitrary order.

In 1975, Jarvis introduced competing searches such as competing local random searches. If N is the number of such searches, a trial (or time unit) is spent on the i-th search with probability p_i, where p_i is adapted as time evolves; a possible formula is to replace p_i by $\alpha p_i + (1 - \alpha)(c/Q(X_i))^b$, where $\alpha \in (0,1)$ is a weight, c and b are constants, and X_i is the trial point for the i-th competing search. More energy is spent on promising searches.

This idea was pushed further by several researchers in one form or another. Several groups realized that when two searches converge to the same local minimum, many function evaluations could be wasted. Hence the need for on-line clustering, the detection of points that belong somehow to the same local valley of the function. See Becker and Lago (1970), Törn (1974, 1976), de Biase and Frontini (1978), Boender et al (1982), and Rinnooy Kan and Timmer (1984, 1987).

The picture is now becoming clearer—it pays to keep track of several base points, i.e., to increase the storage. In Price's controlled random search for example (Price, 1983), one has a cloud of points of size about $25d$, where d is the dimension of the space. A random simplex is drawn from these points, and the worst point of this simplex is replaced by a trial point, if this trial point is better. The trial point is picked at random inside the simplex.

Independently, the German school developed their "Evolutionsstrategie" (Rechenberg, 1973; Schwefel, 1981). Here too we have a population of base points, each giving rise to some trial points (mutations). Of the group of trial points, we keep the best N, and repeat the process. In space, if we draw the points in different generations, we will obtain a certain tree that moves over the space towards the global minimum. We will say more about this in the section on genetic algorithms.

Bilbro and Snyder (1991) propose tree annealing: all trial points are stored in tree format, with randomly picked leaves spawning two children. The leaf probabilities are determined as products of edge probabilities on the path to the root, and the tree represents the classical k-d tree partition of the space. Their approach is at the same time computationally efficient and fast.

Finally, to deal with high-dimensional spaces, the coordinate projection method of Zakharov (1969) and Hartman (1973) deserves some attention. Picture the space as being partitioned by a $N \times \cdots \times N$ regular grid. With each marginal interval of each coordinate we associate a weight proportional to to the likelihood that the global minimum is in that interval. A cell is grabbed at random in the grid according to these (product) probabilities, and the marginal weights are updated. While this method is not fool-proof, it attempts at least to organize global search effort in some logical way.

Genetic algorithms.

Consider a population of points, called a generation. By selecting good points, modifying or mutating good points, and combining two or more good points, one may generate a new generation, which, hopefully, is an improvement over the parent generation. Iterating this process leads to the evolutionary search method (Bremermann, 1962, 1968; Rechenberg, 1973; Schwefel, 1977; Jarvis, 1975) and the body of methods called genetic algorithms (Holland, 1975). Mutations may be visualized as little perturbations by noise vectors in a continuous space. However, if $\mathcal{X}$ is the space $\{0,1\}^d$, then mutations become bit flips, and combinations of points are obtained by merging bit strings in some way. The term cross-over is often used. In optimization on graphs, mutations correspond to picking a random neighbor. The selection of good

points may be extinctive or preserving, elitist or non-elitist. It may be proportional or based on ranks. As well, it may be adaptive and allow for immigration (new individuals). In some cases, parents never die and live in all subsequent generations. The population size may be stable or explosive. Intricate algorithms include parameters of the algorithm itself as part of the genetic structure. Convergence is driven by mutation and can be proved under conditions not unlike those of standard random search.

Evolution strategies aim to mimic true biological evolution. In this respect, the early work of Bremermann (1962) makes for fascinating reading. He conjectured in 1962 that "no data processing system, artificial or living, can process more than 2×10^{47} bits per second–gram". So here is Bremermann's early computation on why biological evolution is indeed a very very hard optimization problem. A DNA string of a human consists of 4×10^9 nucleotide bridges, each taking values in the set $\{a, t, g, c\}$. To store this requires thus roughly 10^{10} (in fact, 8×10^9) bits. The number of possible humans is thus about $2^{10^{10}}$, which is much more than the number of particles in the universe. The number of second-grams needed to solve this problem is staggering. Not only is biological evolution hard, but so are many standard problems encountered in game playing, operations research, and applied mathematics. In 1968, Bremermann advocated the use of evolutionary strategies in solving linear equations, for example. A mere optimization of Q over $\{0,1\}^n$ with $n = 300$ would need more than 10^{40} second-grams for a brute-force solution if Bremermann's conjecture is valid. If all of humanity would dedicate its entire weight to this problem, this would still require about 10^{21} years for solution! In the face of such hard problems, we must set modest goals and consider the successes of biological evolution when designing search strategies.

Ackley's iterated genetic hillclimbing.

In Ackley's thesis (1987), we find a beautiful example of a genetic algorithm in action. He takes the population size to be about 50 and attaches great importance to θ, the average Q-value over the population. The following steps are repeated, starting from an initial random population. Two individuals are chosen at random and we create a new individual by performing a bitwise merge as follows: the new i-th bit is obtained with probability p by flipping a coin. Otherwise the bit is taken from one of the two parents with equal probability. After constructing the bits of the new individual, hillclimbing is done via neighbors at Hamming distance one until a local minimum is reached. This new point replaces a point picked at random from among those points in the population whose Q-value exceeds θ.

The method of generations.

As a second example of evolutionary search, this time in a continuous space, we present the method of generations as designed by Ermakov and Zhigljavsky (1983). The population size may change over time. To form a new generation, we pick parents with probability proportional to

$$\frac{Q^k(X_i)}{\sum_j Q^k(X_j)}$$

and add random perturbation vectors to each individual., where k is to be specified. The latter are distributed as $\sigma_n Z_n$, where the Z_n's are i.i.d. and σ_n is a time-dependent scale factor. This tends to maximize Q if we let k tend to infinity at a certain rate.

Additive noise.

If we have additive noise, i.e., each $Q(x)$ is corrupted by an independent realization of a random variable Z, so that we observe $Q(x) + Z$, then the standard random search algorithm may still be convergent. Formally, if $Y_1, Z_1, Y_2, Z_2, \ldots$ aree independent realizations of Z, the algorithm uses the following basic step:

$$X^*_{n+1} = \begin{cases} X_{n+1} & \text{if } Q(X_{n+1}) + Y_{n+1} \\ & \quad \leq Q(X^*_n) + Z_{n+1} \\ X^*_n & \text{otherwise.} \end{cases}$$

Assume furthermore that with probability at least $\alpha > 0$, X_{n+1} is sampled according to a fixed distribution with support on $\mathcal{X}$. Even though the decisions are arbitrary, as in simulated annealing, and even though there is no converging temperature factor, the above algorithm may be convergent in some cases, i.e., $Q(X^*_n) \to \inf Q$ in probability. For stable noise, i.e., noise with distribution function G satisfying

$$\lim_{x \downarrow -\infty} \frac{G(x-\epsilon)}{G(x)} = 0, \text{ all } \epsilon > 0 \ ,$$

such as normally distributed noise, or indeed, any noise with tails that decrease faster to zero than exponential, then we have convergence in the given sense. The reason is that for an i.i.d. sequence $\eta_1, \ldots, \eta_n$ drawn from G, $\min(\eta_1, \ldots, \eta_n) - a_n \to 0$ in probability for some sequence a_n. See for example Rubinstein and Weissman, 1979.

Bandit problems.

The classical set-up in bandit problems is that it is possible to observe a sample drawn from distribution F_x at each x, with F_x possibly different for each x. The mean of F_x is $Q(x)$. If there are just two x's, and the probe points selected by us are $X_1, \ldots, X_n$, then the purpose is to minimize

$$A_n = \frac{1}{n}\sum_{i=1}^{n} Q(X_i) \ .$$

Sometimes, we may wish to minimize

$$B_n = \sum_{i=1}^{n} I_{X_i \neq x^*}$$

where x^* is the global minimum of Q. This is relevant whenever we want to optimize a system on the fly, such as an operational control system or a game-playing program. Strategies have been developed based upon certain parametric assumptions on the F_x's or in a purely nonparametric setting. A distinction is also made between finite horizon and infinite horizon solutions. With a finite number of bandits, if at least one F_x is nondegenerate, then for any algorithm, we must have $\mathbf{E}B_n \geq c \log n$ for some constant $c > 0$ on some optimization problem (Robbins, 1952; Lai and Robbins, 1985).

In the case of bounded noise, Yakowitz and Lowe (1991) devised a play-the-leader strategy in which the trial point X_n is the best point seen thus far (based on averages) unless $n = \lfloor ae^k + b \rfloor$ for some integer k (a and b are fixed positive numbers), at which times X_n is picked at random from all possible choices. This guarantees $\mathbf{E}B_n = O(\log n)$. Thus, the optimum is missed at most $\log n$ times out of n.

Another useful strategy for parametric families F_x was proposed by Lai and Robbins (1985). Here confidence intervals are constructed for all $Q(x)$, $x \in \mathcal{X}$. The x with the smallest lower confidence interval endpoint is sampled. Exact lower bounds were derived by them for this situation. For two normal distributions with means $\mu_1 < \mu_2$ and variances σ_1^2 and σ_2^2, Holland (1973) has shown that $\mathbf{E}B_n \geq (2\sigma_1^2/(\mu_2 - \mu_1) + o(1)) \log n$.

Yakowitz and Lugosi (1989) illustrate how one may optimize an evaluation function on-line in the Japanese game of gomoku. Here each F_x represents a Bernoulli distribution and $Q(x)$ is nothing but the probability of winning against a random opponent with parameters x.

Random search in noise.

In a noisy situation when $\mathcal{X}$ is uncountable, we may minimize Q if we are given infinite storage. More formally, let $X_1, X_2, \ldots$ be trial points, with the only restriction being that at each n, with probability at least α_n, X_n is sampled from a distribution spanning the support of $\mathcal{X}$ (such as the normal density, or the uniform density on a compact). We also make sure that at least λ_n observations are available for each X_i at time n. If the noise is additive, we may consider the λ_n^2 pairings for all the observations at each of X_i and X_j, recording all values of $W(i,j)$, the number of wins of X_i over X_j, $1 \leq i \leq j \leq n$. For each X_i, let $Z_i = \min_{j \neq i} W(i,j)$, and define X_n^* as the trial point with maximal Z_i value. If $\lambda_n / \log n \to \infty$, and $\sum \alpha_n = \infty$, then $Q(X_n^*) \to \operatorname{ess\,inf} Q(X)$ almost surely (Devroye, 1977; Fisher and Yakowitz, 1973). Interestingly, there are no conditions whatever on the noise distribution. With averaging instead of a statistic based on ranks, a tail condition on the noise would have been necessary. For non-additive noise,

$$\sup_x \mathbf{E}\left\{e^{t|Y|} | X = x\right\} < \infty$$

for all $0 < t \leq t_0$ (where Y is drawn from F_x) suffices for example when X_n^* is obtained by minimizing the λ_n-averages at the trial points.

Gurin (1966) was the first to explore the idea of averages of repeated measurements. Assume again the α_n condition on the selection of trial points and let $\widehat{Q}$ denote the average of λ_n observations. Then, if $\epsilon_n \geq 0$, Gurin proceeds by setting

$$X_{n+1}^* = \begin{cases} X_{n+1} & \text{if } \widehat{Q}(X_{n+1}) < \widehat{Q}(X_n^*) - \epsilon_n \\ X_n^* & \text{otherwise.} \end{cases}$$

This is contrary to all principles of simulated annealing, as we are gingerly accepting new best points by virtue of the threshold ϵ_n. Devroye (1976) has obtained some sufficient conditions for the strong convergence of $Q(X_n^*) \to \operatorname{ess\,inf} Q(X)$. One set includes $\epsilon_n \equiv 0$, $\sup_x \operatorname{Var}\{Y|X = x\} < \infty$, and $\sum 1/\sqrt{\lambda_n} = \infty$ (a very strong condition indeed). If $\epsilon_n > 0$ and for each x, $|Y - Q(x)|$ is stochastically smaller than Z where $\mathbf{E}e^{tZ} < \infty$ for some $t > 0$, then $\epsilon_n \to 0$ and $\lambda_n \epsilon_n^2 / \log n \to 0$ are sufficient as well. In the latter case, the conditions insure that with probability one, we make a finite number of incorrect decisions. Other references along the same lines include Marti (1982), Pinter (1984), Karmanov (1974), Solis and Wets (1978), Koronacki (1976) and Tarasenko (1977).

Optimization and nonparametric estimation.

To extract the maximum amount of information from past observations, we might store these observations and construct a nonparametric estimate of the regression function $Q(x) = \mathbf{E}\{Y|X = x\}$, where Y is an observation from F_x. Assume that we have n pairs (X_i, Y_i), $1 \leq i \leq n$, where a diverging number of X_i's are drawn from a global distribution, and the Y_i's are corresponding noisy observations. Estimate $Q(x)$ by $\widehat{Q}(x)$, which may be obtained by averaging those Y_i's whose X_i is among the k nearest neighbors of x. It should be obvious that if $\|\widehat{Q} - Q\|_\infty \to 0$ almost surely, then $Q(X_n^*) \to \operatorname{ess\,inf} Q(X)$ almost surely if $X_n^* = \arg\min_i \widehat{Q}(X_i)$. It suffices for example that $k/n \to 0$, $k/\log n \to \infty$, that the noise be uniformly bounded, and that $\mathcal{X}$ be compact. Such nonparametric estimates may also be used to identify local minima.

References.

E. Aarts and J. Korst, *Simulated Annealing and Boltzmann Machines*, Wiley, NY, 1989.

D. H. Ackley, *A Connectionist Machine for Genetic Hillclimbing*, Kluwer Academic Publishers, Boston, 1987.

D. Aldous, "On the time taken by random walks on finite groups to visit every state," *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, vol. 62, pp. 361–374, 1983.

D. Aldous, "Minimization algorithms and random walk on the d-cube," *Annals of Probability*, vol. 11, pp. 403–413, 1983.

D. Aldous, "An introduction to covering problems for random walks on graphs," *Journal of Theoretical Probability*, vol. 2, pp. 87–89, 1989.

D. J. Aldous, "Lower bounds for covering times for reversible Markov chains and random walks on graphs," *Journal of Theoretical Probability*, vol. 2, pp. 91–100, 1989.

R. Aleliunas, R. M. Karp, R. J. Lipton, L. Lovasz, and C. Rackoff, "Random walks, universal transversal sequences, and the complexity of maze problems," in: *Proceedings of the 20th Annual Symposium on the Foundations of Computer Science*, pp. 218–223, 1979.

F. Aluffi-Pentini, V. Parisi, and F. Zirilli, "Global optimization and stochastic differential equations," *Journal of Optimization Theory and Applications*, vol. 47, pp. 1–16 , 1985.

S. Anily and A. Federgruen, "Simulated annealing methods with general acceptance probabilities," *Journal of Applied Probability*, vol. 24, pp. 657–667, 1987.

R. W. Becker and G. V. Lago, "A global optimization algorithm," in: *Proceedings of the 8th Annual Allerton Conference on Circuit and System Theory*, pp. 3–12, 1970.

G. A. Bekey and M. T. Ung, "A comparative evaluation of two global search algorithms," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-4, pp. 112–116, 1974.

G. L. Bilbro and W. E. Snyder, "Optimization of functions with many minima," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-21, pp. 840–849, 1991.

C. G. E. Boender and L. Stougie, G. T. Timmer, "A.H.G. Rinnooy Kan," "A stochastic method for global optimization," *Mathematical Programming*, vol. 22, pp. 125–140, 1982.

I. O. Bohachevsky, M. E. Johnson, and M. L. Stein, "Generalized simulated annealing for function optimization," *Technometrics*, vol. 28, pp. 209–217, 1986.

H. J. Bremermann, "Optimization through evolution and recombination," in: *Self-Organizing Systems*, ed. M. C. Yovits, G. T. Jacobi and G. D. Goldstein, pp. 93–106, Spartan Books, Washington, D.C., 1962.

H. J. Bremermann, "Numerical optimization procedures derived from biological evolution preocesses," in: *Cybernetic Problems in Bionics*, ed. H. L. Oestreicher and D. R. Moore, pp. 597–616, Gordon and Breach Science Publishers, New York, 1968.

H. J. Bremermann, "Numerical optimization procedures derived from biological evolution preocesses," in: *Cybernetic Problems in Bionics*, ed. H. L. Oestreicher and D. R. Moore, pp. 597–616, Gordon and Breach Science Publishers, New York, 1968.

A. Z. Broder and A. R. Karlin, "Bounds on the cover time," *Proceedings of the 29th IEEE Symposium on the Foundations of Computer Science*, pp. 479–487, 1987.

S. H. Brooks, "A discussion of random methods for seeking maxima," *Operations Research*, vol. 6, pp. 244–251, 1958.

S. H. Brooks, "A comparison of maximum-seeking methods," *Operations Research*, vol. 7, pp. 430–457, 1959.

T. Bäck, F. Hoffmeister, and H.-P. Schwefel, "A survey of evolution strategies," in: *Proceedings of the Fourth International Conference on Genetic Algorithms*, ed. R. K. Belew and L. B. Booker, pp. 2–9, Morgan Kaufmann Publishers, San Mateo, CA, 1991.

V. Cerny, "Thermodynamical aproach to the traveling salesman problem: an efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, pp. 41–51, 1985.

J. H. Conway and N. J. A. Sloane, *Sphere-Packings, Lattices and Groups*, Springer-Verlag, 1993.

H. S. M. Coxeter, L. Few, and C. A. Rogers, "Covering space with equal spheres," *Mathematika*, vol. 6, pp. 147–157, 1959.

A. Dekkers and E. Aarts, "Global optimization and simulated annealing," *Mathematical Programming*, vol. 50, pp. 367–393, 1991.

L. Devroye, "The compound random search algorithm," in: *Proceedings of the International Symposium on Systems Engineering and Analysis, Purdue University*, vol. 2, pp. 195–110, 1972.

L. Devroye, "On the convergence of statistical search," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-6, pp. 46–56, 1976.

L. Devroye, "On random search with a learning memory," in: *Proceedings of the IEEE Conference on Cybernetics and Society, Washington*, pp. 704–711, 1976.

L. Devroye, "An expanding automaton for use in stochastic optimization," *Journal of Cybernetics and Information Science*, vol. 1, pp. 82–94, 1977.

L. Devroye, "An expanding automaton for use in stochastic optimization," *Journal of Cybernetics and Information Science*, vol. 1, pp. 82–94, 1977.

L. Devroye, "Progressive global random search of continuous functions," *Mathematical Programming*, vol. 15, pp. 330–342, 1978.

L. Devroye, "The uniform convergence of nearest neighbor regression function estimators and their application in optimization," *IEEE Transactions on Information Theory*, vol. IT-24, pp. 142–151, 1978.

L. Devroye, "Global random search in stochastic optimization problems," Proceedings of Optimization Days 1979, Montreal, 1979.

L. Devroye and A. Sbihi, "Random walks on highly symmetric graphs," *Journal of Theoretical Probability*, vol. 3, pp. 497–514, 1990.

L. de Biase and F. Frontini, "A stochastic method for global optimization: its structure and numerical performance," in: *Towards Global Optimisation 2*, ed. L. C. W. Dixon and G. P. Szegö, pp. 85–102, North Holland, Amsterdam, 1978.

S. M. Ermakov and A. A. Zhiglyavskii, "On random search for a global extremum," *Theory of Probability and its Applications*, vol. 28, pp. 136–141, 1983.

Yu. Ermoliev and R. Wets, "Stochastic programming, and introduction," in: *Numerical Techniques of Stochastic Optimization*, ed. R. J.-B. Wets and Yu. M. Ermoliev, pp. 1–32, Springer-Verlag, New York, 1988.

M. Gaviano, "Some general results on the convergence of random search algorithms in minimisation problems," in: *Towards Global Optimisation*, ed. L. C. W. Dixon and G. P. Szegö, pp. 149–157, North Holland, New York, 1975.

S. B. Gelfand and S. K. Mitter, "Weak convergence of Markov chain sampling methods and annealing algorithms to diffusions," *Journal of Optimization Theory and Applications*, vol. 68, pp. 483–498, 1991.

S. Geman and C.-R. Hwang, "Diffusions for global optimization," *SIAM Journal on Control and Optimization*, vol. 24, pp. 1031–1043, 1986.

B. Gidas, "Global optimization via the Langevin equation," in: *Proceedings of the 24th IEEE Conference on Decision and Control, Fort Lauderdale*, pp. 774–778, 1985.

D. E. Goldberg, *Genetic Algorithms in Search Optimization and Machine Learning*, Addison-Wesley, Reading, Mass, 1989.

L. S. Gurin, "Random search in the presence of noise," *Engineering Cybernetics*, vol. 4, pp. 252–260, 1966.

H. Haario and E. Saksman, "Simulated annealing process in general state space," *Advances in Applied Probability*, vol. 23, pp. 866–893, 1991.

B. Hajek, "Cooling schedules for optimal annealing," *Mathematics of Operations Research*, vol. 13, pp. 311–329, 1988.

B. Hajek and G. Sasaki, "Simulated annealing—to cool or not," *Systems and Control Letters*, vol. 12, pp. 443–447, 1989.

J. H. Holland, "Genetic algorithms and the optimal allocation of trials," *SIAM Journal on Computing*, vol. 2, pp. 88–105, 1973.

R. A. Jarvis, "Adaptive global search by the process of competitive evolution," *IEEE Transactions on Systems, Man and Cybernetics*, vol. SMC-5, pp. 297–311, 1975.

D. S. Johnson, C. R. Aragon, L. A. McGeogh, and C. Schevon, "Optimization by simulated annealing: an experimental evaluation; part I, graph partitioning," *Operations Research*, vol. 37, pp. 865–892, 1989.

J. D. Kahn, N. Linial, N. Nisan, and M. E. Saks, "On the cover time of random walks on graphs," *Journal of Theoretical Probability*, vol. 2, pp. 121–128, 1989.

V. G. Karmanov, "Convergence estimates for iterative minimization methods," *USSR Computational Mathematics and Mathematical Physics*, vol. 14(1), pp. 1–13, 1974.

S. Kirkpatrick, C. D. Gelatt, and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, pp. 671–680, 1983.

J. Koronacki, "Convergence of random-search algorithms," *Automatic Control and Computer Sciences*, vol. 10(4), pp. 39–45, 1976.

H. L. Kushner, "Asymptotic global behavior for stochastic approximation via diffusion with slowly decreasing noise effects: global minimization via Monte Carlo," *SIAM Journal on Applied Mathematics*, vol. 47, pp. 169–185, 1987.

T. L. Lai and H. Robbins, "Asymptotically efficient adaptive allocation rules," *Advances in Applied Mathematics*, vol. 6, pp. 4–22, 1985.

K. Marti, "Minimizing noisy objective functions by random search methods," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 62, pp. T377–T380, 1982.

K. Marti, "Stochastic optimization in structural design," *Zeitschrift für Angewandte Mathematik und Mechanik*, vol. 72, pp. T452–T464, 1992.

P. Matthews, "Covering problems for Brownian motion on spheres," *Annals of Probability*, vol. 16, pp. 189–199, 1988.

J. Matyas, "Random optimization," *Automation and Remote Control*, vol. 26, pp. 244–251, 1965.

S. M. Meerkov, "Deceleration in the search for the global extremum of a function," *Automation and Remote Control*, vol. 33, pp. 2029–2037, 1972.

N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller, "Equation of state calculation by fast computing machines," *Journal of Chemical Physics*, vol. 21, pp. 1087–1092, 1953.

R. Männer and H.-P. Schwefel, "Parallel Problem Solving from Nature," vol. 496, Lecture Notes in Computer Science, Springer-Verlag, Berlin, 1991.

J. Pintér, "Convergence properties of stochastic optimization procedures," *Mathematische Operationsforschung und Statistik, Series Optimization*, vol. 15, pp. 405–427, 1984.

W. L. Price, "Global optimization by controlled random search," *Journal of Optimization Theory and Applications*, vol. 40, pp. 333–348, 1983.

I. Rechenberg, *Evolutionsstrategie—Optimierung technischer Systeme nach Prinzipien der biologischen Evolution*, Frommann-Holzboog, Stuttgart, 1973.

A. H. G. Rinnooy Kan and G. T. Timmer, "Stochastic global optimization methods part II: multi level methods," *Mathematical Programming*, vol. 39, pp. 57–78, 1987.

A. H. G. Rinnooy Kan and G. T. Timmer, "Stochastic global optimization methods part I: clustering methods," *Mathematical Programming*, vol. 39, pp. 27–56, 1987.

H. Robbins, "Some aspects of the sequential design of experiments," *Bulletin of the American Mathematical Society*, vol. 58, pp. 527–535, 1952.

C. A. Rogers, "Lattice coverings of space," *Mathematika*, vol. 6, pp. 33–39, 1959.

R. Y. Rubinstein and I. Weissman, "The Monte Carlo method for global optimization," *Cahiers du Centre d'Etude de Recherche Operationelle*, vol. 21, pp. 143–149, 1979.

M. A. Schumer and K. Steiglitz, "Adaptive step size random search," *IEEE Transactions on Automatic Control*, vol. AC-13, pp. 270–276, 1968.

H.-P. Schwefel, *Modellen mittels der Evolutionsstrategie*, Birkhäuser Verlag, Basel, 1977.

H.-P. Schwefel, *Numerical Optimization of Computer Models*, John Wiley, Chichester, 1981.

C. Sechen, *VLSI Placement and Global Routing using Simulated Annealing*, Kluwer Academic Publishers, 1988.

B. O. Shubert, "A sequential method seeking the global maximum of a function," *SIAM Journal on Numerical Analysis*, vol. 9, pp. 379–388, 1972.

F. J. Solis and R. B. Wets, "Minimization by random search techniques," *Mathematics of Operations Research*, vol. 1, pp. 19–30, 1981.

G. S. Tarasenko, "Convergence of adaptive algorithms of random search," *Cybernetics*, vol. 13, pp. 725–728, 1977.

G. T. Timmer, "A.H.G. Rinnooy Kan," "Stochastic methods for global optimization," *American Journal of Mathematical and Management Sciences*, vol. 4, pp. 7–40, 1984.

A. Törn, *Global Optimization as a Combination of Global and Local Search*, Skriftserie Utgiven av Handelshogskolan vid Abo Akademi, Abo, Finland, 1974.

A. Törn, "Probabilistic global optimization, a cluster analysis approach," in: *Proceedings of the EURO II Conference, Stockholm, Sweden*, pp. 521–527, North Holland, Amsterdam, 1976.

A. Törn and A. Žilinskas, *Global Optimization*, Lecture Notes in Computer Science, vol. 350, Springer-Verlag, Berlin, 1989.

K. Uosaki, H. Imamura, M. Tasaka, and H. Sugiyama, "A heuristic method for maxima searching in case of multimodal surfaces," *Technology Reports of Osaka University*, vol. 20, pp. 337–344, 1970.

D. Vanderbilt and S. G. Louie, "A Monte Carlo simulated annealing approach to optimization over continuous variables," *Journal of Computational Physics*, vol. 56, pp. 259–271, 1984.

P. J. M. Van Laarhoven and E. H. L. Aarts, *Simulated Annealing: Theory and Applications*, D. Reidel, Dordrecht, 1987.

M. T. Wasan, *Stochastic Approximation*, Cambridge University Press, New York, 1969.

S. J. Yakowitz and L. Fisher, "On sequential search for the maximum of an unknown function," *Journal of Mathematical Analysis and Applications*, vol. 41, pp. 234–259, 1973.

S. J. Yakowitz and W. Lowe, "Nonparametric bandit methods," *Annals of Operations Research*, vol. 28, pp. 297–312, 1991.

S. J. Yakowitz and E. Lugosi, "Random search in the presence of noise, with application to machine learning," *SIAM Journal on Scientific and Statistical Computing*, vol. 11, pp. 702–712, 1990.

A. A. Zhigljavsky, *Theory of Global Random Search*, Kluwer Academic Publishers, Hingham, MA, 1991.