MATHEMATICS OF OPERATIONS RESEARCH
Vol. 6, No. 1, February 1981
Printed in U.S.A.

MINIMIZATION BY RANDOM SEARCH TECHNIQUES*†

FRANCISCO J. SOLIS AND ROGER J-B. WETS

University of Kentucky

We give two general convergence proofs for random search algorithms. We review the literature and show how our results extend those available for specific variants of the conceptual algorithm studied here. We then exploit the convergence results to examine convergence rates and to actually design implementable methods. Finally we report on some computational experience.

A large class of optimization problems can be handled by random search techniques. These methods become competitive in some specific circumstances, for instance when the function characteristics—except possibly function evaluations—are difficult to compute, when there is only limited computer memory available, when the function to be minimized is very "bumpy," when it is highly desirable to find the global minimum of a function having many local minima, ... Random search techniques were first proposed by Anderson [1] and later by Rastrigin [2] and Karnopp [3]. Two questions arise naturally: Does the random search converge to the global infimum? What is the rate of convergence?

In this short note we derive convergence results for two versions of a conceptual algorithm and exhibit the relationship to the existing literature. In the second algorithm the arguments used to obtain the convergence also provide a lower bound on the convergence rate. We then exploit these convergence results to actually design algorithmic procedures. Finally we report on some computational experience.

We consider the following problem:

P *Given a function f from R^n to R and S a subset of R^n. We seek a point x in S which minimizes f on S or at least which yields an acceptable approximation of the infimum of f on S.*

Conceptual algorithm.

Step 0. Find x^0 in S and set $k = 0$.

Step 1. Generate ξ^k from the sample space $(R^n, \mathfrak{B}, \mu_k)$.

Step 2. Set $x^{k+1} = D(x^k, \xi^k)$, choose μ_{k+1}, set $k = k + 1$ and return to Step 1.

The map D with domain $S \times R^n$ and range S satisfies the following condition:

$(H1)$ $f(D(x,\xi)) \leqslant f(x)$ and if $\xi \in S$, $f(D(x,\xi)) \leqslant f(\xi)$.

The μ_k are probability measures corresponding to distribution functions defined on R^n. By M_k we denote the support of μ_k, i.e., M_k is the smallest closed subset of R^n of measure 1. Nearly all random search methods are *adaptive* by which we mean that μ_k depends on the quantities, in particular $x^0, x^1, \ldots, x^{k-1}$, generated by the preceding iterations; the μ_k are then viewed as conditional probability measures.

Clearly, here convergence must mean that with probability 1 we obtain a "monotone" sequence $\{f(x^k)\}_{k=1}^{\infty}$ which converges to the infimum of f on S. It will be

*Received March 9, 1979.
AMS 1980 subject classification. Primary 90C30.
IAOR 1973 subject classification. Main: Programming: nonlinear.
OR/MS Index 1978 subject classification. Primary 661 Programming, nonlinear, algorithms, unconstrained.
Key words. Minimization, random search.
†Supported in part by a grant of the National Science Foundation MCS 78-02864.

0364-765X/81/0601/0019$01.25

shown that the conceptual algorithm in fact produces such a sequence under very weak assumptions but we can not avoid excluding some pathological situations. Typically if the infimum of f on S occurs at a point at which f is singularly discontinuous, there is no hope to find this minimum point, barring an exhaustive examination of every point in S. Simply consider $f(x) = x^2$ when $x \neq 1$ and $f(1) = -10$, then unless the algorithm specifically tests $x = 1$, the true minimum will never be discovered. This leads to replacing the search for the infimum by that for α, the *essential infimum* of f on S, defined as follows:

$$\alpha = \inf\{t : v[x \in S | f(x) < t] > 0\},$$

i.e., the set of points that yield values close to the essential infimum α has nonzero v-measure, where v is a nonnegative measure defined on the (Borel) subsets $\mathfrak{B}$ of R^n with $v(S) > 0$. Typically $v(A)$ is simply the n-dimensional volume of the set A, more generally v is the Lebesgue measure.

By its nature the algorithm precludes the convergence to the actual minimum, which might not even exist. Hence we will seek to establish convergence to a small region surrounding (in some sense) the candidates for a minimum. The *optimality region* for P is defined by

$$R_{\epsilon,M} = \begin{cases} \{x \in S \mid f(x) < \alpha + \epsilon\} & \text{if } \alpha \text{ is finite,} \\ \{x \in S \mid f(x) < M\} & \text{if } \alpha = -\infty. \end{cases}$$

where $\epsilon > 0$ and $M < 0$. The unbounded case, which we have not excluded, is only treated for the sake of completeness; as we shall see no separate convergence argument is required.

In the implementation of the conceptual algorithm, we distinguish between local and global search methods depending on the properties of the sequence of probability measures $\{\mu^k\}$ utilized. *Local search methods* have the μ_k with bounded support M_k and for all k, but possibly a finite number, $v(S \cap M_k) < v(S)$. *Global search methods* satisfy the following assumption:

(*H*2) *For any (Borel) subset A of S with $v(A) > 0$, we have that*

$$\prod_{k=0}^{\infty} [1 - \mu_k(A)] = 0.$$

It means that given any subset A of S with positive "volume," the probability of repeatedly missing the set A, when generating the random samples ξ^k, must be zero. This requires that the sampling strategy–determined by the choice of the μ_k–cannot rely *exclusively* on distribution functions concentrated on proper subsets of S of lower dimension (such as discrete distributions) or that consistently ignore a part of S with positive "volume" (with respect to v).

We derive a convergence result for both global and local search methods. In the first case we need only minimal technical assumptions–measurability of f and S–that are always satisfied in practice.

CONVERGENCE THEOREM (GLOBAL SEARCH). *Suppose that f is a measurable function, S is a measurable subset of R^n and (H1) and (H2) are satisfied. Let $\{x^k\}_{k=0}^{\infty}$ be a sequence generated by the algorithm. Then*

$$\lim_{k \uparrow \infty} P[x^k \in R_{\epsilon,M}] = 1$$

where $P[x^k \in R_{\epsilon,M}]$ is the probability that at step k, the point x^k generated by the algorithm is in $R_{\epsilon,M}$.

PROOF. From $(H1)$ it follows that x^k or ξ^k in $R_{\epsilon,M}$ implies that $x^{k'} \in R_{\epsilon,M}$ for all $k' \geqslant k+1$. Thus

$$P[x^k \in R_{\epsilon,M}] = 1 - P[x^k \in S \setminus R_{\epsilon,M}] \geqslant 1 - \prod_{l=0}^{k} (1 - \mu_l(R_{\epsilon,M}))$$

and hence

$$1 \geqslant \lim_{k\uparrow\infty} P[x^k \in R_{\epsilon,M}] \geqslant 1 - \lim_{k\uparrow\infty} \prod_{l=0}^{k-1} (1 - \mu_l(R_{\epsilon,M})) = 1$$

where the last equality follows from $(H2)$. This completes the proof.

The convergence proofs of a number of algorithms suggested in the literature are often involved versions of the preceding, rather trivial, theorem. In [4] Gaviano proposes the following version of the conceptual algorithm. Set

$$D(x^k, \xi^k) = (1 - \lambda_k)x^k + \lambda_k \xi^k$$

where

$$\lambda_k = \arg\min_{\lambda} [f((1-\lambda)x^k + \lambda\xi^k) : (1-\lambda)x^k + \lambda\xi^k \in S]$$

and for each k, μ_k is the uniform distribution on an n-dimensional sphere centered at x_k and with radius $\geqslant 2\,\mathrm{diam}S = 2\max\{\mathrm{dist}(x, y), x, y \in S\}$. Convergence is proved when f is continuous and S is the closure of its interior and bounded. With similar assumptions Baba *et al.* [5] establish convergence when

$$D(x^k, \xi^k) = \begin{cases} \xi^k & \text{when } f(\xi^k) < f(x^k), \\ x^k & \text{otherwise,} \end{cases}$$

and each μ_k is an n-dimensional gaussian distribution with mean x^k and covariance I (the identity). Previously Archetti, Betrò and Steffè [6] considered the same algorithm but with weaker assumptions on f, replacing continuity by measurability and with μ_k, the uniform distribution on S. Matyas [7], [8] was the first to give a convergence proof for random search techniques. His method does not quite fit in our framework. He has $S = R^n$, f continuous with a unique minimum. The μ_k are again n-dimensional gaussian distributions with mean x^k and covariance I; the map D, proposed by Matyas,

$$D(x^k, \xi^k) = \begin{cases} \xi^k & \text{if } f(\xi^k) \leqslant f(x^k) - \epsilon', \\ x^k & \text{otherwise,} \end{cases}$$

for a fixed $\epsilon' > 0$, does not satisfy $(H1)$. However provided that the ϵ appearing in the definition of the optimality region $R_{\epsilon,M}$ is larger than ϵ', the same arguments yield the desired convergence. Note that if f is continuous and $S = \mathrm{cl}(\mathrm{int}\, S)$, in particular when $S = R^n$, then $\alpha = \mathrm{ess.inf}\, f = \inf f$.

Local search methods (the support M_k of μ_k is bounded and $v(S \cap M_k) < v(S)$) fail to satisfy $(H2)$ and thus convergence to the essential infimum is endangered unless we subject f, S and the μ_k to rather drastic restrictions. In general it is not possible to obtain convergence for this class of algorithms. In fact, for any one of these algorithms it is not difficult to find a function f and a set S that will trap the sequence, generated by the algorithm, in a nonoptimal region. The following condition $(H3)$ is sufficient to ensure convergence but is often difficult to verify:

$(H3)$ *To any* $x^0 \in S$, *there corresponds* $0 < \gamma$ *and* $0 < \eta \leqslant 1$ *such that*

$$\mu_k[\mathrm{dist}(D(x,\xi), R_{\epsilon,M}) < \mathrm{dist}(x, R_{\epsilon,M}) - \gamma \quad \text{or} \quad D(x,\xi) \in R_{\epsilon,M}] \geqslant \eta$$

for all k and all x in the compact set $L_0 = \{x \in S \mid f(x) \leqslant f(x^0)\}$,
where dist.(x, A) denotes the distance between a point x and a set A, i.e.,

$$\operatorname{dist}(x, A) = \inf_{y \in A} \operatorname{dist}(x, y).$$

CONVERGENCE THEOREM (LOCAL SEARCH). *Suppose that f is a measurable function, S is a measurable subset of R^n and $(H1)$ and $(H3)$ are satisfied. Let $\{x^k\}_{k=0}^{\infty}$ be a sequence generated by the algorithm. Then*

$$\lim_{k \uparrow \infty} P\left[x^k \in R_{\epsilon,M}\right] = 1$$

where $P[x^k \in R_{\epsilon,M}]$ is the probability that at step k, the point x^k generated by the algorithm is in $R_{\epsilon,M}$.

PROOF. Let x^0 be the point generated in Step 0 of the algorithm. Since L_0 is compact, by assumption $(H3)$, there always exists an integer p such that

$$\gamma p > \operatorname{dist}(x, y) \qquad \text{for all} \quad x, y \quad \text{in} \quad L_0.$$

From $(H3)$ it then follows that

$$P\left[x^p \in R_{\epsilon,M}\right] \geqslant \eta^p$$

Hence, for $k = 1, 2, \ldots$

$$P\left[x^{kp} \in R_{\epsilon,M}\right] = 1 - P\left[x^{kp} \notin R_{\epsilon,M}\right] \geqslant 1 - (1 - \eta^p)^k.$$

Now $(H1)$ implies that $x^1, \ldots, x^{p-1}$ all belong to L_0 and by the above it then follows that

$$P\left[x^{kp+l} \in R_{\epsilon,M}\right] \geqslant 1 - (1 - \eta^p)^k \quad \text{for} \quad l = 0, 1, \ldots, p - 1.$$

This completes the proof, since $(1 - \eta^p)^k$ tends to 0 as k goes to $+\infty$.

If f and S are "nice" then local search methods have a better convergence behavior than global search methods. This seems to justify their use when only a *local minimum* is requested. A number of local search methods have been suggested in the literature [2], [9]–[13]. In nearly all of these methods the vector ξ^k is obtained as a sample of a uniform distribution on M_k, a (hyper)sphere with center x^k and radius ρ_k. In the *fixed step size method* ρ_k is constant, say $\rho_k = 1$, and

$$D(x, \xi) = \begin{cases} \xi & \text{if} \quad f(\xi) < f(x) \quad \text{and} \quad \xi \in S, \\ x & \text{otherwise;} \end{cases}$$

see Rastrigin [2]. A variant is given by Mutseniyeks and Rastrigin [9], the so-called *optimized step size method*,

$$D(x, \xi) = \arg\min\left[f(y) \mid y = (1 - \lambda)x + \lambda\xi, \lambda \in R, y \in S\right]$$

with ρ_k constant. In [10] Lawrence and Steiglitz describe a fixed step size method but *with reversals*, here

$$D(x, \xi) = \begin{cases} \xi = x + (\xi - x) & \text{if} \quad f(\xi) < f(x) \quad \text{and} \quad \xi \in S, \\ 2x - \xi = x - (\xi - x) & \text{if} \quad f(2x - \xi) < f(x) \leqslant f(\xi) \quad \text{and} \quad 2x - \xi \in S, \\ x & \text{otherwise.} \end{cases}$$

Whatever the definition of D, it can always be modified to include reversals. There is substantial empirical evidence that confirms the advisability to always include rever-

sals, see for example the analysis of Schrack and Choit [11, §4]. Schumer and Steiglitz [12] introduce *adaptive step size methods*; here ρ_k is increased or decreased depending on the number of successes or failures in finding lower values of f on S in the preceding iterations. A variety of policies can be pursued in the updating of ρ_k, one such is the *optimized relative step size* with

$$\rho_{k+1} = \begin{cases} \alpha\rho_k & \text{if } x^{k+1} \neq x^k, \\ \rho_k & \text{otherwise,} \end{cases}$$

where α is a "relative optimal" constant that depends on the dimension of the problem, cf. Schrack and Choit [11, §7 and 8].

Not only do the above methods fail to satisfy $(H3)$ but they nearly always fail to converge to the essential infimum except when f and S possess specific properties. Guimier [13] proposed a *modified adaptive step size method*; where μ_k is the uniform probability measure on the *ball* of radius ρ_k and center x^k, the value of ρ_k is adjusted as a function of the quantities generated in the previous iterations. However the ρ_k must remain strictly bounded away from 0. If in addition for all $\alpha \in R$, the sets $S \cap \{x \mid f(x) \leqslant \alpha\}$ are convex and compact and S has nonempty interior, then $(H3)$ is satisfied and this algorithm will converge to the minimum. To see this, let $\inf\rho_k = \rho > 0$ and let x^0 be the point generated by the algorithm in Step 0. The set $L_0 = \{x \in S \mid f(x) \leqslant f(x_0)\}$ is compact and convex by the above. Also $R_{\epsilon,M}$ is a compact convex set with nonempty interior since S has nonempty interior. Let B' denote a ball of radius ϵ' of center c' contained in $R_{\epsilon,M}$ and suppose that $x' \in \operatorname{argmax}\{\operatorname{dist}(c', x) \mid x \in L_0\}$. Then for all x in L_0

$$\mu_k\left[\operatorname{dist}(D(x,\xi), R_{\epsilon,M}) < \operatorname{dist}(x, R_{\epsilon,M}) - 1/2\rho\right] \geqslant \eta = \mu\left[C \cap B\right] > 0,$$

where μ is the uniform measure on the ball with center x' and radius ρ, B is the ball of center c' and radius equal to $\operatorname{dist}(c', x') - \rho/2$ and C is the convex hull of x' and B'.

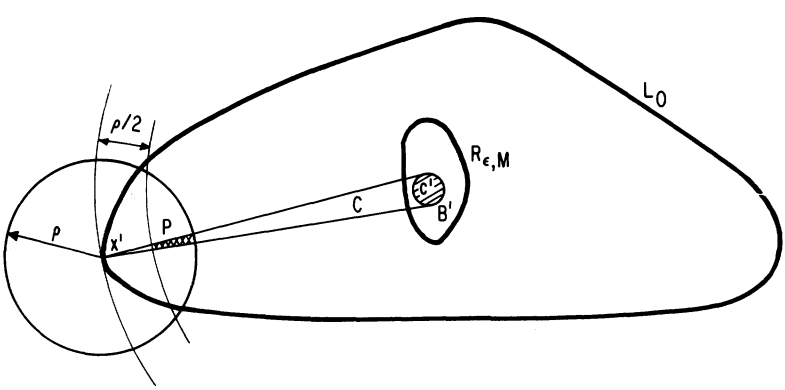

FIGURE 1

The sets $S \cap \{x \mid f(x) \leqslant \alpha\}$ will be convex-compact whenever f is quasi-convex (convex level sets) and either S is compact or f is inf-compact (bounded level sets). This certainly holds when f is strictly convex and inf-compact, the case for which Guimier [13] proves the convergence of the algorithm.

Stopping criteria. The construction of a sequence $\{x^k\}_{k=0}^{\infty}$ with $\alpha = \lim_k f(x^k)$ is at best of academic interest. In practice, we need a criterion that allows us to stop the algorithm after a finite number of iterations. Ideally, given $\beta \in]0, 1[$, we like to

compute N_β such that

$$P\left[x^k \notin R_{\epsilon,M}\right] \leqslant \beta \quad \text{for all} \quad k \geqslant N_\beta.$$

Suppose that $\mu_k(R_{\epsilon,M}) \geqslant m > 0$ for all k, then

$$P\left[x^k \notin R_{\epsilon,M}\right] \leqslant (1-m)^k.$$

Choosing an integer $N_\beta \geqslant [\ln \beta/\ln(1-m)]$ has the required property, since for $k \geqslant N_\beta$ it follows that $k \geqslant \ln \beta/\ln(1-m)$ and hence $\beta \leqslant (1-m)^k$. Unfortunately we seldom know a positive lower bound for $\mu_k(R_{\epsilon,M})$, at least not in the situations when we need to resort to random search techniques.

The most interesting research on stopping criteria involves deriving estimates for the infimum value by finding an approximation of the distribution function of the random variable $f(x^k)$. Chichinadze [14] relies on polynomial fits to approximate this distribution function, when f is continuous, S is compact, $R_{\epsilon,M} \subset \text{int}S$ for ϵ sufficiently small and for all k, μ_k is the uniform distribution on S. However, in general this distribution function does not have a polynomial behavior, as pointed out by A. Curtis, see [15]. Archetti, Betrò and Steffè [6] have, under some regularity conditions, found a good description of this distribution provided that we sample a sufficient number of points in the neighborhood of $R_{\epsilon,M}$, ϵ being sufficiently small. Unfortunately this last requirement precludes the development of a truly practical criterion. In fact, *the search for a good stopping criterion seems doomed to fail.* In [15], Dixon notes that even with S compact convex and f twice continuously differentiable, at each step of the algorithm there will remain an unsampled square region of nonzero measure v (volume) on which f can be redefined (by means of spline fits) so that the unsampled region now contains the global minimum.

Rates of convergence. The description of acceptable norms to evaluate the efficiency of a random search technique by comparison to other random search techniques or deterministic algorithms remains the major research question in this area. The most promising and tractable approach is the study of the distribution of the number of steps required to reach the essential infimum, more specifically by comparison of the expected number of steps and/or higher moments of this distribution. To do this we must rely on idealized situations, clearly not all possible (measurable) functions can serve as test functions.

Some piecemeal results about the "convergence rate" of local search methods have appeared in the literature. In [2] Rastrigin shows that with f linear, $S = R^n$, $n \geqslant 2$, the fixed step random search algorithm "moves" faster (in the direction of steepest descent) than a sequential coordinatewise minimization algorithm. On the other hand Lawrence and Steiglitz [10] have epxerimental results that indicate that near a local minimum coordinatewise search converges faster than random search but is more likely to get trapped on the boundary of S. In [11], Schrack and Choit develop heuristic rules for choosing ("optimize") the step size ρ_k (see above) and report encouraging experimental results [16]. They also prove that the "convergence" rate will be improved if the function D includes reversals [11, §5]. Finally, Schumer and Steiglitz [12] derive an asymptotic formula which shows that the adaptive step size method converges at a "linear" rate depending on the dimension. We will return to this question in connection with our experimental results.

Experimental results. Although we tested a wide variety of variants of the conceptual algorithm, we came to rely almost exclusively on the versions described below, which seem to exhibit the best convergence properties. Although we did solve a

number of contrained minimization problems, even some hard problems, all the data recorded here is for unconstrained problems, i.e., $S = R^n$. It is the only class of problems for which comparison data is available.

BASIC ALGORITHM.

Step 0. Pick $x^0 \in R^n$, set $k = 0$, $\#s = \#f = 0$ and $b^0 = 0 \in R^n$. Fix $\rho_{-1}, \rho_{lb}, ex, ct$, s_{ex} and f_{ct}.

Step 1. Set

$$\rho_k = \begin{cases} \rho_{k-1} \cdot ex & \text{if} \quad \#s \geqslant s_{ex}, \\ \rho_{k-1} \cdot ct & \text{if} \quad \#f \geqslant f_{ct}, \\ \rho_{k-1} & \text{otherwise.} \end{cases}$$

If $\rho_k \leqslant \rho_{lb}$, stop; the point x^k is "optimal". Otherwise obtain ξ as a sample of the multivariate distribution μ_k.

Step 2. Set

$$x^{k+1} = \begin{cases} \xi & \text{if} \quad f(\xi) < f(x^k), \text{ set} \quad \#s = \#s + 1, \#f = 0 \\ & \text{and} \quad b^{k+1} = 0.4(\xi - x^k) + 0.2b^k, \\ 2x^k - \xi & \text{if} \quad f(2x - \xi) < f(x^k) \leqslant f(\xi), \text{ set} \quad \#s = \#s + 1, \#f = 0 \\ & \text{and} \quad b^{k+1} = b^k - 0.4(\xi - x^k), \\ x^k & \text{otherwise, set} \quad \#s = 0, \#f = \#f + 1, b^{k+1} = (0.5)b^k; \end{cases}$$

and return to Step 1 with $k = k + 1$.

The parameters that appear in the basic algorithm have the following interpretation:

k is the iteration counter;

ρ_k determines the "diameter" of the region from which ξ is most likely to be extracted;

ρ_{lb} is a lower bound on the value of the ρ_k. If ρ_k reaches the value ρ_{lb}, the "region" being sampled is so small that it appears useless to continue the search for better values of f. The only way ρ_k can reach this value is by a long string of failures in finding smaller values of f;

b^k is a bias factor, slanting the sampling in favor of the directions where success has been recorded (the introduction of this bias factor is due to Matyas);

$\#s$ and $\#f$ are respectively the number of successive successes and failures in decreasing the value of f in the preceding interations; if $\#s$ or $\#f$ reach respectively s_{ex} or f_{ct} we expand (by a factor ex) or contract (by a factor ct) the "diameter" ρ_{k-1} of the probability mass.

In theory ρ_{lb} could be chosen to be 0. In this case, the algorithm would produce a sequence of points which converge to the minimum for a suitable f. But since we want to stop when we are "sufficiently close", in practice we choose ρ_{lb} to be a small positive quantity. In the running of the algorithm, we used $s_{ex} = 5$, $f_{ct} = 3$, $ex = 2$, $ct = 0.5$, $\rho_{-1} = 1$ and ρ_{lb} to be picked as a function of the desired accuracy and the dimension of the space.

ALGORITHM 1. Use the Basic Algorithm with μ_k the multivariate normal with center $(x^k + b^k)$ and covariance matrix $\rho_k I$.

It would not be too hard to make Algorithm 1 satisfy $(H2)$ and thus become a global search method. For example if for all k, with μ_k the multivariate normal with covariance $\rho_k I$, and mean y^k, where y^k is the projection of x^k on a compact subset C

of R^n, for example $C = [-1, 1]^n$. We replace the stopping criterion:

$$\text{if} \quad \rho_k \leqslant \rho_{lb}, \quad \text{then stop}$$

by

$$\text{if} \quad \rho_k \leqslant \rho_{lb}, \quad \text{set} \quad \rho_k = \rho_{ub} > \rho_{lb} > 0.$$

The quantity ρ_{ub} is fixed at the outset. Unfortunately this is not very practical and the convergence will generally be very slow.

ALGORITHM 2. Use the Basic Algorithm with μ_k the uniform distribution on the hypercube of center $(x^k + b^k)$ and side-length ρ_k.

If f is quasi-convex and inf-compact, and $\rho_{lb} > 0$ then this algorithm will find a point in the neighborhood of a minimum point since $(H1)$ and $(H3)$ are satisfied. In general, this is a local search method with reversals. Note again, that in general there is no guarantee for convergence to the global minimum. The method will lead us to a local minimum. In running the algorithm we used $s_{ex} = 5$, $f_{ct} = 3$, $ex = 2$, $ct = 0.5$, $\rho_{-1} = 1$ and ρ_{lb} as in Algorithm 1.

ALGORITHM 3. Same as the conceptual algorithm with μ_k the uniform measure on S (a bounded Borel set) and

$$D(x, \xi) = \begin{cases} A(\xi) & \text{if } f(A(\xi)) < f(x) \\ x & \text{otherwise,} \end{cases}$$

with $A(\xi)$ the local minimum of f generated by Algorithm A when starting at the point ξ. In the implementation of Algorithm 3, we used the following two methods to generate $A(\xi)$:

(a) conjugate direction methods (with no derivatives) as available in the Harwell subroutines.

(b) Algorithm 2 as described above.

This is a global search method, conditions $(H1)$ and $(H2)$ are satisfied. The convergence rate will tend to be of the same order as that of the algorithm used to find $A(\xi)$. This algorithm invariably provided the best results; it would find the global (essential) minimum with, on the average, the minimum amount of work.

To test the local convergence properties we minimized the function $f(x) = x^T \cdot x$. For the test runs associated with Tables 1, 2 and 3 the iterations were stopped when $\|x^k\| \leqslant 10^{-3}$.

Among other things, the preceding tables indicate that there is a linear correlation between the mean number of function evaluations and the dimension, i.e., $N = Kn$. The constant K is a function of the algorithm used, in Table 1 this constant appears to be near 40, whereas in Table 2 K seems to be near 33 (with Algorithm 2 but with the optimal choice for the step size parameters, one should be able to reach 29.5 [12]).

TABLE 1.

Minimize $x^T \cdot x$, $x^0 = (1, 0, \ldots, 0)$, *Algorithm* 1, 20 *Runs for each n.*

Dimension	Mean numb. Funct. eval.	Stand. deviat. from mean numb.	Stand. Error		
n	N	σ_N	$\sigma_N/\sqrt{20}$	$\sigma_N/n\sqrt{20}$	N/n
2	73.3	15.4	3.4	1.7	36.7
3	114.0	23.0	5.1	1.7	38.0
5	201.0	33.0	7.4	1.5	40.3
10	408.0	59.0	12.2	1.3	40.8

TABLE 2.

Minimize $x^T \cdot x, x^0 = (1, 0, \ldots, 0)$, *Algorithm* 2, 20 *Runs for each n.*

Dimension n	Mean numb. funct. eval. N	Stand. deviation from mean numb. σ_N	Standard Error $\sigma_N/\sqrt{20}$	$\sigma_N/n\sqrt{20}$	N/n
2	62.8	12.61	2.8	1.4	31.4
3	100.3	18.74	4.2	1.4	33.4
5	160.9	25.8	5.8	1.5	32.2
10	348.0	38.0	8.5	0.8	34.8

These last results are similar to those obtained by Schrack and Borowski [16]. This "linearity" was first observed by Schumer and Steiglitz [12], the algorithm that they propose has $K \simeq 80$. To explain the linear relation we note, that for large n and with optimal choice for the step size, the probability p of success (i.e., to obtain a decrease in the value of f) is given by

$$p \simeq \frac{(1 - \rho^2/4)^{(n-1)/2}}{(n-1) \cdot 2\rho(\pi/2n)^{1/2}}$$

where ρ is the optimal step size, see [12]. Now the adaptive step size methods, Algorithm 1 and 2, do not tend to produce the optimal step size ρ but they tend to maintain the probability of success constant. Since the above relation is valid when the step size ρ is small, for a given p we can use it to find ρ as a function of the dimension n. We start from the assumption that ρ is of the form $c/\sqrt{n}$ (as is the case for the optimal choice of ρ with $c = 1.225$), we get

$$p \simeq \frac{\rho/2 - (n-2)\rho^3/48 + (n-2)(n-3)\rho^5/640}{2\rho(\pi/2n)^{1/2}}$$

$$\simeq \frac{c/2\sqrt{n-2} - (n-2)c^3/48(n-2)\sqrt{n-2} + (n-2)(n-3)c^5/640(n-2)^2\sqrt{n-2}}{2\left(\pi/\sqrt{n-2}\right)^{1/2}}$$

$$= \frac{c}{4\sqrt{\pi}}\left[1 - c^2/24 + c^4(n-3)/320(n-2)\right]$$

which tends to a constant as n goes to $+\infty$. So we see that these adaptive algorithms tend to fix the step size near $Kn^{-1/2}$. Since the expected decrease in the function value is $\rho^2 p$ [12] we see that the expected step in the direction of the solution is cp/n. Now cp is constant and this would then explain the linear correlation. (This is only a heuristic justification of this linearity result, the argument is weak since the Algorithms 1 and 2 do not necessarily maintain the probability of success equal to the *same* constant for different dimensions.)

Nonetheless, this correlation coefficient K allows us to compare various random search techniques at least as far as their effectiveness in obtaining local minima. To illustrate this point let us compare the performances of Gaviano's algorithm [4] and Algorithm 2 when minimizing $x^T \cdot x$. Gaviano's algorithm proceeds as follows:

STEP 0. Pick x^0, set $k = 0$;

STEP 1. Generate a sample ξ from a uniform distribution on the unit hypersphere;

STEP 2. Set $x^{k+1} = \operatorname{argmin}\{f(y) \mid y = x^k + \lambda\xi, \lambda \in R\}$ and return to Step 1 with $k = k + 1$.

It is not difficult to see that then

$$E\{\|x^{k+1}\|\} = r_n \cdot \|x^k\|$$

where

$$r_n = \left(\int_0^{\pi/2} \sin^{n-1}\alpha \, d\alpha\right) \cdot \left(\int_0^{\pi/2} \sin^{n-2}\alpha \, d\alpha\right)^{-1}$$

$$\simeq \left[(n-2)/(n-1)\right]^{1/2}$$

for n sufficiently large. We are interested in a number N, such that after N steps we may expect the distance to the minimum, here the origin, to decrease by a factor $\epsilon < 1$. Now $N = \log\epsilon/\log r_n$. Since r_n tends to 1, for large n we can approximate $\log r_n$ by $(r_n - 1)$. For large n

$$n(r_n - 1) \simeq n\left(\sqrt{(n-2)/(n-1)} - 1\right)$$

$$= -n/\left[(n+1) + (n^2 - 3n + 2)^{1/2}\right] \simeq -n/(2n-1).$$

The last term tends to $-1/2$ as n goes to $+\infty$. Thus

$$N \simeq -2n\log\epsilon.$$

For $\epsilon = 10^{-3}$, $N \simeq (13.81)n$. The experimental results reported in Table 2 show that the same reduction is achieved in about $35n$ steps. Thus Gaviano's method becomes competitive if the 1-dimensional minimization in Step 2 of this algorithm involves, on the average, less than $35/13.81 \simeq 2.6$ function evaluations.

Table 3 exhibits the *linear convergence rate* of Algorithm 2. To take into account the random nature of the algorithm we considered the ratios

$$\|\bar{x}^{20(k+1)}\|/\|\bar{x}^{20k}\| = s_k$$

rather than to measure the step by step reduction. Also $\bar{x}^k$ is generated by averaging 20 samples obtained for x^k. We let k range from 1 to 10 to compute μ. For each n, the regression line

$$s = a_n k + b_n$$

is obtained from the pairs $\{(k, s_k),\ k = 1, \ldots, 10\}$. Note that all the coefficients a_n are negative which seems to indicate that the convergence rate is slightly better than "linear". The "linear" convergence rate is the expected value of $\|x^{k+1}\|/\|x^k\|$, here approximated by r.

TABLE 3.

Minimize $x^T \cdot x$, x_0 arbitrarily chosen, Algorithm 2.

Dimension n	Mean Reduction in 20 Steps μ	Stand. Dev. of Mean Reduc. σ_μ	Regression Coefficient a	Regression Constant b	Mean Conver. Rate r
2	0.14	0.017	− 0.0012	0.145	0.906
3	0.25	0.042	− 0.0034	0.262	0.932
4	0.34	0.067	− 0.0119	0.394	0.948
5	0.43	0.040	− 0.0071	0.459	0.958
6	0.50	0.059	− 0.0102	0.548	0.966
10	0.68	0.042	− 0.0014	0.688	0.981

The remaining table gives the average number of function evaluations μ, the standard deviation σ and the maximum number of function evaluations recorded in 20 runs, when solving a number of classical problems in global optimization. Algorithm 3 is used with variants *a* or *b* as indicated. Computations were halted when the known minimum was attained (with a tolerance of .001 in the norm of x). These results compare favorably to any other reported in the literature, cf. [15]. Note however that in [15] the number of function evaluations recorded include those required to "verify" that the optimal solution has been reached.

TABLE 4.

Algorithm 3, x_0 randomly generated, 20 runs.

Problem	SQRN 5	SQRN 7	SQRN 10	Hartm. 3	Hartm. 6	6 Hump C
Variant	3*a*	3*a*	3*a*	3*a*	3*a*	3*b*
Mean numb. fct. eval. μ	187	273	246	149	158	135
Stand. dev. σ	86	157	198	78	14	32
Max, numb. fct. eval.	405	644	936	345	185	Not avail.

$m = 5, 7, 10$

SQRN$m = -\sum_{l=1}^{m}[(x - a^l)^T \cdot (x - a^l) + c_l]^{-1}$, $S = [0, 10]^4 \subset R^n$.

l	1	2	3	4	5	6	7	8	9	10
a^l	0.4	1.0	8.0	6.0	3.0	2.0	5.0	8.0	6.0	7.0
	0.4	1.0	8.0	6.0	7.0	9.0	5.0	1.0	2.0	3.6
	0.4	1.0	8.0	6.0	3.0	2.0	3.0	8.0	6.0	7.0
	0.4	1.0	8.0	6.0	7.0	9.0	3.0	1.0	2.0	3.6
c^l	0.1	0.2	0.2	0.4	0.4	0.6	0.3	0.7	0.5	0.5

$d = 3, 6.$

Hartm $d = -\sum_{i=1}^{4} c_i \exp(-\sum_{j=1}^{d} \alpha_{ij}(x_j - p_{ij})^2)$, $S = [0, 1]^d \subset R^d$.

$d = 3$

$$[\alpha_{ij}] = \begin{bmatrix} 3.0 & 0.1 & 3.0 & 0.1 \\ 10.0 & 10.0 & 10.0 & 10.0 \\ 30.0 & 35.0 & 30.0 & 35.0 \end{bmatrix} \quad [p_{ij}] = \begin{bmatrix} 0.3689 & 0.4699 & 0.1091 & 0.03815 \\ 0.117 & 0.4387 & 0.8732 & 0.5743 \\ 0.2673 & 0.747 & 0.5547 & 0.8828 \end{bmatrix}$$

$c = [1, 1.2, 3, 3.2]$

$d = 6$

$$[\alpha_{ij}] = \begin{bmatrix} 10.0 & 0.05 & 3.0 & 17.0 \\ 3.0 & 10.0 & 3.5 & 8.0 \\ 17.0 & 17.0 & 1.7 & 0.05 \\ 3.5 & 0.1 & 10.0 & 10.0 \\ 1.7 & 8.0 & 17.0 & 0.1 \\ 8.0 & 14.0 & 8.0 & 14.0 \end{bmatrix} \quad [p_{ij}] = \begin{bmatrix} 0.1312 & 0.2329 & 0.2348 & 0.4047 \\ 0.1696 & 0.4135 & 0.1451 & 0.8828 \\ 0.5569 & 0.8307 & 0.3522 & 0.8732 \\ 0.0124 & 0.3736 & 0.2883 & 0.5743 \\ 0.8283 & 0.1004 & 0.3047 & 0.1091 \\ 0.5886 & 0.9991 & 0.6650 & 0.0381 \end{bmatrix}$$

$c = [1, 1.2, 3, 3.2]$

6 Hump C. function

$$S = [-3, 3] \times [-1.5, 1.5] \subset R^2$$

$$f(x) = 4x_1^2 - 2.1x_1^4 + \tfrac{1}{3}x_1^6 + x_1x_2 - 4x_2^2 + 4x_2^4.$$

Addendum. In "On Accelerations of the Convergence in Random Search Methods," to appear in *Operation Research Verfahren,* K. Marti obtains related convergence results for the conceptual algorithm as well as for more structured algorithmic procedures.

References

[1] Anderson, R. L. (1953). Recent Advances in Finding Best Operating Conditions. *J. Amer. Statist. Assoc.* **48** 789–798.

[2] Rastrigin, L. A. (1963). The Convergence of the Random Search Method in the Extremal Control of a Many-Parameter System. *Automat. Remote Control.* **24**. 1337–1342.

[3] Karnopp, D. C. (1963). Random Search Techniques for Optimization Problems. *Automatica.* **1** 111–121.

[4] Gaviano, M. (1975). Some General Results on the Convergence of Random Search Algorithms in Minimization Problems. In *Towards Global Optimization*, L. Dixon and G. Szegö, eds. North Holland, Amsterdam.

[5] Baba, N., Shoman, T. and Sawaragi, Y. (1977). A Modified Convergence Theorem for a Random Optimization Algorithm. *Information Sci.* **13**.

[6] Archetti, F., Betrò, B. and Steffè, S. (1975). A Theoretical Framework for Global Optimization via Random Sampling. Cuaderni del Dipartimento di Ricerca Operative e Scienze Statische, Universitàa di Pisa, A–25.

[7] Matyas, J. (1965). Random Optimization. *Automat. Remote Control.* **26**. 246–253.

[8] ———. (1968). Das Zufällige Optimierungs Verfahren und Seine Konvergenz. In *5th International Analogue Computation Meetings*. 540–544.

[9] Mutseniyeks, V. A. and Rastrigin, L. (1964). Extremal Control of Continuous Multi-Parameter Systems by the Method of Random Search. *Eng. Cybernetics*. **1**. 82–90.

[10] Lawrence, J. P., III and Steiglitz, K. (1972). Randomized Pattern Search. *IEEE Trans. Computers.* **C–21** 382–385.

[11] Schrack G., and Choit, M. (1976). Optimized Relative Step Size Random Searches. *Math Programming*. **10** 230–244.

[12] Schumer, M. A. and Steiglitz, K. (1968). Adaptive Step Size Random Search. *IEEE Trans. Automatic Control*. **AC–13** 270–276.

[13] Guimier, A. (1975). Algorithmes d'Optimisation á Stratégie Aleatoire. Thèse, Université de Provence.

[14] Chichinadze, V. K. (1967). Random Search to Determine the Extremum of the Function of Several Variables. *Eng. Cybernetics*. **1** 115–123.

[15] Dixon, L. C. (1977). Global Optima without Convexity. Tech. Report, Num. Optim. Center, Hatfield Polytechnic.

[16] Schrack, G. and Borowski, N. (1972). An Experimental Comparison of Three Random Searches. In *Numerical Methods for Nonlinear Optimization*, F. Lootsma, ed. Academic Press, London, 137–147.

DEPARTMENT OF MATHEMATICS, UNIVERSITY OF KENTUCKY, LEXINGTON, KENTUCKY 40506