Journal of Global Optimization **6:** 1–37, 1995.

An Adaptive Simulated Annealing Algorithm for Global Optimization over Continuous Variables*

ANDREW E. W. JONES and G. W. FORBES
The Institute of Optics, University of Rochester, Rochester, NY 14627, USA.

(Received: 16 April 1992; accepted: 19 September 1994)

Abstract. A method is presented for attempting global minimization for a function of continuous variables subject to constraints. The method, called *Adaptive Simulated Annealing* (ASA), is distinguished by the fact that the fixed temperature schedules and step generation routines that characterize other implementations are here replaced by heuristic-based methods that effectively eliminate the dependence of the algorithm's overall performance on user-specified control parameters. A parallel-processing version of ASA that gives increased efficiency is presented and applied to two standard problems for illustration and comparison.

Key words: Global optimization, simulated annealing, Monte Carlo optimization.

1. Introduction

The numerical optimization method known as *simulated annealing* was proposed by Kirkpatrick *et al.* [8] and by Černý [2] for attempting the solution of combinatorial optimization problems. It is based upon a method developed by Metropolis *et al.* [9] for equation-of-state calculations for systems of interacting particles. Vanderbilt and Louie (1984) generalized the algorithm to the optimization of functions of continuous variables.[1] Several variants of annealing have since been developed (e.g., [13], [1], [3]); these methods typically require the user to specify values for a variety of input parameters that are specific to the problem at hand and must generally be found by trial and error. The performance of these algorithms generally depends sensitively upon the values of these control parameters, and the results presented are typically best-case performance. As a result, much of the true cost of the computation is effectively hidden.

The *Adaptive Simulated Annealing* (ASA) algorithm presented here attempts the location of the global minimum of an objective function, $f(\mathbf{x})$, of N continuous variables $\mathbf{x} = (x_1, \ldots, x_N)$ within a region of interest Ω that is bounded by J inequality constraints $q_j(\mathbf{x}) > 0$, $j = 1, \ldots, J$. The goal in the design of ASA is to eliminate the hand-picked, problem-specific parameters while retaining a performance that, at least, rivals the best-case efficiency of other variants of simulated annealing. That is, the goal is to avoid hidden costs.

* This research was supported by the University Research Initiative of the U.S. Army Research Office.

Superficially, the annealing algorithm is simple. First, an initial value for the *acceptance parameter* β is chosen (β is the reciprocal of what is called the *temperature* in some variants of annealing). Next, a starting point is chosen as the initial *base point* $\mathbf{x}_b$,and f_b is initialized: $f_b = f(\mathbf{x}_b)$. Annealing then consists of the cyclic repetition of the following operations until a termination criterion is satisfied:

1. *Generation and Evaluation:* A random *step* $\mathbf{s}$ is selected from a distribution called the *generator* to form the *trial point* $\mathbf{x}_t = \mathbf{x}_b + \mathbf{s}$, and the objective function is evaluated at $\mathbf{x}_t$, yielding $f_t = f(\mathbf{x}_t)$.
2. *Examination:* If $f_t < f_b$ the trial point is accepted (i.e. $\mathbf{x}_b$ and f_b are set equal to $\mathbf{x}_t$ and f_t, respectively). If $f_t \geqslant f_b$ the trial point is accepted with probability $e^{-\beta(f_t - f_b)}$.
3. *Parameter updating:* The values of β and the parameters that control the generator are all updated.

Annealing is a pseudo-random process that can be modelled by using three entities: first, the *occupation density* $p_i(\mathbf{x})$ is defined such that $p_i(\mathbf{x})\,\mathrm{d}V_x$ is the probability that the base point after iteration i is located within the volume element $\mathrm{d}V_x$ at the position $\mathbf{x}$. The *generator* $g(\mathbf{y}, \mathbf{x}; \Psi)$ is defined such that $g(\mathbf{y}, \mathbf{x}; \Psi)\,\mathrm{d}V_y$ is the probability that a step is generated to a trial point within the volume element $\mathrm{d}V_y$ at $\mathbf{y}$, given that the base point is $\mathbf{x}$. Here, Ψ represents a set of parameters that determines the form of the generator. Finally, the *acceptor* $a(\mathbf{y}, \mathbf{x}; \beta)$ gives the probability of accepting the move from $\mathbf{x}$ to $\mathbf{y}$:[2]

$$a(\mathbf{y}, \mathbf{x}; \beta) = e^{-\beta \max[f(\mathbf{y}) - f(\mathbf{x}), 0]} \tag{1}$$

With these entities, a single cycle of the annealing process can be modelled as

$$\begin{aligned} p_{i+1}(\mathbf{x}) \;=\; & p_i(\mathbf{x}) + \int_\Omega [p_i(\mathbf{y}) a(\mathbf{x}, \mathbf{y}; \beta) g(\mathbf{x}, \mathbf{y}; \Psi) \\ & - p_i(\mathbf{x}) a(\mathbf{y}, \mathbf{x}; \beta) g(\mathbf{y}, \mathbf{x}; \Psi)]\,\mathrm{d}V_y, \end{aligned} \tag{2}$$

where $\int_\Omega$ denotes integration over the entire region of interest.

Since the annealing process is a discrete-time, continuous-space Markov process, the occupation density approaches an *equilibrium density* $\underline{\pi}(\mathbf{x}; \beta, \Psi)$.[3] That is, provided that β and Ψ are held fixed,[4] in the limit of large i, and regardless of the form of $p_0(\mathbf{x})$, the occupation density $p_i(\mathbf{x})$ approaches $\underline{\pi}(\mathbf{x}; \beta, \Psi)$. The equilibrium density can be determined by first setting $p_{i+1} = p_i = \underline{\pi}$ in equation (2):

$$\begin{aligned} 0 \;=\; & \int_\Omega [\underline{\pi}(\mathbf{y}; \beta, \Psi) a(\mathbf{x}, \mathbf{y}; \beta) g(\mathbf{x}, \mathbf{y}; \Psi) \\ & - \underline{\pi}(\mathbf{x}; \beta, \Psi) a(\mathbf{y}, \mathbf{x}; \beta) g(\mathbf{y}, \mathbf{x}; \Psi)]\,\mathrm{d}V_y. \end{aligned} \tag{3}$$

When the generator is *symmetric* [i.e. $g(\mathbf{y}, \mathbf{x}; \Psi) \equiv g(\mathbf{x}, \mathbf{y}; \Psi)$], the equilibrium distribution is independent of Ψ and can be found by using the condition of *detailed balance*.[5]

$$\underline{\pi}(\mathbf{x}; \beta) = \alpha(\beta) e^{-\beta f(\mathbf{x})}, \tag{4}$$

where $\alpha(\beta)$ is simply a normalization factor. That is, $\alpha(\beta)$ is given by

$$\alpha(\beta) = \left[\int_{\Omega} e^{-\beta f(\mathbf{y})}\, \mathrm{d}V_y\right]^{-1}. \tag{5}$$

It follows that locating the global minimum of f with certainty corresponds to attaining equilibrium for $\beta = \infty$ – an impossibility given only a finite number of iterations. The goal then is to realize a density that is sufficiently "close" to equilibrium for a sufficiently large value of β.

In general, a quasi-equilibrium density for large β cannot be attained efficiently simply by fixing β at that large value from the outset. This is impractical because equilibration is then unworkably slow, and a significant fraction of the occupation density becomes "trapped" in various local minima. The strategy, therefore, is to begin the annealing process with β much less than its final value, so that quasi-equilibrium may be attained relatively quickly (in ASA, β is initially set to zero so that the equilibrium density is initially uniform); β is then raised gradually while quasi-equilibrium conditions are maintained. The sequence of values of β used during the process – called the *annealing schedule* – is one of the two principal components that determine the algorithm's efficiency. If β is raised too quickly, the process is driven far from equilibrium: a significant portion of the occupation density becomes effectively trapped within the local minima. If β is raised too slowly, an excessively large number of iterations is required to reach a value of β that is sufficiently large to give adequate concentration about the global minimum.

Many variants of simulated annealing proceed according to a fixed, *ad hoc* annealing schedule (e.g., set $\beta_i = \beta_1 \ln(1 + i)/\ln(2)$, where i is the iteration number and β_1 is a problem-specific parameter). Sometimes the generator is also fixed. Instead of fixing the forms of the annealing schedule and the generator, ASA is designed according to the following heuristic:

> Raise β as quickly as possible while attempting to keep the occupation density within a specified distance of the current equilibrium density.

One measure of the *distance* between two occupation densities is defined in Section 2.2. For a typical problem, it can be expected that it is possible to raise β relatively quickly during certain stages of the annealing process without straying too far from equilibrium.

In ASA, *ad hoc* annealing schedules and generators are replaced by methods that are based upon statistical analyses of the process up to the current iteration. Since the past behavior of the process is considered, it becomes non-Markov and conventional equilibration is no longer assured; in fact, it would be possible for such "feedback" to cause collapse or stagnation. By careful design, however, it is possible to maintain approximate validity for equation (4). The conceptual foundations for the adaptive control of β and Ψ in ASA are presented in Section 2. This treatment is divided into

subsections that relate to equilibration measures, step generation, and the annealing schedule. The more detailed ideas related to the algorithm's implementation are considered in Section 3, along with modifications that *parallelize* the process. The paper ends with empirical comparisons and concluding remarks.

2. The ASA Process

In most applications of optimization algorithms, the time spent on evaluating the objective function dominates all other aspects of the computation. Efficiency is therefore defined by reference to the number of function evaluations, and it turns out that the efficiency of an adaptive simulated annealing algorithm can be enhanced by using parallel processing. One convenient parallel processing arrangement for ASA comprises a single *master* processor and K *peripheral* processors. During each iteration, each peripheral processor simultaneously generates an *independent* step from a distribution that is identical to that used by the other peripheral processors (that is, Ψ and $\mathbf{x}_b$ are identical for all peripheral processors, but their pseudo-random number generators are seeded independently). Each peripheral processor then evaluates the objective function at its generated trial point and sends the results (the trial point and the objective function value) to the master processor.[6]

Figure 1 presents a flowchart of the parallelized annealing process. Notice that the K trial points are generated and evaluated *in parallel*, but examined *sequentially* by the master processor. During each iteration, the master processor examines trial points until one of two events occurs: one of the trial points is accepted, or all K trial points have been rejected. The definitions of the occupation density and the equilibrium density given above need no modification when the trial points are examined in this manner. Notice also that, for $K = 1$, the parallel annealing process is identical to the single-processing case. In fact, the algorithm can be implemented on a *single* processor where, during each iteration, not one but K steps are generated – any gain in efficiency is then due to the more accurate statistics. Sections 2.1, 2.2, and 2.3 present the methods by which step generation and control of β are made *adaptive*. In each section, these methods are developed for the case $K = 1$.

2.1. EQUILIBRATION MEASURES

Although the equilibrium density $\underline{\pi}(\mathbf{x}; \beta)$ is independent of the generator, the generator is crucial in determining the *rate* at which the occupation density approaches $\underline{\pi}(\mathbf{x}; \beta)$. The generator is here designed to maximize the rate of equilibration. The rate of equilibration is defined as the rate of decrease of a scalar measure of the distance between the current occupation density $p_i(\mathbf{x})$ and the equilibrium density corresponding to the current value of β, $\underline{\pi}(\mathbf{x}; \beta_i)$. The *distance*, $D[p', p'']$, between two occupation densities p' and p'' is taken to be

$$D[p', p''] \doteq \frac{1}{2} \int_{\Omega} |p'(\mathbf{y}) - p''(\mathbf{y})| \, \mathrm{d}V_y. \tag{6}$$

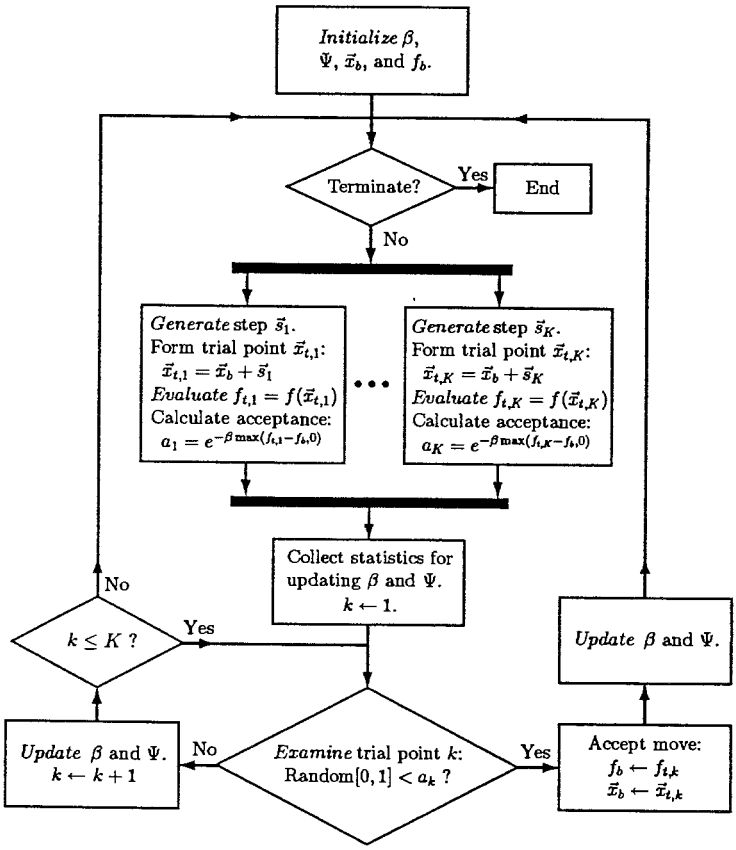

Fig. 1. Flowchart of the parallel simulated annealing process.

If p' and p'' are identical, $D[p', p'']$ is zero; if they do not overlap at all, $D[p', p'']$ is unity.[7]

A measure of the rate at which a process at equilibrium re-equilibrates after the perturbation caused by a change in β may now be defined. Consider an annealing process that is at equilibrium with the acceptance parameter equal to β after iteration i, i.e. $p_i(\mathbf{x}) = \underline{\pi}(\mathbf{x}; \beta)$. If the acceptance parameter is raised before iteration $i+1$ to $\beta + \Delta\beta$, the process is no longer at equilibrium, but at a distance $D_i^* = D[p_i, \underline{\pi}(\beta + \Delta\beta)]$ from equilibrium. After iteration $i+1$, the occupation density $p_{i+1}(\mathbf{x})$ follows from equation (2) and the distance to equilibrium is reduced to $D_{i+1} = D[p_{i+1}, \underline{\pi}(\beta + \Delta\beta)]$. The ratio of the distances to equilibrium before and after iteration $i+1$, D_{i+1}/D_i^*, is a measure of the rate of re-equilibration. The *reduction* R' is defined as the limit of this ratio as $\Delta\beta \to 0$:

$$R'(\beta, \Psi) = \lim_{\Delta\beta \to 0} \frac{D[p_{i+1}, \underline{\pi}(\beta + \Delta\beta)]}{D[p_i, \underline{\pi}(\beta + \Delta\beta)]}, \tag{7}$$

where, recall, $p_i(\mathbf{x}) = \underline{\pi}(\mathbf{x};\beta)$. The complement of the reduction, $1 - R'$, is adopted as the measure of the rate at which the process is approaching equilibrium. That is, the *equilibrium rate* R is defined by

$$R(\beta,\Psi) \doteq 1 - R'(\beta,\Psi). \tag{8}$$

The form of $p_{i+1}(\mathbf{x})$ is found by taking $p_i(\mathbf{x}) = \underline{\pi}(\mathbf{x};\beta)$ in equation (2):

$$\begin{aligned} p_{i+1}(\mathbf{y}) &= \underline{\pi}(\mathbf{y};\beta) + \int_\Omega [\underline{\pi}(\mathbf{x};\beta)a(\mathbf{y},\mathbf{x};\beta+\Delta\beta)g(\mathbf{y},\mathbf{x};\Psi) \\ &\quad -\underline{\pi}(\mathbf{y};\beta)a(\mathbf{x},\mathbf{y};\beta+\Delta\beta)g(\mathbf{x},\mathbf{y};\Psi)]\,\mathrm{d}V_y. \end{aligned} \tag{9}$$

Since the limit $\Delta\beta \to 0$ is being considered, it is sufficient to approximate to first order in $\Delta\beta$, and the acceptor defined in equation (1) is then found to satisfy

$$a(\mathbf{x},\mathbf{y};\beta+\Delta\beta) \approx a(\mathbf{x},\mathbf{y};\beta)\{1 - \Delta\beta \max[f(\mathbf{x}) - f(\mathbf{y}), 0]\}. \tag{10}$$

Similarly, the new equilibrium density is given by

$$\begin{aligned} \underline{\pi}(\mathbf{x};\beta+\Delta\beta) &= \alpha(\beta+\Delta\beta)e^{-(\beta+\Delta\beta)f(\mathbf{x})} \\ &\approx \underline{\pi}(\mathbf{x};\beta)\{1 - \Delta\beta[f(\mathbf{x}) - \overline{f}(\beta)]\}, \end{aligned} \tag{11}$$

where $\overline{f}(\beta)$ is the mean value of f at equilibrium with acceptance parameter β:

$$\overline{f}(\beta) \doteq \int_\Omega f(\mathbf{x})\underline{\pi}(\mathbf{x};\beta)\,\mathrm{d}V_x. \tag{12}$$

Since $g(\mathbf{x},\mathbf{y};\Psi) \equiv g(\mathbf{y},\mathbf{x};\Psi)$ and $\underline{\pi}(\mathbf{x};\beta)a(\mathbf{y},\mathbf{x};\beta) = \underline{\pi}(\mathbf{y};\beta)a(\mathbf{x},\mathbf{y},\beta)$, it follows that the equilibration rate satisfies

$$\begin{aligned} R(\beta,\Psi) &= 1 - \lim_{\Delta\beta\to 0} \\ &\qquad \frac{\Delta\beta\left\{\int_\Omega \underline{\pi}(\mathbf{y};\beta)|f(\mathbf{y}) - \overline{f}(\beta) + F(\mathbf{y};\beta,\Psi)|\,\mathrm{d}V_y + O(\Delta\beta)\right\}}{\Delta\beta\left\{\int_\Omega \underline{\pi}(\mathbf{y};\beta)|f(\mathbf{y}) - \overline{f}(\beta)|\,\mathrm{d}V_y + O(\Delta\beta)\right\}} \\ &= 1 - \frac{1}{S(\beta)}\int_\Omega \underline{\pi}(\mathbf{y};\beta)|f(\mathbf{y}) - \overline{f}(\beta) + F(\mathbf{y};\beta\Psi)|\,\mathrm{d}V_y. \end{aligned} \tag{13}$$

In equation (13), $F(\mathbf{y};\beta,\Psi)$ is defined by

$$F(\mathbf{y};\beta,\Psi) \doteq \int_\Omega [f(\mathbf{x}) - f(\mathbf{y})]a(\mathbf{x},\mathbf{y};\beta)g(\mathbf{x},\mathbf{y};\Psi)\,\mathrm{d}V_x, \tag{14}$$

and corresponds to the expected change in f when attempting a move from base point $\mathbf{y}$, and the *sensitivity* $S(\beta)$ is defined by

$$S(\beta) \doteq \int_\Omega \underline{\pi}(\mathbf{y};\beta)|f(\mathbf{y}) - \overline{f}(\beta)|\,\mathrm{d}V_y. \tag{15}$$

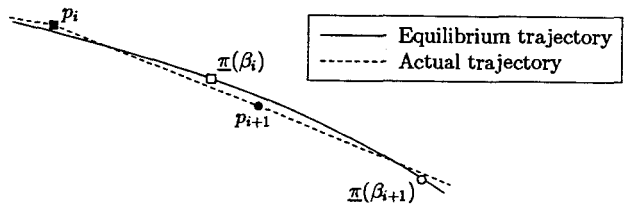

Fig. 2. Actual and equilibrium trajectories.

The sensitivity is a measure of the effect of a perturbation of the acceptance parameter:

$$D[\underline{\pi}(\beta, \underline{\pi}(\beta + \Delta\beta)] = \Delta\beta S(\beta) + O((\Delta\beta)^2), \tag{16}$$

and the equilibration rate is a measure of the rate of "recovery."

In this definition of the equilibration rate, it is assumed that the process is initially at equilibrium [i.e. $p_i(\mathbf{x}) = \underline{\pi}(\mathbf{x}; \beta)$]. Although this is unrealistic, a more accurate model of the behavior of the annealing process can be developed by introducing the concept of *trajectories* in the infinite-dimensional state space of occupation densities, as depicted in Figure 2. The *equilibrium* trajectory for an annealing process consists of the path on which the process would pass if it were kept at equilibrium at each stage – that is, if β were raised infinitely slowly. The *actual* trajectory, for which β is not raised infinitely slowly, consists of the sequence of points in this state space through which the process actually passes. Since the value of β is slowly and continually increasing, it seems reasonable to assume – at least locally – that (i) the equilibrium trajectory can be approximated linearly, as in equation (11), and (ii) that the actual trajectory and the equilibrium trajectory are roughly collinear:

$$p_i(\mathbf{y}) - \underline{\pi}(\mathbf{y}; \beta + \Delta\beta) \approx \gamma[\underline{\pi}(\mathbf{y}; \beta) - \underline{\pi}(\mathbf{y}; \beta + \Delta\beta)], \tag{17}$$

for all $\mathbf{y}$ and for some value of the constant γ. From equation (11) it now follows that

$$p_i(\mathbf{y}) - \underline{\pi}(\mathbf{y}; \beta + \Delta\beta) \approx \gamma\Delta\beta\underline{\pi}(\mathbf{y}; \beta)\{f(\mathbf{y}) - \overline{f}(\beta)\}, \tag{18}$$

and, from equations (9) and (10), it follows that

$$p_{i+1}(\mathbf{y}) - \underline{\pi}(\mathbf{y}; \beta + \Delta\beta) \approx \gamma\Delta\beta\underline{\pi}(\mathbf{y}; \beta)\{f(\mathbf{y}) - \overline{f}(\beta) + F(\mathbf{y}; \beta, \Psi)\}. \tag{19}$$

With this more realistic model, the reduction in the distance to equilibrium per examination is now seen to be identical to the earlier result what was derived more simplistically.

2.2. STEP GENERATION

The generator's parameter set is here chosen to maximize the equilibration rate. Since it seems impractical to design a generator that truly maximizes R for an

arbitrary objective function and for all β, it is appropriate to adopt a certain parameterized form for g and to choose values for the parameter set Ψ that are consistent with the criterion of maximizing R. We have chosen a Gaussian form for g because it is fairly simple to implement an approximately Gaussian generator. The general form of a Gaussian generator with positive-definite covariance matrix $\mathbf{C}$ is

$$g(\mathbf{y}, \mathbf{x}; \mathbf{C}) = \frac{1}{\sqrt{(2\pi)^N \det \mathbf{C}}} e^{-\frac{1}{2}(\mathbf{y}-\mathbf{x})\cdot\mathbf{C}^{-1}(\mathbf{y}-\mathbf{x})}. \tag{20}$$

In this case, the generator's parameter set Ψ consists of the elements of $\mathbf{C}$.

2.2.1. *Simple quadratic model*

In the adaptive updating of the control parameters (β and Ψ), ASA depends upon the statistical estimation of several quantities. A simple appreciation of the behavior of the simulated annealing process follows upon an *analytical* calculation of the dependence of various statistics upon the control parameters for an N-dimensional unconstrained quadratic objective function:

$$\hat{f}_N(\mathbf{x}) = \sum_{i=1}^{N} x_i^2 = \mathbf{x} \cdot \mathbf{x}, \tag{21}$$

and an isotropic Gaussian generator. Since, by design, the behavior of ASA is invariant under *linear* transformations of both the objective function and the coordinates, the results derived here are valid for a general unconstrained positive-definite quadratic provided the Gaussian generator is chosen in the only consistent fashion: its covariance matrix must be proportional to the inverse Hessian matrix of the quadratic. This analysis provides approximate values for many of the control parameters.

Since the quadratic is unimodal, it is of little practical value as a test case for global minimization. However, over the accessible domain at a given value of β, a typical objective function may often be approximated coarsely by a quadratic. This model provides analytical results that approximate the behavior of the annealing process and provide a measure of the dependence of the algorithm's performance upon dimensionality. It is convenient here to relate the radial variance of the generator, say σ^2, to the radial variance of the equilibrium density, $\sigma_r^2(\beta)$, by using the *relative step size* μ:[8]

$$\sigma^2 = \frac{\mu^2 \sigma_r^2(\beta)}{N} = \frac{\mu^2}{N} \frac{\int_0^\infty r^2 e^{-\beta r^2} r^{N-1}\, dr}{\int_0^\infty e^{-\beta r^2} r^{N-1}\, dr} = \frac{\mu^2}{2\beta}, \tag{22}$$

so that the generator can be written as

$$g(\mathbf{x}+\mathbf{s}, \mathbf{x}; \mu) = \left(\frac{N\beta}{\pi\mu^2}\right)^{N/2} e^{-N\beta \mathbf{s}\cdot\mathbf{s}/\mu^2}. \tag{23}$$

In this case, the generator parameter set Ψ contains only one member: the relative step size μ. In the following analysis, it is convenient to consider the inverse of the covariance matrix of the recently accepted steps as a provisional, relative metric for the coordinate space. In this way, at any fixed value of β, it becomes meaningful to speak of relatively "large" and "small" steps. In general, the metric form changes with β although, due to the invariance of ASA under scale transformations, the form of this metric is essentially fixed in the case of the quadratic.

Entities evaluated for the N-dimensional quadratic are written here with hats ($\hat{}$) and carry a subscripted N. So, for example, $\hat{\bar{f}}_N$ is found to satisfy

$$\hat{\bar{f}}_N(\beta) = \int_\Omega \hat{f}_N(\mathbf{x}) \hat{\underline{\pi}}_N(\mathbf{x};\beta)\, \mathrm{d}V_x = \frac{N}{2\beta}, \tag{24}$$

and $\hat{S}_N$ is given by

$$\hat{S}_N(\beta) = \int_\Omega \hat{\underline{\pi}}_N(\mathbf{y};\beta) \left| \hat{f}_N(\mathbf{y}) - \hat{\bar{f}}_N(\beta) \right| \mathrm{d}V_y = \frac{\left(\frac{N}{2e}\right)^{N/2}}{\beta\Gamma\left(\frac{N}{2}\right)} \approx \frac{\sqrt{\frac{N}{4\pi}}}{\beta}. \tag{25}$$

The integrals here are evaluated by adopting N-dimensional spherical polar coordinates, and the same can be done to find a simplified expression for $\hat{R}_N$. As shown in the Appendix, $\hat{R}_N$ is independent of β and is written here as $\hat{R}_N(\mu)$, and this function is plotted in Figure 3 for several values of N. For each value of N, $\hat{R}_N$ approaches zero as $\mu \to 0$ – smaller steps, although very likely to be accepted, do not allow rapid equilibration. The equilibration rate also approaches zero as $\mu \to \infty$ – larger steps are nearly always rejected, and also give slow equilibration. For each value of N, there is a value of μ, $\mu_{opt}(N)$, that maximizes $\hat{R}_N$. $\hat{R}_N(\mu_{opt}(N))$ and $\mu_{opt}(N)$ are shown in Figure 4 for a range of values of N. It is at $\mu = \mu_{opt}(N)$ that the step size is small enough to allow acceptance of a sufficiently large fraction of attempted moves, yet large enough that the accepted moves produce significant progress toward equilibrium. Notice that the maximum equilibration rate, $\hat{R}_N(\mu_{opt}(N))$, is a decreasing function of N and that the factor of N introduced into the definition of μ is responsible for the fact that $\mu_{opt}(N)$ is roughly constant for large N.

This analysis of the significance of step sizes in N dimensions provides a foundation for applying our heuristic in the step generation component of ASA. However, for objective functions containing anisotropic "valleys" that may not be aligned with the coordinate axes, a simple isotropic Gaussian generator (with $\mathbf{C}$ equal to a scalar multiple of the identity matrix) gives relatively slow equilibration; when optimally balanced in this form, steps in some directions are generally too short, while in other directions the steps are generally too long. Furthermore, as β increases during the annealing process, the characteristic widths and the orientations of the principal axes of the "valleys" to which the generator must conform may undergo significant change. The generator must adapt to such variations.

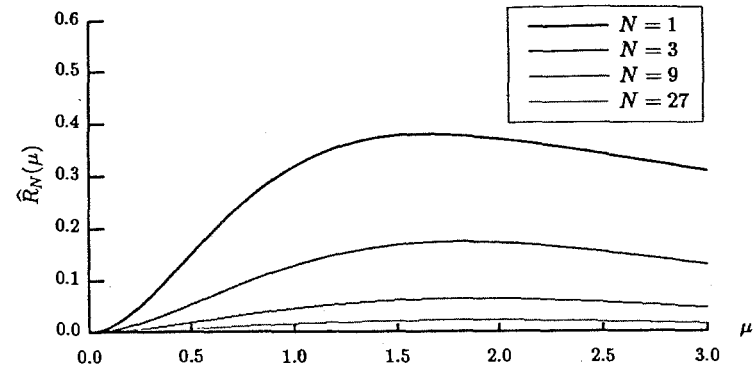

Fig. 3. Equilibration rate vs. relative step size for the quadratic.

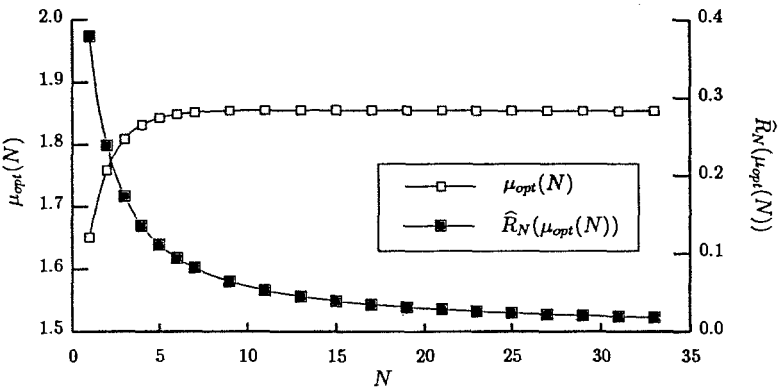

Fig. 4. Optimal relative step size and maximum equilibration rate vs. dimensionality for the quadratic.

2.2.2. *Archives and expansion factor*

It is possible to create a generator that *approximates* an N-dimensional Gaussian distribution with an appropriate covariance matrix by forming a randomly-weighted sum of several *previously accepted* steps:

$$\mathbf{s} = \frac{\rho}{\sqrt{M}} \sum_{m=1}^{M} w_m \mathbf{r}_m . \tag{26}$$

The set $\{\mathbf{r}_m : m = 1, 2, \ldots, M\}$ contains the M most recently accepted steps stored as a first-in-first-out queue referred to here as the *archives*; M is the *archive size*. The w's are independent pseudo-random numbers from a distribution with zero mean and unit variance. Since, according to the Central Limit Theorem, as $M \to \infty$, the generated steps assume a Gaussian distribution that has a covariance equal to ρ^2 times the covariance of the archives, ρ is called the *expansion factor*. By choosing M to be sufficiently large, it is possible to obtain a good approximation

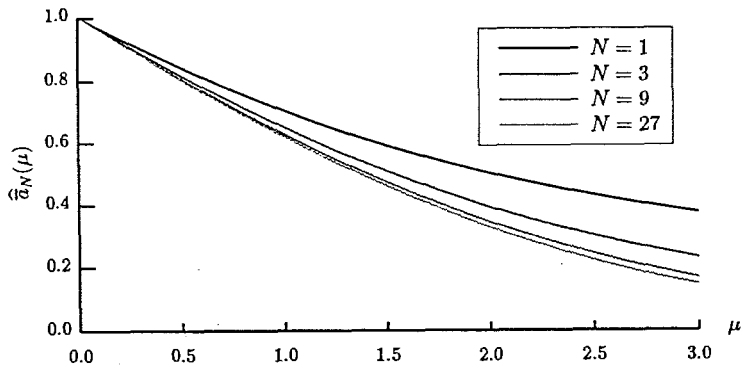

Fig. 5. Mean acceptance vs. relative step size for the quadratic.

to a Gaussian distribution. The archives are initially set equal to the differences between successive pairs of the elements of a set of $M + 1$ points chosen from a uniform random distribution over Ω.

Since the steps are generated by forming a sum of previously accepted steps, it is important to examine the characteristics of the accepted steps. The simplest of such characteristics is the *mean acceptance*, $\bar{a}(\beta, \Psi)$, which is just the fraction of attempted moves accepted at equilibrium:

$$\bar{a}(\beta, \Psi) = \int_{\Omega} \underline{\pi}(\mathbf{x}; \beta) \int_{\Omega} a(\mathbf{y}, \mathbf{x}; \beta) g(\mathbf{y}, \mathbf{x}, \Psi)\, \mathrm{d}V_y\, \mathrm{d}V_x. \tag{27}$$

The mean acceptance for the quadratic model is independent of β and, by again using N-dimensional spherical polar coordinates, is found to be given by

$$\hat{a}_N(\mu) = \begin{cases} \frac{2}{\pi}\left[\psi - \sin\psi\cos\psi \sum_{i=0}^{\frac{N-3}{2}} \frac{(2i)!!}{(2i+1)!!} \sin^{2i}\psi\right] & \text{(odd } N\text{)}, \\ 1 - \cos\psi\left[1 + \sum_{i=1}^{\frac{N}{2}-1} \frac{(2i-1)!!}{(2i)!!} \sin^{2i}\psi\right] & \text{(even } N\text{)}, \end{cases} \tag{28}$$

where $\psi = \tan^{-1}(2\sqrt{N}/\mu)$. Figure 5 shows $\hat{a}_N(\mu)$ vs. μ for several values of N. For smaller step sizes, $\hat{a}_N$ is close to unity and larger values of μ give values of $\hat{a}_N$ that approach zero. $\hat{a}_N(\mu_{opt}(N))$ is presented in Figure 6. For larger N, $\hat{a}_N(\mu_{opt}(N))$ is about 0.36; that is, for high-dimensional problems, equilibration for a quadratic objective function and a Gaussian generator is most rapid by our measure when only about 36% of the generated steps are accepted. This value contrasts with the assumption made in the design of several other variants of simulated annealing that the majority of the generated steps should be accepted in order to maintain efficiency.

Since the larger steps are generally more likely to be rejected, the accepted steps are typically "shorter" than the generated steps. To generate new steps that have a covariance that approximately maximizes R, the expansion factor ρ that appears in equation (26) must be taken to be greater than unity. Here, the expansion factor

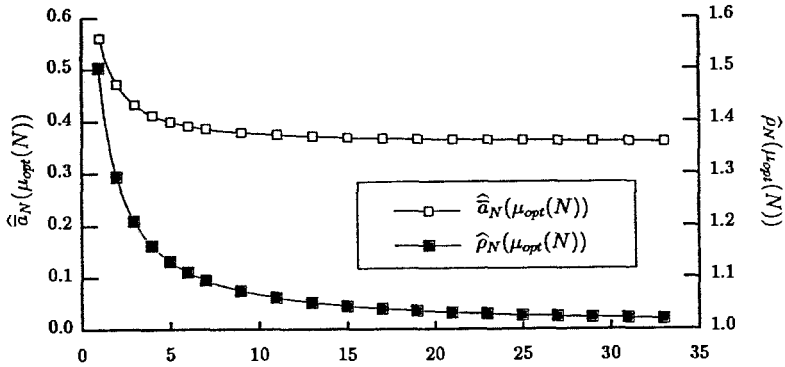

Fig. 6. Mean acceptance and expansion at optimal relative step size vs. dimensionality for the quadratic.

is the ratio of the RMS *generated step size* σ to the RMS *accepted* step size σ_a for the quadratic, where the accepted step size is defined by

$$\sigma_a^2(\beta,\mu) = \frac{1}{\hat{\bar{a}}_N(\beta,\mu)} \int_\Omega \pi(\mathbf{x};\beta) \int_\Omega a(\mathbf{y},\mathbf{x};\beta) g(\mathbf{y},\mathbf{x};\mu)(\mathbf{y}-\mathbf{x})(\mathbf{y}-\mathbf{x})\, \mathrm{d}V_y\, \mathrm{d}V_x. \tag{29}$$

It turns out that the ratio of the generated step size to the accepted step size, $\hat{\rho}_N$, is independent of β and is simply related to $\hat{\bar{a}}_N$ given in closed form in equation (28):

$$\hat{\rho}_N(\mu) = \frac{\sigma(\beta,\mu)}{\sigma_a(\beta,\mu)} = \sqrt{\frac{\mu^2}{2\beta\sigma_a^2(\beta,\mu)}} = \sqrt{\frac{\hat{\bar{a}}_N(\mu)}{\hat{\bar{a}}_{N+2}\left(\mu\sqrt{(N+2)/N}\right)}}. \tag{30}$$

For small μ, nearly all generated steps are accepted and therefore $\hat{\rho}_N$ is roughly unity. For larger μ, few of the larger generated steps are accepted, hence $\hat{\rho}_N$ is significantly greater than unity. In ASA, ρ is taken to be equal to $\hat{\rho}_N(\mu_{opt}(N))$ as a first step in the attempt to ensure *approximately* the proper scale for the generated steps. Figure 6 also shows $\hat{\rho}_N(\mu_{opt}(N))$ vs. N; these values for ρ generate steps with approximately the correct covariance.

2.2.3. *Inflation and boost factors*

There are two other components to the adaptive step generation scheme in ASA. The first of these depends upon the archive size, M. Clearly, M must be at least N in order that the generated steps span the space. In fact, for the generated step distribution to be roughly Gaussian, it can be expected that M should be at least several times N. On the other hand, if M becomes too large step generation

becomes costly and, since the value of β is changing, the archives may become largely "outdated". We have found that an archive size of

$$\boxed{M \sim 10N} \tag{31}$$

is generally appropriate. (A box is used here to distinguish any result that enters directly into the algorithm.) With any finite choice of M, the steps generated according to equation (26) tend to be "shorter" than optimal, due to the feedback: the "short" generated steps tend to be accepted and placed into the archives, causing subsequently generated steps also to be "shorter". It is therefore appropriate to introduce to the step generation process a compensating scale factor ν, called the *inflation factor*:

$$\mathbf{s} = \frac{\rho\nu}{\sqrt{M}} \sum_{m=1}^{M} w_m \mathbf{r}_m . \tag{32}$$

The optimal inflation factor for given N and M has been determined empirically by performing the annealing process on the unconstrained quadratic for fixed β and with $\rho = \hat{\rho}_N(\mu_{opt}(N))$. The value of ν is then adjusted to obtain approximately the desired statistics (i.e. $\overline{a}$, σ, and σ_a). In this way, an appropriate value of the inflation factor is bound to be given by

$$\boxed{\nu(M,N) \approx 1 + \frac{0.5 - 0.3N^{-2}}{M/N}} . \tag{33}$$

It is not essential to determine the optimal inflation factor precisely since a third, fine-scale mechanism is now proposed to complete the adaptive control of step size.

Both the expansion factor and the inflation factor are based upon analysis of the quadratic model and are fixed during the ASA process. They provide approximately the proper scale between the generated steps and the accepted steps. In practice, however, these fixed parameters alone are not always adequate to maintain an efficient adaptive step generator, so an *adaptive* parameter ξ, called the *boost factor*, is also introduced. The final prescription for step generation follows upon a slight modification of equation (32):

$$\boxed{\mathbf{s} = \frac{\rho\nu\xi}{\sqrt{M}} \sum_{m=1}^{M} w_m \underline{\mathbf{r}}_m} . \tag{34}$$

Before it is archived, each previously accepted step $\mathbf{r}_m$ is now divided by the value of the boost factor at the time that $\mathbf{r}_m$ was generated, and the result is written here as $\underline{\mathbf{r}}_m$. This normalization ensures that all steps in the archives contribute roughly equally to the generation of new steps even when the boost is changing during the time needed for M steps to be accepted (the "archive turn-over time").

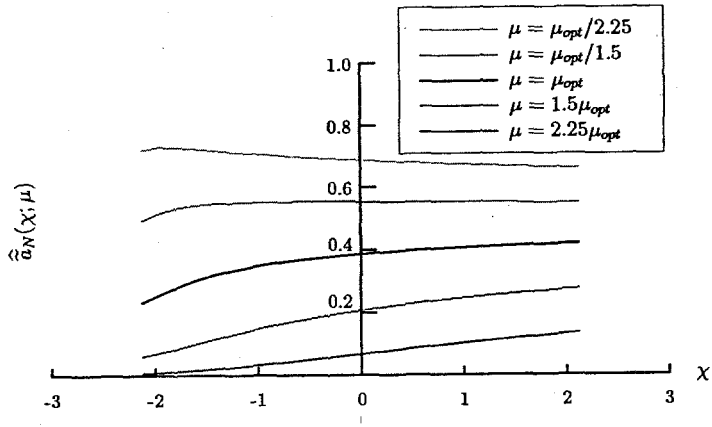

Fig. 7. $\hat{\bar{a}}_N(\chi;\mu)$ vs. χ, $N = 9$.

This final component of the step generation scheme is necessary to ensure efficiency when the objective function is radically different from a quadratic over the region of exploration. Without this dynamic step-size factor, there are cases in which the step generation process may stagnate – nearly all steps are rejected. Although demand upon the boost factor control scheme is minimized by the choice of appropriate values for ρ and ν, the control of ξ is the most complicated part of ASA. Here, we first consider the adaptive control of ξ in a single-processing environment; this method is generalized for the parallel-processing environment in Section 3.1.

The control of ξ follows upon examination of the statistics of the *accepted* moves. A simple scheme for controlling ξ consists of examining $\bar{a}$ and adjusting ξ after each iteration in order to keep $\bar{a}$ within a specified tolerance of a target value, say $\hat{\bar{a}}_N(\mu_{opt}(N))$. However, in ASA, ξ is controlled by examining the correlations between two statistics: (i) the *mean local acceptance* (i.e. the mean probability of accepting a move *from the current base point*), $\tilde{a}$, defined by

$$\tilde{a}(\mathbf{x};\beta,\Psi) \doteq \int_{\Omega} a(\mathbf{y},\mathbf{x};\beta)g(\mathbf{y},\mathbf{x};\Psi)\,\mathrm{d}V_y, \tag{35}$$

and (ii) the scaled deviation of f from $\bar{f}(\beta)$, χ, defined by

$$\chi(\mathbf{x};\beta) \doteq \frac{f(\mathbf{x}) - \bar{f}(\beta)}{\sigma_f(\beta)}. \tag{36}$$

Here, $\sigma_f(\beta)$ represents the standard deviation of f at equilibrium with acceptance parameter β:

$$\sigma_f(\beta) \doteq \left\{\int_{\Omega} [f(\mathbf{x}) - \bar{f}(\beta)]^2 \underline{\pi}(\mathbf{x};\beta)\,\mathrm{d}V_x\right\}^{1/2}. \tag{37}$$

For the quadratic, $\hat{\sigma}_{f,N}(\beta)$ is equal to $[N/(2\beta^2)]^{1/2}$. Figure 7 shows $\hat{\tilde{a}}_N(\chi;\mu)$ for a nine-dimensional quadratic and for several values of μ (the plots are of similar form for other values of N). It is found that when the step size is close to optimal, this curve is approximately linear with a weak positive gradient for values of χ that are within the predominant lobe of the equilibrium density (roughly $|\chi| < 1$). That is, when the step size is optimal, steps from the higher points are more likely to be accepted than are steps from the lower points – just the opposite of the situation when the step size is too small. This is a suggestive and useful observation.

The relationship between $\tilde{a}$ and χ may be approximated linearly in the form

$$\tilde{a}(\mathbf{x};\beta,\Psi) \approx A\chi(\mathbf{x};\beta) + \overline{a}(\beta,\Psi), \tag{38}$$

where an appropriate value for the constant A can be determined by performing a least-squares fit:

$$\begin{aligned} A &= \frac{\int_\Omega \underline{\pi}(\mathbf{x};\beta)\chi(\mathbf{x};\beta)\tilde{a}(\mathbf{x};\beta,\Psi)\,\mathrm{d}V_x - \overline{a}(\beta,\Psi)\int_\Omega \underline{\pi}(\mathbf{x};\beta)\chi(\mathbf{x};\beta)\,\mathrm{d}V_x}{\int_\Omega \underline{\pi}(\mathbf{x};\beta)\chi^2(\mathbf{x};\beta)\,\mathrm{d}V_x} \\ &= \frac{\overline{\chi\tilde{a}} - \overline{\chi}\,\overline{a}}{\overline{\chi^2}}. \end{aligned} \tag{39}$$

The ratio of A to $\overline{a}$ is written here as

$$\eta(\beta,\Psi) \doteq \frac{A(\beta,\Psi)}{\overline{a}(\beta,\Psi)}. \tag{40}$$

The value of η depends sensitively upon the step size: as the step size increases, $\overline{a}$ decreases and A is found to increase (see Figure 7). It turns out that this ratio for the quadratic, $\hat{\eta}_N$, satisfies

$$\begin{aligned} \hat{\eta}_N(\mu) = \sqrt{\frac{N}{2}}\frac{1}{\hat{\tilde{a}}_N(\mu)}\Bigg[&\frac{1}{2}\frac{\mu^2}{N}\hat{\tilde{a}}_{N+2}\left(\mu\sqrt{(N+2)/N}\right) \\ &- \frac{\Gamma\left(\frac{N+1}{2}\right)}{\sqrt{\pi}\Gamma\left(\frac{N+2}{2}\right)}\frac{\mu}{2\sqrt{N}}\left(\frac{4}{\mu^2/N+4}\right)^{\frac{N+1}{2}}\Bigg]. \end{aligned} \tag{41}$$

Figure 8 shows $\hat{\eta}_N(\mu)$ vs. μ, and Figure 9 shows $\hat{\eta}_N(\mu_{opt}(N))$.

It is proposed that the boost factor be adjusted in order to maintain an approximately optimal value of η. In ASA, A and $\overline{a}$ are estimated and ξ is adjusted in order to keep η within a tolerance factor δ (say about 0.8) of the target value from equation (41):

$$\hat{\eta}_N(\mu_{opt}(N))\delta < \eta < \frac{\hat{\eta}(\mu_{opt}(N))}{\delta}. \tag{42}$$

For a typical objective function, it can be expected that, within the operating region, η is generally a monotonically increasing function of step size – raising the step

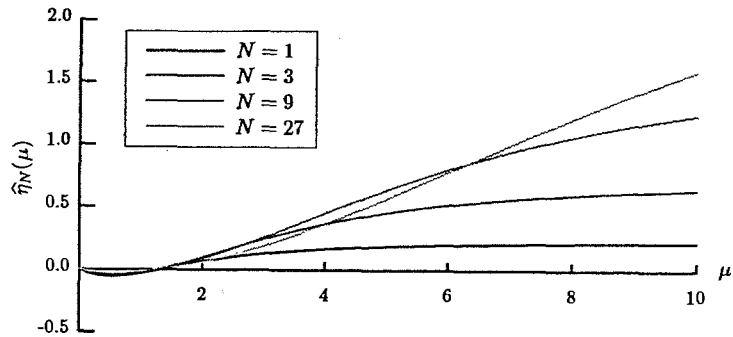

Fig. 8. $\hat{\eta}_N$ vs. relative step size for the quadratic

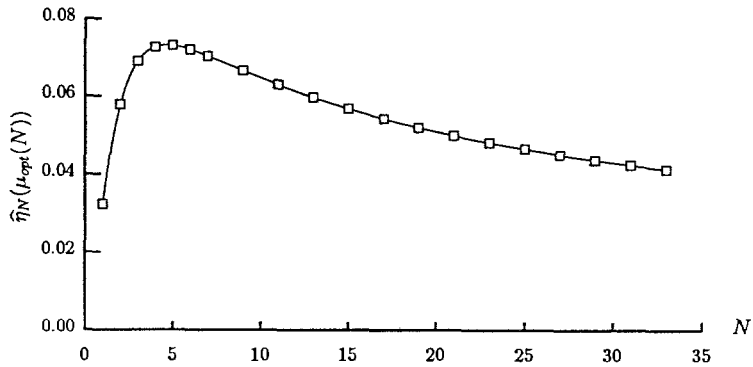

Fig. 9. $\hat{\eta}_N$ at optimal relative step size vs. dimensionality for the quadratic.

size typically lowers the mean acceptance $\overline{a}$ and gives and even greater reduction in the mean local acceptance $\tilde{a}$ for points where the objective function is below the mean $\overline{f}(\beta)$. Accordingly, after each iteration of the ASA process, η is estimated and the boost factor is adjusted as follows:

$$\boxed{\xi \leftarrow \begin{cases} \xi\theta^{\overline{a}}, & \text{if } \eta < \hat{\eta}_N(\mu_{opt}(N))\delta \\ \xi/\theta, & \text{if } \eta > \hat{\eta}_N(\mu_{opt}(N))/\delta \\ \xi, & \text{otherwise} \end{cases}}, \tag{43}$$

where θ is the *boost change factor* ($\theta > 1$). Notice that ξ may increase only by a factor of $\theta^{\overline{a}}$. This feature prevents the process from raising ξ quickly when $\overline{a}$ is small. It is proposed here that the value of θ be related to the expected change in β (discussed in Section 2.3), and this final detail is therefore discussed in Section 3.2.

2.2.4. *Inequality constraints*

In most other variants of simulated annealing, inequality constraints are treated simplistically: if a generated trial point lies outside the region of interest, it is dis-

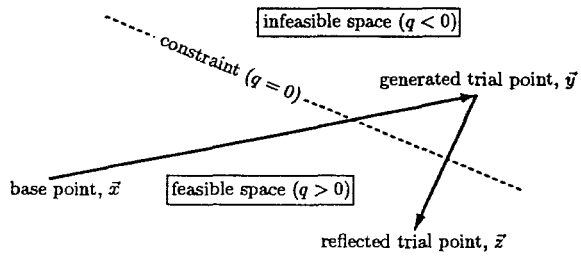

Fig. 10. Reflecting an infeasible step.

carded and the step generation process is repeated until a feasible point is obtained. In many problems it is imperative to consider more carefully the influence of constraints upon the problem at hand. During the early stages of the annealing process, the generator must generate steps that cover the entire region of interest nearly uniformly in order to maintain quasi-equilibrium. Many of the generated steps are infeasible (especially for large N), and the step generation process becomes unworkably inefficient. Furthermore, discarding infeasible steps makes the generator asymmetric: consider the situation depicted in Figure 10. Because the probability of generating an infeasible trial point from base point $\mathbf{x}$ is not equal to the probability of generating an infeasible trial point from base point $\mathbf{z}$, the renormalization of g associated with rejecting infeasible trial points causes $g(\mathbf{x}, \mathbf{z}; \Psi)$ to be unequal to $g(\mathbf{z}, \mathbf{x}; \Psi)$. As a result, the equilibrium densities become dependent upon the generator: near constraints, the occupation densities are lower than those given by the expression in equation (4). This effect is even more pronounced in multidimensional "corners". As a result both the efficiency of the algorithm and the probability of locating the global minimum can be reduced significantly.

Instead of discarding an infeasible trial point, we use *reflection* to render the generated step feasible. This process of reflection maintains the symmetry of the generator because the probability of generating a step from base point $\mathbf{x}$ to an infeasible trial point $\mathbf{y}$ (that is then reflected to a feasible point $\mathbf{z}$) is equal to the probability of generating a step from base point $\mathbf{z}$ to an infeasible trial point that is then reflected to $\mathbf{x}$. Figure 10 depicts a situation in which an infeasible step $\mathbf{s}$ (with corresponding infeasible trial point $\mathbf{y}$) has been generated from base point $\mathbf{x}$. The reflection process is performed as follows:

1. Determine the scalar ω $(0 < \omega < 1)$ for which the point $\mathbf{x} + \omega\mathbf{s}$ lies on the constraint $q(\mathbf{x}) = 0$. For nonlinear constraints, ω is estimated using an iterative root-finding algorithm (the existence of a bracket ensures robustness). If $\mathbf{y}$ violates more than one constraint, reflection is performed about the constraint for which ω is smallest.

2. Evaluate $\mathbf{u} = \nabla q$, the gradient of the constraint at $\mathbf{x} + \omega\mathbf{s}$. The scalar ζ for which $\mathbf{y} + \zeta\mathbf{Cu}$ lies on the (linearized) constraint is then evaluated according to
$$\zeta = (\omega - 1)\frac{\mathbf{s} \cdot \mathbf{u}}{\mathbf{uCu}}. \tag{44}$$
Here $\mathbf{C}$ is the provisional metric, i.e. the covariance matrix of the archives.
3. Reflect the step from the constraint by adding $2\zeta\mathbf{Cu}$ to $\mathbf{y}$, yielding the new trial point $\mathbf{z}$.
$$\mathbf{z} = \mathbf{x} + \mathbf{s} + 2\zeta\mathbf{Cu} = \mathbf{y} + 2\zeta\mathbf{Cu}. \tag{45}$$

The reflection process maintains generator symmetry for a single linear constraint and approximately does so for nonlinear constraints. Reflection is repeated until either a feasible trial point is located or a specified maximum number of reflections has been attempted, in which case reflection is aborted and a new random step is generated.

2.3. THE ANNEALING SCHEDULE

In this section the adaptive control of β is described. The ASA heuristic requires that the distance to equilibrium be kept below a user-specified value, say ε, while β is raised. The maximum general rate of increase of β is an increasing function of ε – that is, β can increase more rapidly if the process is to be allowed to stray further from equilibrium. The appropriate change in β after each examination, $\Delta\beta$, is then equal to the largest change that is consistent with keeping the distance to equilibrium below ε.

Consider an annealing process with occupation density $p_i(\mathbf{x})$ after examination i, that has been carried out with acceptance parameter β; the distance to equilibrium is $D_i = D[p_i, \underline{\pi}(\beta)]$. By using equation (17), the value of γ can be related directly to this distance:
$$D_i \approx \frac{1}{2}\int_\Omega \underline{\pi}(\mathbf{x};\beta)|(\gamma - 1)\Delta\beta[f(\mathbf{x}) - \overline{f}(\beta)]|\,\mathrm{d}V_x \approx (\gamma - 1)\Delta\beta S(\beta). \tag{46}$$

If the acceptance parameter is raised to $\beta + \Delta\beta$ before examination $i+1$, the distance to equilibrium is raised to $D_i^+ = D[p_i, \underline{\pi}(\beta + \Delta\beta)]$. According to equation (18), this distance is approximately given by
$$D_i^+ \approx \frac{1}{2}\int_\Omega \underline{\pi}(\mathbf{x};\beta)|\gamma\Delta\beta[f(\mathbf{x}) - \overline{f}(\beta)]|\,\mathrm{d}V_x \approx \gamma\Delta\beta S(\beta). \tag{47}$$

If one examination is performed with acceptance parameter $\beta + \Delta\beta$, the distance to equilibrium is reduced by a factor of $1 - R$:
$$D_{i+1} \approx \gamma\Delta\beta[1 - R(\beta, \Psi)]S(\beta). \tag{48}$$

The value of $\Delta\beta$ that satisfies the ASA heuristic is the value for which $D_{i+1} = D_i$ – i.e. the system is in a steady state – and $D_i^+ = \varepsilon$. The value of γ follows from

this steady-state condition, i.e. $\gamma = 1/R(\beta, \Psi)$. With this, the requirement $D_i^+ = \varepsilon$ leads to the allowed change in β:

$$\boxed{\Delta\beta = \frac{\varepsilon R(\beta, \Psi)}{S(\beta)}}. \tag{49}$$

During the annealing process R and S are to be estimated statistically, so equation (49) completely determines the adaptive annealing schedule (without reference to the quadratic model).

Recall that the sensitivity is given by equation (15):

$$\boxed{S(\beta) = \overline{|f - \overline{f}|}}. \tag{50}$$

The equilibration rate is relatively difficult to estimate accurately by using the form given in equation (13). Empirical results suggest that the expected change in the value of f when attempting a move from base point $\mathbf{y}$, i.e. $F(\mathbf{y}; \beta, \Psi)$, is roughly proportional to $f(\mathbf{y}) - \overline{f}(\beta)$:

$$F(\mathbf{y}; \beta, \Psi) \approx -\phi(\beta, \Psi)[f(\mathbf{y}) - \overline{f}(\beta)]. \tag{51}$$

This property is quite intuitive: for base points $\mathbf{y}$ such that $f(\mathbf{y})$ is higher than the mean value, the change in f is generally *negative*, and, conversely, when stepping from points where the value of the objective function is below the mean, the change in f is generally *positive*, and it turns out that $F(\mathbf{y})$ is generally well approximated by the expression in equation (51). Equation (13) may now be rewritten:

$$\begin{aligned} R(\beta, \Psi) &\approx 1 - \frac{[1 - \phi(\beta, \Psi)] \int_\Omega \pi(\mathbf{y}; \beta) |f(\mathbf{y}) - \overline{f}(\beta)| \, \mathrm{d}V_y}{\int_\Omega \pi(\mathbf{y}; \beta) |f(\mathbf{y}) - \overline{f}(\beta)| \, \mathrm{d}V_y} \\ &= \phi(\beta, \Psi). \end{aligned} \tag{52}$$

That is, in order to estimate R, it is sufficient to determine the constant of proportionally between $f(\mathbf{y}) - \overline{f}(\beta)$ and $F(\mathbf{y}; \beta, \Psi)$. In ASA, this is done by performing a linear least-squares fit between the values of F and $f(\mathbf{y}) - \overline{f}$:

$$\begin{aligned} \phi(\beta, \Psi) &\approx \frac{-\overline{[f(\mathbf{y}) - \overline{f}(\beta)][F(\mathbf{y}; \beta, \Psi)]}}{\overline{[f(\mathbf{y}) - \overline{f}(\beta)]^2}} \\ &= -\overline{\left[\frac{f(\mathbf{y}) - \overline{f}(\beta)}{\sigma_f(\beta)}\right]\left[\frac{F(\mathbf{y}; \beta, \Psi)}{\sigma_f(\beta)}\right]} \Bigg/ \overline{\left[\frac{f(\mathbf{y}) - \overline{f}(\beta)}{\sigma_f(\beta)}\right]^2}. \end{aligned} \tag{53}$$

The division by σ_f essentially normalizes the values of F and $f(\mathbf{y}) - \overline{f}$, and thereby stabilizes these statistics against changes as β increases. In terms of the parameter introduced in equation (36), the estimate of the equilibration rate used in ASA can be written as

$$\boxed{R(\beta, \Psi) \approx -\overline{\chi F/\sigma_f}/\overline{\chi^2}}. \tag{54}$$

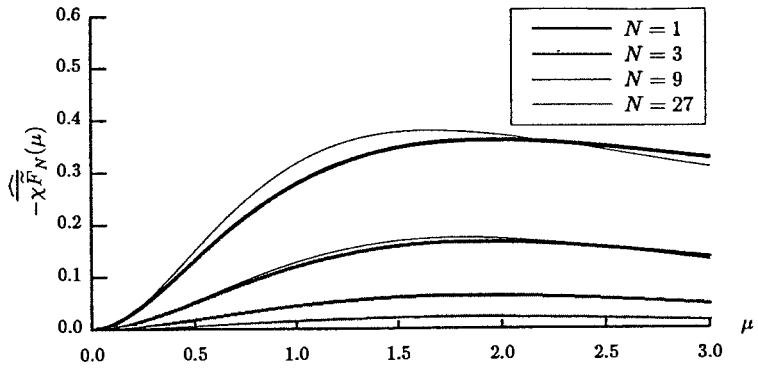

Fig. 11. Approximate equilibration rate vs. relative step size for the quadratic. The thin curves represent the exact values of the equilibration rate shown in Figure 3.

In principle, $\overline{\chi^2}$ is equal to unity, but we have found it better not to assume this during the statistical estimation of R. Figure 11 shows this approximation to R for the quadratic:

$$\begin{aligned}\hat{R}_N(\mu) = {} & \frac{\mu^2}{N}\hat{\bar{a}}_{N+2}\left(\mu\sqrt{(N+2)/N}\right) \\ & + \frac{N+2}{4}\left(\frac{\mu^2}{N}\right)^2 \hat{\bar{a}}_{N+4}\left(\mu\sqrt{(N+4)/N}\right) \\ & - \frac{\Gamma\left(\frac{N+3}{2}\right)}{\sqrt{\pi}\Gamma\left(\frac{N+2}{2}\right)}\left(\frac{\mu^2}{N}\right)^{3/2}\left(\frac{4}{\mu^2/N+4}\right)^{\frac{N+3}{2}}. \end{aligned} \tag{55}$$

It turns out that, for values of μ in the region of greatest interest (roughly $1 < \mu < 3$), the relative error of this approximation decreases as N increases.

3. Implementation of ASA

The conceptual foundation of ASA is now essentially complete. To this point, however, the calculation of the statistical averages that appear in equations such as equations (50) and (54) has not been addressed. These averages are an essential component of the proposed annealing schedule. Furthermore, there are modifications to enhance the performance when the algorithm is to be run on a multiprocessing system.

3.1. Step generation for the parallel-processing case

To this point, the generator has been adjusted to maximize the equilibration rate R. By definition, R is the equilibration rate "per examination" – that is, each time a trial point is examined, the distance to equilibrium is reduced by a factor of $1 - R$. This

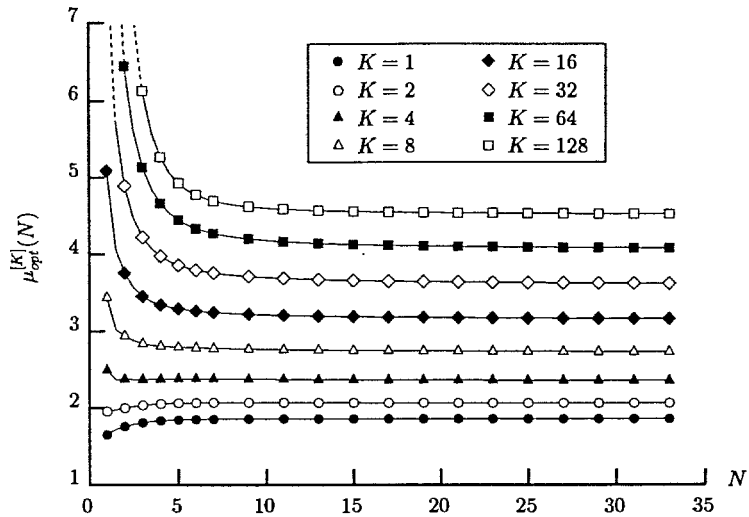

Fig. 12. Optimal relative step size vs. dimensionality for the quadratic.

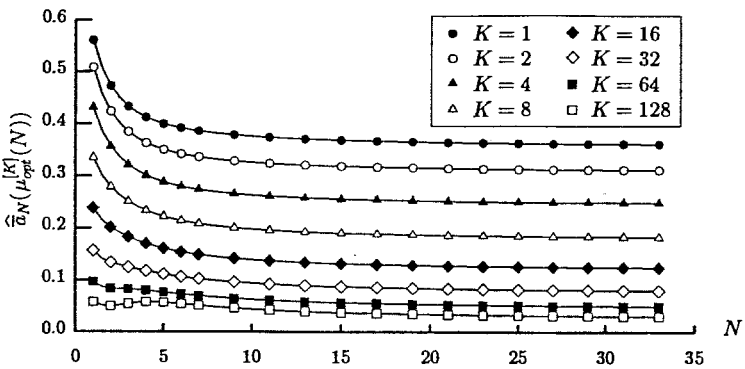

Fig. 13. Mean acceptance at optimal relative step size vs. dimensionality for the quadratic.

criterion for choosing the generator must be modified for the parallel-processing case, in which multiple examinations may occur during each iteration. Recall that, in the single-processor case, β is raised after each examination according to the expression in equation (49). The value of $\Delta\beta$ is chosen so that the distance moved along the equilibrium trajectory [i.e. $S(\beta)\Delta\beta$] is roughly equal to the distance that the actual density would be moved in a steady state by the next examination.

This simple picture suggests an appropriate generalization to the multiprocessing case: choose the step size to maximize the expected *total* distance moved *per iteration*. Given that $\Delta\beta S(\beta) = \varepsilon R$ is the distance moved along the trajectories *per examination*, the expected total distance moved per iteration using K processors, $E^{[K]}$, is given by

$$E^{[K]}(\beta, \Psi, \varepsilon) = \varepsilon R \cdot (\text{expected number of examinations per iteration})$$

TABLE I. Coefficients for fit of $\mu_{opt}^{[K]}(N)$ vs. K and N [see equation (58)]

c_{ij}	$j=0$	$j=1$	$j=2$	$j=3$	$j=4$	$j=5$	$j=6$
$i=0$	1.85076	0.232638	0.0676955	0.0251900	−0.0106925	0.00143886	−6.98653e-5
$i=1$	0.120293	0.181110	0.0234745	0.0423060	−0.00777322	−9.86733e-4	1.47574e-4
$i=2$	−0.784101	−0.323338	−0.243144	0.531403	−0.538065	0.162461	−0.0152431
$i=3$	−0.671876	4.55783	−15.3263	12.1604	−4.21906	0.933441	−0.0897940
$i=4$	3.17938	−27.1617	94.0994	−84.8508	37.8704	−8.98941	0.835983
$i=5$	−2.51663	50.8540	−170.304	159.915	−75.9218	17.9218	−1.64737
$i=6$	0.472807	−28.0532	91.9147	−87.8914	42.0015	−10.0425	0.918072

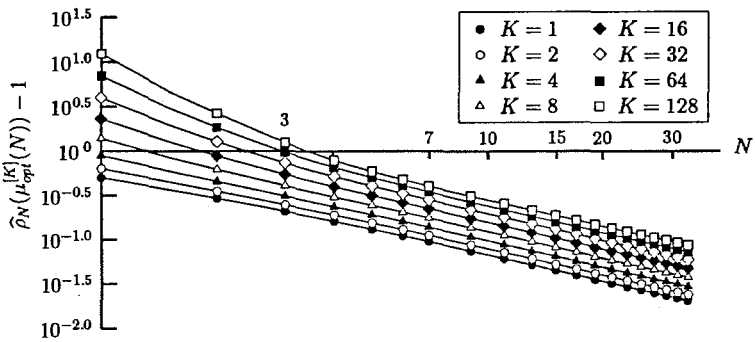

Fig. 14. Expansion −1 at optimal relative step size vs. dimensionality for the quadratic.

$$= \varepsilon R \left[\sum_{k=1}^{K} k\overline{a}(1-\overline{a})^{k-1} + K(1-\overline{a})^{K-1} \right]$$

$$= \frac{\varepsilon R}{\overline{a}} [1 - (1-\overline{a})^K]. \tag{56}$$

It is proposed that appropriate values of ρ and η be determined by finding the value of μ that maximizes this total distance for the quadratic, $\hat{E}_N^{[K]}(\mu, \varepsilon)$:

$$\hat{E}_N^{[K]}(\mu, \varepsilon) = \varepsilon \hat{R}_N(\mu) \frac{1}{\hat{\overline{a}}_N(\mu)} \left\{ 1 - [1 - \hat{\overline{a}}_N(\mu)]^K \right\}. \tag{57}$$

Figure 12 is a generalization of Figure 4 and shows these optimal values of μ, $\mu_{opt}^{[K]}(N)$, vs. N for $K = 1, 2, 4, \ldots, 128$. Evidently, the increase in the expected number of examinations more than counters the decrease in the equilibration rate associated with the larger step sizes. Since these results will be used in the implementation of ASA, it is useful to fit an algebraic form to $\mu_{opt}^{[K]}(N)$ vs. N:

$$\mu_{opt}^{[K]}(N) \approx \sum_{i=0}^{6} \sum_{j=0}^{6} c_{ij} (\ln K)^j N^{-i}, \tag{58}$$

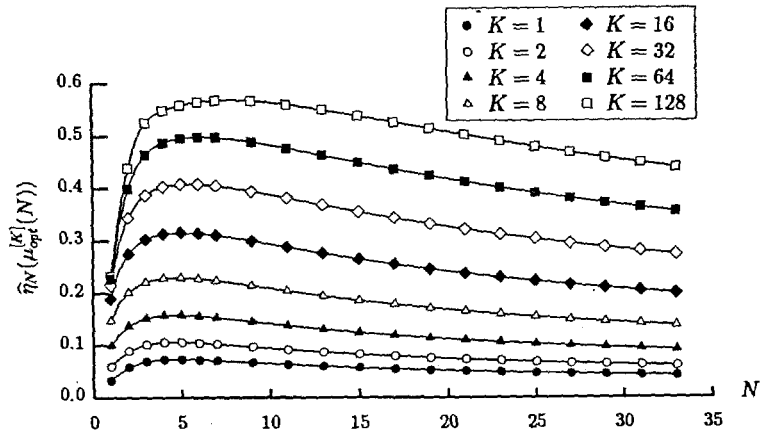

Fig. 15. $\hat{\eta}_N$ at optimal relative step size vs. dimensionality for the quadratic.

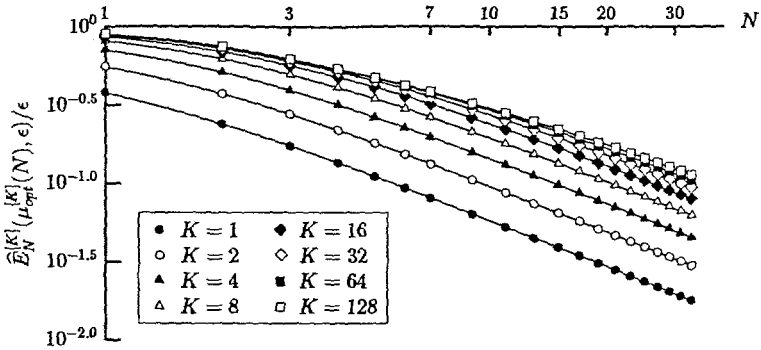

Fig. 16. Total relative distance moved along trajectories per iteration at optimal relative step size vs. dimensionality for the quadratic.

where the coefficients c_{ij} are listed in Table I. The associated estimates for $\mu_{opt}^{[K]}(N)$ are accurate to three significant figures for $1 \leqslant K \leqslant 128$ and $1 \leqslant N \leqslant 75$. (This form is found to be invalid for $K \gg 100$ – it turns out that, as $K \to \infty$, $\mu_{opt}^{[K]}(N)$ increases without bounds for both large and small N.) The resulting values of $\widetilde{a}$, ρ and η that are used in ASA are then found by using equations (28), (30), and (41), respectively, and are shown in Figures 13, 14 and 15 as functions of N and K. The value of the total *relative* distance moved (that is, the total distance moved, divided by ε) along the trajectories per iteration at the optimal relative step size, $\hat{E}_N^{[K]}(\mu_{opt}^{[K]}(N), \varepsilon)/\varepsilon$, is shown in Figure 16.[9] According to this measure, diminishing returns are obtained by raising K (e.g., doubling K does not double $\hat{E}$). Also notice that the total relative distance moved per iteration is asymptotically proportional to N^{-1}. This result, combined with the asymptotic expression for $\hat{S}_N$ given in equation (25), indicates that – according to equation (49) – the relative change in β per iteration for the quadratic is approximately proportional to $\varepsilon N^{-3/2}$. That is, the number of iterations required to raise β by a given relative amount

is approximately proportional to $N^{3/2}/\varepsilon$, giving an indication of the scaling of computational effort with dimensionality for an objective function that is roughly quadratic. Finally, notice that $\hat{\bar{a}}_N(\mu_{opt}^{[K]}(N))$ is a monotonically decreasing function of K (for example, for $K = 8$, $\hat{\bar{a}}$ is only about 0.2, meaning that about 80% of the examined trial points are rejected). Information from the rejected points, however, is used in the calculation of statistics: raising K allows the collection of generally more reliable statistics, and it is found in Section 4 that this advantage can more than offset the lack of a proportional increase in E.

3.2. The boost change factor

After each examination the boost factor is updated according to equation (43). An appropriate value for the boost change factor θ can now be determined by first considering the quadratic model: if the acceptance parameter is raised by a factor of $1 + \Delta\beta/\beta$, it is necessary to lower the *absolute* step size σ by a factor of $(1 + \Delta\beta/\beta)^{1/2}$ in order to maintain the same *relative* step size μ [see equation (22)]. The role of the boost factor ξ is analogous to the relative step size μ, so it seems reasonable that, for a quadratic, θ should be chosen to admit a similar relative change in ξ:

$$\hat{\theta}_N = \left[1 + \frac{\Delta\beta}{\beta}\right]^{1/2} = \left[1 + \frac{\varepsilon \hat{R}_N(\mu_{opt}^{[K]}(N))}{\beta \hat{S}_N(\beta)}\right]^{1/2}$$

$$\approx \left[1 + \varepsilon \hat{R}_N(\mu_{opt}^{[K]}(N))\sqrt{\frac{4\pi}{N}}\right]^{1/2}, \tag{59}$$

where the approximation given in equation (25) has been used. Here, $\hat{R}_N$ may be approximated as in equation (55). An examination of a number of non-quadratic objective functions indicated that $\hat{\theta}_N$ is a suitable value for the boost change factor factor in general. Given that the boost factor is not necessarily changed at every examination – but only as required – it is unnecessary to specify the boost change factor accurately: a different value of the boost change factor can be accommodated automatically by adjustment of the frequency of the changes in the boost.

3.3. Estimation of statistics

The modifications of β and Ψ that follow each examination performed by the master processor depend upon the statistics of the earlier behavior. With the assumption of ergodicity, it is possible to identify the ensemble averages [e.g. $\overline{f}$ in equation (12)] with averages over many examinations in a single run. This averaging is complicated by the fact that the underlying conditions are changing. Since fixing β and Ψ temporarily during annealing in order to determine more accurate statistics significantly reduces efficiency, the estimation process implemented in ASA

dynamically updates statistics by "folding" new data into the current statistics. For example, if $\overline{v}$ is an estimate of the mean of a quantity v, then a sample v_i is folded into $\overline{v}$ as follows:

$$\overline{v} \leftarrow \kappa v_i + (1 - \kappa)\overline{v}, \tag{60}$$

where κ is a *folding factor* of the form

$$\kappa = 1 - e^{-1/\tau}. \tag{61}$$

The constant τ is called the *lifetime* of the statistic. Statistics collected in this manner are simply weighted averages for which the weight decays exponentially.

In choosing statistical lifetimes, both accuracy and responsiveness must be considered: longer life-times provide greater accuracy if the underlying conditions are changing slowly but give inadequate responsiveness under rapidly changing conditions. Therefore, the statistics in ASA are separated into three classes, and two of these classes have *dynamic* lifetimes:

1. *Long-lifetime statistics:* $\overline{f}$, σ_f^2 and S. At equilibrium, these statistics are independent of the generator and depend only upon β. New data are folded into these statistics at every examination, and for maximum accuracy, the lifetime is made as long as possible while remaining consistent with the changing conditions. This long lifetime is denoted by τ'' and the corresponding folding factor is κ''. In ASA, τ'' is generally about one quarter of the number of examinations necessary for β to increase by a factor of e. If the relative change in β per examination, $\Delta\beta/\beta$, in constant and smaller than unity, then the number of examinations needed for β to increase by a factor of e, say c_e, is given by
$$c_e = 1/\ln\left(1 + \frac{\Delta\beta}{\beta}\right) \approx \frac{\beta}{\Delta\beta}. \tag{62}$$
The long lifetime must be shorter than c_e for the associated statistics to react adequately to changing conditions. Further, in order to provide stability, it is necessary to set a minimum value for the long lifetime, and this is found by analyzing the quadratic model: τ''_{min} is set equal to one-quarter of the number examinations needed for β to increase by a factor of e for the quadratic, $\hat{c}_e$:
$$\tau''_{min} = \frac{1}{4}\hat{c}_e \approx \frac{1}{4}\frac{\beta\hat{S}_N(\beta)}{\varepsilon\hat{R}_N(\mu_{opt}^{[K]}(N))} = \frac{1}{4}\frac{\left(\frac{N}{2e}\right)^{N/2}}{\varepsilon\hat{R}_N(\mu_{opt}^{[K]}(N))\Gamma\left(\frac{N}{2}\right)}, \tag{63}$$
where $\hat{R}_N(\mu_{opt}^{[K]}(N))$ can be approximated by the relation given in equation (55). Therefore, τ'' is updated in ASA after each iteration, as follows:
$$\tau'' \leftarrow \max\left[\tau''_{min}, 1/\left(4\left|\frac{\Delta\beta}{\beta}\right| + \frac{0.1}{\tau''_{min}}\right)\right]. \tag{64}$$
This form for the long lifetime gives values that are, in general, roughly $c_e/4$, but are constrained to lie between τ''_{min} and $10\tau''_{min}$.
2. *Short-lifetime statistics:* R, $\overline{a}$, and η. The values of these statistics are not independent of the generator and involve an additional complication since

they are defined in terms of averages over the steps attempted from each base point. Since these short-lifetime statistics are updated only upon *acceptance* of a move, it is appropriate that the short lifetime be smaller than τ'' by about a factor of $\overline{a}$. Further, when each sample is folded into the statistics, it is weighted in proportion to the total number of *examinations*, c, performed at that base point.

In ASA, the short lifetime, written as τ' with associated folding factor κ', is updated after each iteration as follows:

$$\tau' \leftarrow \max[\tau'_{min}, \overline{a}\tau''']. \tag{65}$$

As before, the minimum value provides stability, and is determined from quadratic model statistics:

$$\tau'_{min} = \hat{\tilde{a}}_N(\mu_{opt}^{[K]}(N))\tau''_{min}. \tag{66}$$

The collection of the short-lifetime statistics is less straightforward than the collection of the long-lifetime statistics. For example, the mean acceptance $\overline{a}$ is estimated as follows:

$$\overline{a} \approx \frac{\langle c[\tilde{a}]\rangle}{\langle c\rangle}, \tag{67}$$

where $[\tilde{a}]$ represents an estimate of $\tilde{a}(\mathbf{x}_b)$. Here, $[\tilde{a}]$ is taken to be the mean acceptance probability of the s steps *generated* from the current base point (recall that, at each iteration, K steps are generated, so s is an integer multiple of K):

$$[\tilde{a}] = \frac{1}{s}\sum_{k=1}^{s} a_k. \tag{68}$$

The expected value of the estimate in equation (67) is close to $\overline{a}$ provided that the value of the short lifetime is larger compared to the mean number of examinations between accepted moves (i.e. $\overline{a}\tau' \gg 1$). The prescription for choosing τ' generally ensures that this condition is met.

Since the averages involved in the least-squares fits that lead to equations (39) and (54) are not ensemble averages, the weights are chosen according to the accuracy of each data point, and R and η are estimated by

$$\boxed{R \approx \frac{-\overline{\chi F/\sigma_f}}{\overline{\chi^2}} \approx -\frac{\langle s\chi[F]/\sigma_f\rangle}{\langle s\chi^2\rangle}} \tag{69}$$

and

$$\boxed{\eta = \frac{\overline{\chi\tilde{a}} - \overline{\chi}\,\overline{a}}{\overline{a}\overline{\chi^2}} \approx \frac{\langle s\chi[\tilde{a}]\rangle\langle s\rangle - \langle s\chi\rangle\langle s[\tilde{a}]\rangle}{\langle s[\tilde{a}]\rangle\langle s\chi^2\rangle}}, \tag{70}$$

where $[F]$ represents an estimate of $F(\mathbf{x}_b)$: $[F]$ is taken to be the average of the product of the acceptance probabilities and the corresponding changes in f over the s steps generated from the current base point, and is given by

$$[F] = \frac{1}{s}\sum_{k=1}^{s} a_k(f_{t,k} - f_b). \tag{71}$$

Since $[\tilde{a}]$ and $[F]$ represent averages calculated for s steps generated from the current base point, they are called *local aggregates.*

The values of the short-lifetime statistics proper (written with single angle brackets) are updated only upon acceptance of a move; between accepted moves, estimates of these statistics, called *provisional* short-lifetime statistics and written with double angle brackets, are used to estimate R, $\overline{a}$, and η. In the notation of equation (60), these provisional statistics are updated before each examination, as follows:

$$\langle\langle v\rangle\rangle = \kappa' v_i + (1-\kappa')\langle v\rangle, \tag{72}$$

so, for example, a provisional value of $\langle c[\tilde{a}]\rangle$ is given by

$$\langle\langle c[\tilde{a}]\rangle\rangle = \kappa' c[\tilde{a}] + (1-\kappa')\langle c[\tilde{a}]\rangle. \tag{73}$$

The provisional statistics are then used to estimate R, $\overline{a}$, and η, as in equations (67), (69), and (70) above. In this way, the most recent estimates of $[\tilde{a}]$, $[F]$, $\overline{f}$ and σ_f can be used in the evaluation of χ, R and η. On acceptance of a move, all of the provisional short-lifetime statistics are simply copied into the short-lifetime statistics proper and the local aggregates, s, and c are all reset to zero.

3. *Archive covariance matrix:* **C**. All K trial points are examined during each iteration for archival of the corresponding steps. These examinations are performed separately from the examinations that determine the acceptance or rejection of the moves. As a result, it is possible for more than one step to be placed into the archives during a single iteration, thereby reducing the risk that the archives will become "outdated". An approximation to the covariance matrix of the archives is maintained for use during reflection. When a new step, say **s**, is added to the archives, **C** is updated as follows. First, the approximation to the *mean* of the archives, **v** is updated:

$$\mathbf{v} \leftarrow \kappa_a \mathbf{s} + (1-\kappa_a)\mathbf{v}, \tag{74}$$

and the archive covariance **C** is then updated according to

$$\mathbf{C} \leftarrow \kappa_a (\mathbf{s}-\mathbf{v})(\mathbf{s}-\mathbf{v})^T + (1-\kappa_a)\mathbf{C}. \tag{75}$$

Here the folding factor κ_a is taken to be $1 - e^{-1/M}$.

3.4. Summary of the ASA algorithm

The ASA algorithm can be summarized in detail as follows:

1. *Initialization:* The generator parameters ρ, ν, and η are chosen according to equations (30), (33), and (41), respectively, using the appropriate values of $\mu_{opt}^{[K]}(N)$ [given in equation (58)]. The base point $\mathbf{x}_b$ is initialized by choosing a random point from a uniform distribution over Ω. The acceptance parameter β is set to 0 and the boost factor ξ is set to 1. The archives are initialized by generating $M+1$ random points from a uniform distribution over Ω and the differences between successive pairs of these points are placed into the archives. Initial values for the archive mean **v** and the archive covariance matrix **C** are then calculated. The long-lifetime statistics ($\overline{f}$, σ_f^2, and S) are

initialized by evaluating f at τ''_{min} points chosen from a uniform distribution over Ω:

$$\overline{f} = \frac{1}{\tau''_{min}} \sum_{j=1}^{\tau''_{min}} f_j, \quad \sigma_f^2 = \left[\frac{1}{\tau''_{min}} \sum_{j=1}^{\tau''_{min}} f_j^2 \right] - \overline{f}^2$$

$$S \approx \sigma_f \frac{\hat{S}_N(\beta)}{\hat{\sigma}_{f,N}(\beta)} \approx \frac{\sigma_f}{\sqrt{2\pi}}.$$

[The final approximation here follows from equation (25) and the remark following equation (37).] The short-lifetime statistics are initialized:

$$\langle c \rangle = 1, \quad \langle c\chi \rangle = 0,$$
$$\langle s \rangle = K, \quad \langle s\chi \rangle = 0, \quad \langle s\chi^2 \rangle = K, \quad \langle s[\tilde{a}] \rangle = K, \quad \langle s\chi[\tilde{a}] \rangle = 0,$$
$$\langle s\chi[F]/\sigma_f \rangle = -K\hat{R}_N(\mu_{opt}^{[K]}(N)),$$

and the local aggregates, s, and c are initialized:

$$s = 0, \quad c = 0, \quad [\tilde{a}] = 0, \quad [F] = 0.$$

[Although, in principle, the initial equilibration rate is unity, the conservative value of $\hat{R}_N(\mu_{opt}^{[K]}(N))$ from the quadratic model gives greater stability at the outset.]

2. *Generation, reflection, and evaluation:* The master processor sends the current base point and function value $\mathbf{x}_b$ and f_b, boost factor ξ, and acceptance parameter β to each of the K peripheral processors. Each peripheral processor generates an independent step $\mathbf{s}_k$ according to equation (34) and adds it to the current base point $\mathbf{x}_b$ to form a trial point $\mathbf{x}_{t,k}$. If $\mathbf{x}_{t,k}$ is infeasible, $\mathbf{s}_k$ is reflected. Each peripheral processor then returns the new trial point, function value, and acceptance probability to the master processor.
3. *Examination, statistics processing, and parameter updating:* The master processor processes the trial points, function values and acceptance probabilities from the K peripheral processors as outlined in the pseudo-code listed in Figure 17.
4. *Termination criterion check:* If the (user-specified) termination criterion is not satisfied, return to step 2.

4. Examples

In this section the results of testing ASA on two objective functions are presented. A figure of merit is defined which measures the efficiency of the ASA algorithm as a function of the annealing rate parameter ε, and the number of parallel processors K.

4.1. Measuring efficiency

A generally useful measure of efficiency is difficult to define for global optimization algorithms. First, there is a variety of possible definitions of success in a given run.

for $k \leftarrow 1$ to K do begin

Update local aggregates and s:

$[\tilde{a}] \leftarrow ([\tilde{a}]\, s + a_k)/(s+1), \qquad [F] \leftarrow ([F]\, s + a_k(f_{t,k} - f_b))/(s+1), \qquad s \leftarrow s+1$

Examine trial point for archival:

if Random$[0,1] < a_k$ then send $\vec{x}_{t,k}$ and ξ to each peripheral processor for archival

end for

$acceptedOne \leftarrow$ false

$k \leftarrow 1$

while $(k \leq K)$ and $(acceptedOne =$ false$)$ do begin

Update long-lifetime statistics:

$\overline{f} \leftarrow \kappa'' f_b + (1-\kappa'')\overline{f}, \qquad \sigma_f^2 \leftarrow \kappa''(f_b - \overline{f})^2 + (1-\kappa'')\sigma_f^2,$

$S \leftarrow \kappa''|f_b - \overline{f}| + (1-\kappa'')S$

Calculate provisional short-lifetime statistics:

$\chi \leftarrow (f_b - \overline{f})/\sigma_f, \qquad c \leftarrow c+1,$

$\langle\langle c\rangle\rangle \leftarrow \kappa' c + (1-\kappa')\langle c\rangle, \qquad \langle\langle c[\tilde{a}]\rangle\rangle \leftarrow \kappa' c[\tilde{a}] + (1-\kappa')\langle c[\tilde{a}]\rangle,$

$\langle\langle s\rangle\rangle \leftarrow \kappa' s + (1-\kappa')\langle s\rangle, \qquad \langle\langle s\chi\rangle\rangle \leftarrow \kappa' s\chi + (1-\kappa')\langle s\chi\rangle,$

$\langle\langle s\chi^2\rangle\rangle \leftarrow \kappa' s\chi^2 + (1-\kappa')\langle s\chi^2\rangle, \qquad \langle\langle s[\tilde{a}]\rangle\rangle \leftarrow \kappa' s[\tilde{a}] + (1-\kappa')\langle s[\tilde{a}]\rangle,$

$\langle\langle s\chi[\tilde{a}]\rangle\rangle \leftarrow \kappa' s\chi[\tilde{a}] + (1-\kappa')\langle s\chi[\tilde{a}]\rangle,$

$\langle\langle s\chi[F]/\sigma_f\rangle\rangle \leftarrow \kappa' s\chi[F]/\sigma_f + (1-\kappa')\langle s\chi[F]/\sigma_f\rangle$

Calculate estimates of $\overline{a}$, R, and η:

$\overline{a} \leftarrow \langle\langle c[\tilde{a}]\rangle\rangle \,/\, \langle\langle c\rangle\rangle, \qquad R \leftarrow -\langle\langle s\chi[F]/\sigma_f\rangle\rangle \,/\, \langle\langle s\chi^2\rangle\rangle,$

$\eta \leftarrow \{\, \langle\langle s\rangle\rangle\langle\langle s\chi[\tilde{a}]\rangle\rangle - \langle\langle s\chi\rangle\rangle\langle\langle s[\tilde{a}]\rangle\rangle \,\} \,/\, \{\, \langle\langle s[\tilde{a}]\rangle\rangle\langle\langle s\chi^2\rangle\rangle \,\}$

Examine move:

if Random$[0,1] < a_k$ then

Accept move:

$\vec{x}_b \leftarrow \vec{x}_{t,k}, \qquad f_b \leftarrow f_{t,k}, \qquad acceptedOne \leftarrow$ **true**

Copy provisional short-lifetime statistics into short-lifetime statistics proper:

$\langle c\rangle \leftarrow \langle\langle c\rangle\rangle, \quad \langle c[\tilde{a}]\rangle \leftarrow \langle\langle c[\tilde{a}]\rangle\rangle, \quad \langle s\rangle \leftarrow \langle\langle s\rangle\rangle, \quad \langle s\chi\rangle \leftarrow \langle\langle s\chi\rangle\rangle, \quad \langle s\chi^2\rangle \leftarrow \langle\langle s\chi^2\rangle\rangle,$

$\langle s[\tilde{a}]\rangle \leftarrow \langle\langle s[\tilde{a}]\rangle\rangle, \quad \langle s\chi[\tilde{a}]\rangle \leftarrow \langle\langle s\chi[\tilde{a}]\rangle\rangle, \quad \langle s\chi[F]/\sigma_f\rangle \leftarrow \langle\langle s\chi[F]/\sigma_f\rangle\rangle$

Reset local aggregates, s, and c:

$s \leftarrow 0, \qquad c \leftarrow 0, \qquad [\tilde{a}] \leftarrow 0, \qquad [F] \leftarrow 0$

end if

Update β and ξ:

$\Delta\beta \leftarrow \epsilon R/S$

$\beta \leftarrow \beta + \Delta\beta$

if $\eta < 0.8\,\widehat{\eta}_N(\mu_{opt}^{[K]}(N))$ then $\xi \leftarrow \xi\theta^{\overline{a}}$ else if $\eta > \widehat{\eta}_N(\mu_{opt}^{[K]}(N))/0.8$ then $\xi \leftarrow \xi/\theta$

$k \leftarrow k+1$

end while

Update τ', τ'', κ', and κ'':

$\tau'' \leftarrow \max[\tau''_{min},\, 1/(4|\Delta\beta/\beta| + 0.1/\tau''_{min})], \qquad \kappa'' \leftarrow 1 - e^{-1/\tau''}$

$\tau' \leftarrow \max(\tau'_{min},\, \overline{a}\tau''), \qquad \kappa' \leftarrow 1 - e^{-1/\tau'}$

Fig. 17. Examination, statistics processing, and parameter updating.

For example, it might be said that the global minimum has been located if the point with the lowest function value encountered during annealing is sufficiently close to the global minimum in either location or objective function value. Alternatively, it might be said that the global minimum has been located if the final base point – or the lowest point encountered – is within the "basin" of the global minimum (that is, if a continuous path along $-\nabla f$ from the final base point terminates at the global minimum). Another difficulty in defining efficiency stems from the fact that, generally speaking, the probability of success is a monotonically increasing function of the amount of computational effort applied to the problem. That is, by lowering the rate of increase of β, the probability of success can generally be raised at the cost of more objective function evaluations. Parallel processing further complicates the definition of efficiency since it becomes necessary to ask whether the probability of success is greater for a single run of the annealing process using K processors or for two independent, simultaneous runs, each using $K/2$ processors.

In our definition of efficiency, it is assumed that the amount of time needed to perform each run of the annealing process is proportional to the number of iterations in that run. Recall that each iteration consists of the generation of a different random trial point from a common base point by each of K peripheral processors. If $\tilde{K}$ represents the *total* number of peripheral processors, $\tilde{K}/K$ *independent* runs of the annealing process can be performed concurrently. Let Λ be the *total* number of iterations that can be performed in the available time. If λ is the mean number of iterations performed per run, then approximately Λ/λ sets of $\tilde{K}/K$ simultaneous runs can be performed in that time. The total number of runs that may be performed, therefore, is approximately $(\tilde{K}\Lambda)/(K\lambda)$. Now let p_f represent the probability of failure for a single run of the annealing process. The probability of failure in every one of the $(\tilde{K}\Lambda)/(K\lambda)$ runs is $p_f^{\tilde{K}\Lambda/(K\lambda)}$. Estimates of p_f and λ can be found by performing a number of runs with fixed values for the parameters ε and K. For convenience, the measure of efficiency is expressed as the expected number of function evaluations – i.e. the particular value of $\tilde{K}\Lambda$, denoted $\overline{n}$ – necessary to reduce the probability of failure by a factor of e:

$$\overline{n} = \frac{-K\lambda}{\ln(p_f)}. \tag{76}$$

This analysis is valid whenever there are many independent runs, i.e. whenever $\tilde{K}\Lambda \gg K\lambda$. Notice that $\overline{n}$ is independent of $\tilde{K}$; recall that, in fact, $\tilde{K}$ may be unity in which case the single processor evaluates steps in sets of K during each iteration. This measure of efficiency is applied to the two test cases in the next section.

4.2. TEST CASES

The results of the application of ASA to two test cases are given here. To our knowledge, these examples represent the only classes of non-trivial examples (i.e. the reported success probabilities were typically significantly less than 100%) from papers describing other annealing variants. Like many of those presented in other papers, these examples are contrived – to begin, the functions are roughly isotropic.

In these examples, efficiency ($\overline{n}$) is measured as a function of the annealing rate parameter ε and the number of peripheral processors, K. In this manner, the trade-off between speed and success probability is demonstrated. The results are also compared with estimates of the efficiency of other algorithms. In all cases a run is considered successful if the lowest function value encountered is less than the second-lowest minimum, f^{**}. Since, in practical applications, there is generally a reasonable *a priori* level for changes in the value of the objective function that are considered insignificant, we choose to terminate a run when σ_f falls below a prescribed minimum value, say σ_f^*.

4.2.1. *Example 1: Quartic function*

The first example is the ten dimensional quartic, from Styblinski and Tang [12]

$$f(\mathbf{x}) = \frac{1}{10}\sum_{n=1}^{10}(x_n^4 - 16x_n^2 + 5x_n). \tag{77}$$

The region of interest is $\{\Omega : -5 \leqslant x_n \leqslant 5, n = 1, \ldots, 10\}$. The value of f at the global minimum is $f^* \approx -78.3323$, the value of f at the second-lowest minimum is $f^{**} \approx f^* + 2.8$, and the termination criterion is taken here to be $\sigma_f^* = 0.5$.

This function is chosen to demonstrate the relationship between speed and probability of success as a function of annealing rate parameter and number of peripheral processors. ASA produced the results shown in Figure 18 (each point corresponds to 30 runs, and the grey line gives the number of failures in 30 runs). In each case, $\overline{n}$ is nearly constant over much of the range $10^{-3} \leqslant \varepsilon \leqslant 10^{-2}$: smaller values of ε produce higher success probabilities but require greater numbers of function evaluations. It is encouraging that these two effects essentially cancel each other so that the efficiency is nearly independent of the only user-selectable parameter.

The efficiency is generally best (i.e. $\overline{n}$ is lowest) for K greater than unity. The superior performance in these cases is due to the more accurate statistics that outweigh the lack of a proportional increase in the distance moved per iteration (see the discussion of Figure 16). Eventually, increasing K must result in poorer efficiency, but the turn-around point is bound to be problem-specific. These results indicate that gains in efficiency can be realized by *simulating* multiprocessing on a single-processor computer.

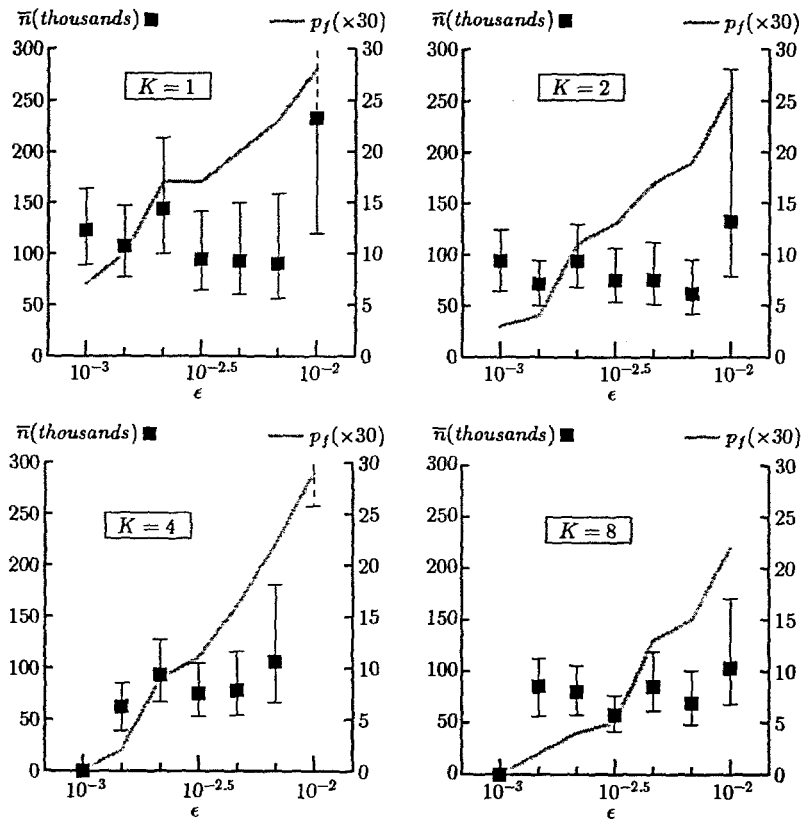

Fig. 18. Efficiency vs. annealing rate parameter for example function 1, $K = 1, 2, 4, 8$. (It is difficult to place meaningful error bars on the two points for which p_f is zero.)

Styblinski and Tang [12] apply their Stochastic Approximation with Convolution Smoothing (SAS) algorithm and the Fast Simulated Annealing (FSA) of Szu [13] to this problem. In three sets of runs using various cooling rates, FSA achieved efficiencies of $\overline{n} \approx 94900$ ($p_f = 27/30$, $\lambda \approx 10000$), $\overline{n} \approx 295000$ ($p_f = 29/30$, $\lambda \approx 10000$), and $\overline{n} \approx 280000$ ($p_f = 7/10$, $\lambda \approx 100000$). The FSA variant lacks adaptivity and involves two problem-specific parameters that must be hand-picked: one relates to the annealing schedule and the other fixes the generator – the characteristic length of the generated steps is proportional to T. Further, FSA is not invariant under linear coordinate transforms and the principal axes of the generator are arbitrarily aligned with the coordinate axes. This particular objective function is roughly isotropic, so the associated best-case results do not expose FSA's weaknesses; nevertheless, ASA is more efficient than the best-case results reported for FSA on this example.

On the face of the reported results, SAS achieves a significantly superior efficiency of $\overline{n} \approx 7570$ ($p_f = 8/30$, $\lambda \approx 10000$) on this particular problem. Although

this result appears to be significantly better, it should be noted that SAS is not only starting-point dependent, but also requires a user-specified series of values for its convolution parameter – that is, there is a large hidden cost in the computation for optimally adjusting all of the control parameters by trial and error. Furthermore, SAS is not invariant under linear coordinate transformations and the assumptions underlying the design of SAS mean that it is only well-suited to problems such as this one, in which the smoothed approximation to the objective function has a minimum that is relatively close to the true global minimum. Keep in mind that, for a unimodal function, all of these algorithms will be soundly outperformed by any simple "downhilling" algorithm.

4.2.2. *Example 2: "Hartman" function*

The second example is the function "F5" from Vanderbilt and Louie [14] (the "Hartman" function ($N = 6$) from Dixon and Szegö [4]):

$$f(\mathbf{x}) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{n=1}^{6} a_{in}(x_n - p_{in})^2\right) \tag{78}$$

where $\mathbf{c} = (1, 1.2, 3, 3.2)$,

$$\mathbf{A} = \begin{pmatrix} 10.00 & 3.00 & 17.00 & 3.50 & 1.70 & 8.00 \\ 0.05 & 10.00 & 17.00 & 0.10 & 8.00 & 14.00 \\ 3.00 & 3.50 & 1.70 & 10.00 & 17.00 & 8.00 \\ 17.00 & 8.00 & 0.05 & 10.00 & 0.10 & 14.00 \end{pmatrix},$$

$$\mathbf{P} = \begin{pmatrix} 0.1312 & 0.1696 & 0.5569 & 0.0124 & 0.8283 & 0.5886 \\ 0.2329 & 0.4135 & 0.8307 & 0.3736 & 0.1004 & 0.9991 \\ 0.2348 & 0.1451 & 0.3522 & 0.2883 & 0.3047 & 0.6650 \\ 0.4047 & 0.8828 & 0.8732 & 0.5743 & 0.1091 & 0.0381 \end{pmatrix}. \tag{79}$$

The region of interest is $\{\Omega : 0 \leqslant x_n \leqslant 1, n = 1, \ldots, 6\}$. The value of f at the global minimum is $f^* \approx -3.322359$, the value of f at the second-lowest minimum is $f^{**} \approx f^* + 0.115$, and the termination criterion is here taken to be $\sigma_f^* = 0.1$. Notice that, as with many of the standard test functions, the valleys here are artificially aligned with the coordinate axes.

In Vanderbilt and Louie [14], the global minimum is located 62 times in 100 runs ($p_f = 0.38$), using an average of $\lambda = 1914$ function evaluations, yielding an efficiency rating of $\overline{n} \approx 1980$. The variant of annealing employed there requires the specification of both an initial value for the acceptance parameter T (temperature) and the rate of reduction of T.

In six sets of 100 runs, ASA obtained the results given in Table II. For purposes of comparison the annealing rate parameter ε was chosen to produce roughly the same number of function evaluations as in Vanderbilt and Louie [14]. It turns out that this problem is poorly suited to algorithms like simulated annealing: generally,

TABLE II. Efficiency vs. K and ϵ for example function 2

K	ϵ	p_f	λ	$\overline{n}$
1	3.00×10^{-2}	39/100	1790 ± 500	1900 ± 700
1	4.00×10^{-2}	47/100	1390 ± 440	1850 ± 740
2	3.00×10^{-2}	50/100	870 ± 240	2510 ± 920
2	4.00×10^{-2}	38/100	770 ± 240	1590 ± 630
4	3.00×10^{-2}	51/100	480 ± 160	2830 ± 1200
4	4.00×10^{-2}	47/100	400 ± 140	2110 ± 920
8	3.00×10^{-2}	38/100	340 ± 100	2790 ± 1200
8	4.00×10^{-2}	41/100	280 ± 90	2510 ± 1100

the random walk wanders until the base point falls into one of the Gaussian pits and remains there until termination. Nevertheless, the efficiency rating $\overline{n}$ for ASA is comparable to the best-case results reported by Vanderbilt and Louie [14]. It is important to realize that ASA requires the specification of only one parameter – ε – and that the acceptance parameter updating and step generation are completely adaptive.

5. Conclusions

With weak dependence on its single user-specified parameter, Adaptive Simulated Annealing successfully avoids the hidden costs of other annealing variants while achieving performance comparable to their best-case results. Through trials with practical problems, ASA has proven to be a robust and effective global optimization method. For example, we have applied ASA to optimization problems in lens system design [6, 7]. In these problems, inequality constraints (both linear and nonlinear) are of central importance and the choice of nonlinear transformations upon the variable space can significantly affect efficiency. ASA has been able to locate solutions to some problems that are superior to the best systems obtained by conventional computer-aided design methods.

A number of options for further refinement of ASA remain to be explored. For example, the implementation of the "smooth" acceptor $a(\mathbf{y}, \mathbf{x}; \beta) = [1 + \exp\{\beta[f(\mathbf{y}) - f(\mathbf{x})]\}]^{-1}$ might be considered. Analysis of the quadratic suggests that, for large K, this acceptor produces larger values of $\hat{E}_N(\mu_{opt}^{[K]}(N), \varepsilon)/\varepsilon$ than those produced by the canonical acceptor. Another possible refinement to consider is the introduction of *adaptive* nonlinear transformations. Although ASA is invariant under *linear* coordinate and objective function transformations, *nonlinear* transformations may be used to raise efficiency. It is, of course, possible to apply *fixed* nonlinear transformations to any problem as the ASA algorithm stands;

further gains in efficiency, however, might be possible by adaptive modification of the nonlinear transformations after each run.

We expect the practicality of algorithms like ASA to blossom. Although the exploration of the performance of ASA to date has been limited chiefly by hardware speed (e.g., the computations for Example 1 required about a month of computer time on an eight-processor system of Inmos T800 Transputers), the continuing increases in affordable computing power – including parallel processing – will facilitate algorithm development. With faster computers, it will be possible to examine ASA's efficiency for problems of interest and to find empirically the optimal values of M, η, θ, and the product $\rho\nu$, and thereby to determine the validity of the general application of the results from the quadratic model. Exploration of alternative schemes for controlling β and ξ would permit more direct validation of our heuristic. With its adaptivity coupled with invariance under linear transformations, ASA provides a sound framework for meeting the challenges posed by a variety of practical global optimization problems.

A. Appendix

The equilibration rate for the N-dimensional quadratic is written as $\hat{R}_N$, and is found to satisfy

$$\begin{aligned}\hat{R}_N(\beta,\mu) \;=\; & 1 - \frac{1}{\hat{S}_N(\beta)} \int_\Omega \hat{\pi}_N(\mathbf{y};\beta) \left| \hat{f}_N(\mathbf{y}) - \hat{\bar{f}}_N(\beta) \right. \\ & + \int_\Omega e^{-\beta \max(\hat{f}_N(\mathbf{x}) - \hat{f}_N(\mathbf{y}),0)} \\ & \left. \cdot [\hat{f}_N(\mathbf{x}) - \hat{f}_N(\mathbf{y})] g(\mathbf{y},\mathbf{x};\mu)\, \mathrm{d}V_x \right| \mathrm{d}V_y. \end{aligned} \tag{80}$$

As described in Section 2.2.1, this can be reduced to a one-dimensional integral that can readily be evaluated numerically:

$$\hat{R}_N(\mu) = 1 - \left(\frac{2e}{N}\right)^{N/2} \int_0^\infty v^{N-1} e^{-v^2} \left| v^2 - \frac{N}{2} + \hat{F}_N(v) \right| \mathrm{d}v. \tag{81}$$

For odd N, $\hat{F}_N(v)$ is given by

$$\begin{aligned}\hat{F}_N(v) \;=\; & X(1-X)^{N/2} \left\{ \frac{1}{2\sqrt{\pi}} \left[v\sqrt{X} e^{-4Nv^2/\mu^2} + v\frac{2-X}{\sqrt{X}} \right] \right. \\ & \left. + e^{Xv^2} \left[\frac{N}{2} - (2-X)v^2 \right] \left[1 - \frac{1}{2}\mathrm{erf}\left(v\frac{2-X}{\sqrt{X}} \right) - \frac{1}{2}\mathrm{erf}(v\sqrt{X}) \right] \right\} \\ & + \frac{\mu^2}{N} \left[\frac{N}{2}\frac{1}{2}\mathrm{erf}\left(2\frac{\sqrt{N}v}{\mu} \right) - \frac{\sqrt{N}v}{\mu\sqrt{\pi}} \right] \\ & + \frac{\mu^2}{N} e^{-2Nv^2/\mu^2} \sum_{m=0}^{\frac{N-3}{2}} \left(\frac{N-1}{2} - m \right) \end{aligned}$$

$$\times \left[(1-X)^{\frac{N+1}{2}-m} - 1\right] I_{m+\frac{1}{2}}\left(2\frac{Nv^2}{\mu^2}\right), \tag{82}$$

and, for even N, it satisfies

$$\hat{F}_N(v) = X(1-X)^{N/2} e^{Xv^2}\left[\frac{N}{2} - (2-X)v^2\right] + \frac{\mu^2}{N} e^{-2Nv^2/\mu^2} \sum_{m=0}^{\infty} (m+1)[(1-X)^{-m} - 1] I_{m+\frac{N}{2}+1}\left(2\frac{Nv^2}{\mu^2}\right), \tag{83}$$

where $X = \mu^2/(\mu^2 + N)$ and I_m is the Modified Bessel function of order m.

Notes

[1] A similar method was developed by Pincus [10] for the analytic solution of global optimization problems.

[2] Other acceptors may be chosen: e.g., $a(\mathbf{y}, \mathbf{x}; \beta) = [1 + \exp\{\beta[f(\mathbf{y}) - f(\mathbf{x})]\}]^{-1}$. The canonical form is adopted here.

[3] In the theory of Markov processes, $\underline{\pi}$ is often called the *invariant* distribution; see Feller ([5], pp. 392–399).

[4] Strictly speaking, it is also necessary that the generator and the region of interest are chosen so that the problem is *irreducible*: for finite β, each point in Ω must be reachable in a finite number of iterations from all other points in Ω.

[5] Under detailed balance, the "flow" of occupation density at equilibrium from $\mathbf{x}$ to $\mathbf{y}$, $\underline{\pi}(\mathbf{x})a(\mathbf{y}, \mathbf{x}; \beta)$ $g(\mathbf{y}, \mathbf{x}; \Psi)$, equals the flow from $\mathbf{y}$ to $\mathbf{x}$, $\underline{\pi}(\mathbf{y})a(\mathbf{x}, \mathbf{y}; \beta)g(\mathbf{x}, \mathbf{y}; \Psi)$, for all $\mathbf{x}$ and $\mathbf{y}$ in Ω. The integrand in equation (3) then vanishes identically and, since the generator is symmetric, this result leads directly to equation (4).

[6] This scheme is similar to the "high-temperature mode" of the parallel-processing annealing scheme devised for circuit layout (combinatorial optimization) by Roussel-Ragot and Dreyfus [11].

[7] Other "distance" measures are possible. The measure proposed here is invariant under *linear* coordinate transformations and, loosely speaking, gives the proportion of the occupation density that is "out of place".

[8] The factor of N appearing in equation (22) has been included for convenience – it turns out that the statistics for the quadratic can be plotted more clearly if μ is defined as $\sqrt{N}\sigma/\sigma_r$ instead of simply σ/σ_r.

[9] In Figures 12–16, the statistics for $K = 1$ are those of the single-processing case, since $\hat{E}_N^{[1]}(\mu, \varepsilon)/\varepsilon$ is just $\hat{R}_N(\mu)$.

References

Bohachevsky, I. O. Johnson, M. E., and Stein, M. L. (1986), Generalized Simulated Annealing for Function Optimization, *Technometrics* **28**, 209–217.

Černý, V. (1985), Thermodynamical Approach to the Traveling Salesman Problem: An Efficient Simulation Algorithm, *J. Optim. Theory Appl.* **45**, 41–51.

Corana, A., Marchesi, M., Martini, C., and Ridella, S. (1987), Minimizing Multimodal Functions of Continuous Variables with the "Simulated Annealing" Algorithm, *ACM Trans. Math. Software* **13**, 263–280.

Dixon, L. C. W. and Szegö, G. P. (1978), The Global Optimization Problem: An Introduction, *Towards Global Optimization 2*, North-Holland, Amsterdam, 1–15.

Feller, W. (1950), *An Introduction to Probability Theory and Its Applications*, Wiley, New York.

Forbes, G. W. and Jones, A. E. W. (1991), Towards Global Optimization with Adaptive Simulated Annealing, *SPIE Conference Proceedings 1354*, Bellingham, WA, 144–153.

Forbes, G. W. and Jones, A. E. W. (1992), Global Optimization in Lens Design, *Optics and Photonics News*, **3**, 22–29.

Kirkpatrick, S., Gelatt Jr., C. D., and Vecchi, M. P. (1983), Optimization by Simulated Annealing, *Science* **220**, 671–680.

Metropolis, N., Rosenbluth, A.W., Rosenbluth, M.N., Teller, A.H., and Teller, E. (1953), Equation of State Calculations by Fast Computing Machines, *J. Chem. Phys.* **21**, 1087–1092.

Pincus, M. (1968), A Closed Form Solution of Certain Programming Problems, *Oper. Res.* **16**, 690–694.

Roussel-Ragot, P. and Dreyfus, G. (1990), A Problem Independent Parallel Implementation of Simulated Annealing: Models and Experiments, *IEEE Trans. Comput. Aided Design* **9**, 827–835.

Styblinski, M. A. and Tang, T.-S. (1990), Experiments in Nonconvex Optimization: Stochastic Approximation with Function Smoothing and Simulated Annealing, *Neural Networks* **3**, 467–483.

Szu, H. H. (1986), Fast Simulated Annealing, *AIP Conference Proceedings 151*, Snowbird, UT, 420–425.

Vanderbilt, D. and Louie, S.G. (1984), A Monte Carlo Simulated Annealing Approach to Optimization over Continuous Variables, *J. Comput. Phys.* **56**, 259–271.