
Journal of Global Optimization **22:** 271–284, 2002.

Approximation of the distribution of convergence times for stochastic global optimisation

G. R. WOOD*, D. L. J. ALEXANDER and D. W. BULGER
Institute of Information Sciences and Technology, Massey University,
*Palmerston North, New Zealand; *Corresponding author (E-mail: G.R.Wood@massey.ac.nz)*

Abstract. How long should we run a stochastic global optimisation algorithm such as simulated annealing? How should we tune such an algorithm? This paper proposes an approach to the study of these questions through successive approximation of a generic stochastic global optimisation algorithm with a sequence of stochastic processes, culminating in a backtracking adaptive search process. Our emerging understanding of backtracking adaptive search can thus be used to study the original algorithm. The first approximation, the averaged range process, has the same expected number of iterations to convergence as the original process.

Key words: Adaptive search, Markov chain, Optimization, Stochastic approximation, Stochastic process

1. Introduction

Stochastic global optimisation methods, such as simulated annealing [6] and genetic algorithms [4], are widely used to solve large scale non-linear optimisation problems. A vexing issue in such work is that of knowing when to conclude a run; how do we know when we are very close to the optimum?

A number of authors, for example [8, 11], have explored the convergence of simulated annealing. Under specified conditions they are able to describe the asymptotic convergence rate of an algorithm. For the practitioner, however, specific information about the number of iterations to run is also needed. For this we need to know something about the distribution of the number of iterations to termination (that is, first obtaining a good enough objective function value) for the algorithm.

Results of this type are available for pure adaptive search (PAS) and its generalisation, hesitant adaptive search (HAS), both Markov range processes whose distribution to termination is fully understood [2, 12, 13]. Convergence rates for non-Markov range processes have been considered in [1]; and for adaptive search, an idealisation of simulated annealing, in [10].

The aim of this paper is to approximate the domain-based stochastic process (DP) of a real algorithm with a range-based Markov process that yields to analysis. Three intermediate processes are used in the approximation. Firstly, the range process (RP) is defined as the image of the domain process in the range. Secondly, range distributions implied at various domain points are mixed at each iteration

to define a time-inhomogeneous Markov process in the range, the averaged range process (ARP). The third process is constructed using the limits over time of the transition matrices appearing in the second process. The asymptotic averaged range process (AARP) constructed in this way is time-homogeneous and Markovian. Finally, for ease of analysis, this process may be approximated by another Markov range process, backtracking adaptive search (BAS). This is a further generalisation of PAS and HAS, for which the analysis is currently being prepared.

Linking an algorithm with a BAS approximation serves two purposes. It provides:

1. Information about the convergence rate of the optimisation algorithm, and
2. A means for investigating how to tune an optimisation algorithm to a given problem.

This work initiates a connection between adaptive search theory and real search algorithms. A framework is laid out and some encouraging theoretical results are presented. Much still remains to be done, however, to complete the picture.

The paper is arranged as follows. In the next section we describe a generic stochastic global optimisation algorithm, upon which the subsequent analysis centres. The sequence of approximations is then described, taking us from the range process through to a backtracking adaptive search process in three stages. The main analytical result of the paper appears in Subsection 2.1, focusing on the first approximation. The range process is replaced with a time-inhomogeneous Markov chain, and it is shown that these two processes have the same expected number of iterations until convergence. We conclude with a summary in Section 3.

2. Approximating a stochastic global optimisation algorithm

The problem under consideration is the very general global optimisation problem

$$\text{minimise } f(x), \text{ subject to } x \in S$$

where S is a measurable space and $f : S \to \mathbb{R}$ is a measurable function. We now introduce a general form for stochastic global optimisation algorithms. We are given an initial domain distribution δ_0 and a local search measure for each $x \in S$. The local search measure at x is used to generate the next candidate point in the domain.

Generic stochastic global optimisation algorithm

Step 1 Generate X_0 in S according to δ_0. Set $Y_0 = f(X_0)$ and $n = 0$.

Step 2 Choose Z according to a local search measure at X_n. If $f(Z) \leq Y_n$ then set $X_{n+1} = Z$. If $f(Z) > Y_n$ then set $X_{n+1} = Z$ with a certain probability (which may depend on X_n and Z) or $X_{n+1} = X_n$ otherwise. Set $Y_{n+1} = f(X_{n+1})$.

Step 3 If a stopping criterion is met, stop. Otherwise, increment n and return to Step 2.

To investigate the behaviour of this algorithm, we set up a sequence of stochastic processes, each approximating the sequence of objective function values obtained by the generic stochastic global optimisation algorithm. As we move through the sequence of processes, the accuracy of the approximation will slowly deteriorate, but in doing so we reach an approximation with behaviour that can be expected to yield to analysis. We will discuss the following stochastic processes:

Domain process: This is the sequence of domain points generated by the generic stochastic global optimisation algorithm.

Range process: This is the sequence of function values generated by the generic stochastic global optimisation algorithm.

Averaged range process: This is a time-inhomogeneous Markov process in the range based on the range process. It is presented in Subsection 2.1.

Asymptotic averaged range process: This is obtained by homogenising the averaged range process. It is presented in Subsection 2.2.

Backtracking adaptive search: This is obtained by requiring the transition distributions in the range, conditioned on improvement or worsening, to be restrictions of a single range probability measure ρ. It is presented in Subsection 2.3.

The first of these processes takes on values in the objective function's domain, whereas the others assume values in the range. This important distinction is illustrated in Figure 1.

The domain process (X_n) as defined above is Markovian. This is general enough to encompass the Metropolis algorithm [9], genetic algorithms [4] (with S then a product space, the number of factors being determined by the population size) and certain evolutionary algorithms.

Note that the range process (Y_n) is not in general Markovian, despite being the image under f of the Markov chain (X_n). We can interpret the Markovian property of the sequence (X_n) as saying that the process has no memory; if we can see the current value of X_n then we know the complete state of the process, and no other information will help us to predict X_{n+1}. If we can see only the range process, however, then unless the objective function f is invertible, we are receiving information about the domain process through a noisy channel. Better information may be found if more than just the most recent range value is used. For instance, suppose the current range value is 2. There may be many points in the domain where the objective function takes this value. Now suppose also that at the previous iteration the range value was 3, and that there is only one domain point with value 2 reachable in one iteration from a domain point with value 3. Knowing the current and previous range values allows us to deduce the domain position, and consequently the exact distribution of range values at the next iteration. The current value of Y_n alone allows only an estimation of the range distribution at the next iteration; thus (Y_n) is not Markovian in this case.

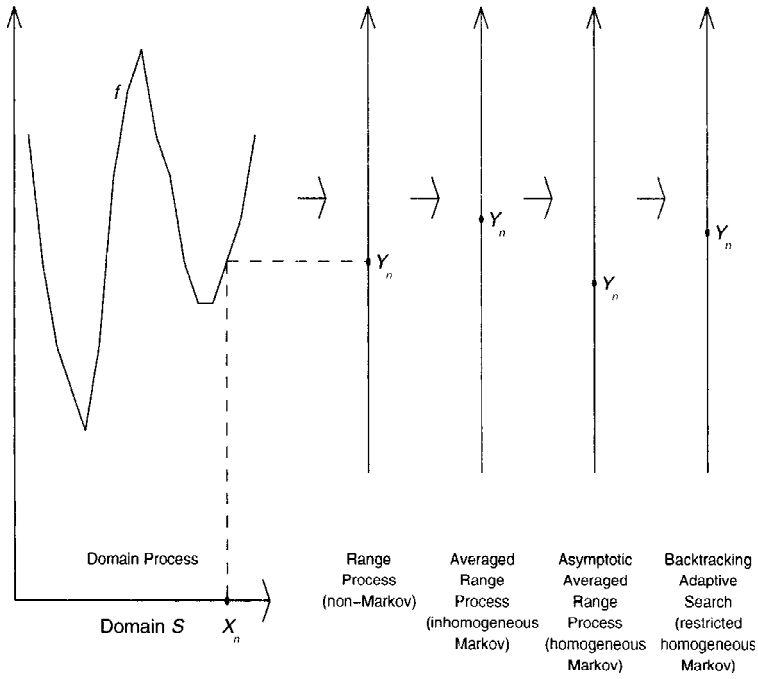

Figure 1. Steps in the approximation of the domain process with a backtracking adaptive search process.

This paper considers only first-order Markov approximations to the range process; redefinition of the state space to include more than one iteration in each state would easily allow generalisation to higher-order Markov approximations, perhaps providing worthwhile improvement in the accuracy of the approximation. This procedure is common in some other search algorithms, such as tabu search [3].

We now introduce an example which will serve to illustrate the concepts introduced throughout the paper.

Example. Let $S = \{1, 2, 3\}$ and $f(1) = 1$, $f(2) = 2$ and $f(3) = 2$. A search algorithm to find the minimum in this simple example is described by a Markov domain process with transition matrix

$$\begin{array}{rc} & \text{Next state} \\ & \begin{array}{ccc} 1 & 2 & 3 \end{array} \\ \text{Current state} \begin{array}{r} 1 \\ 2 \\ 3 \end{array} & \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix} \end{array}.$$

Suppose the process is equally likely to begin in each of the domain states. Then standard theory [5] provides that the expected number of iterations before convergence for this process is $2\frac{1}{3}$. The variance is $8\frac{2}{9}$.

The range process based on this domain algorithm has an absorbing state at the low level, 1. At the high level, 2, RP has two possible range distributions for the next iteration: if it is at domain state 2 then it will move to domain state 3 with certainty and thus remain at the high level; if it is at domain state 3 then RP is equally likely to remain at the high level or move to the low level.

We now present in detail the three stages of approximation of the range process.

2.1. THE AVERAGED RANGE PROCESS

In the first stage RP is approximated with an inhomogeneous Markov process, ARP. The following derivation assumes a discrete domain S. Convergence is held to occur when $Y_n \leq y$ for some real number y, chosen before algorithm commencement; domain locations with objective function value not greater than y are lumped into one absorbing state x_1 with arbitrary objective function level $y_1 \leq y$. It is assumed that all other domain locations are transient. The domain search algorithm may thus be considered as a Markov process with the following transition matrix, in block form:

$$P = \left[\begin{array}{c|c} 1 & \mathbf{0} \\ \hline \boldsymbol{r} & Q \end{array}\right].$$

The first row of this matrix, showing transitions from the absorbing state, has a one in the first position and zeros elsewhere. The column vector $\boldsymbol{r}$ gives each transient state's probability of moving directly to the absorbing state. The remaining submatrix Q is substochastic, and gives the transition probabilities between all of the transient states.

We let the transient domain states be $x_2, x_3, \ldots, x_l$ and the transient objective function levels be $y_2, y_3, \ldots, y_m$. We then let δ_n denote a row vector of length l comprising the probabilities of occupying each of the l domain states at the nth iteration. Then $\delta_{n+1} = \delta_n P$ for all $n \geq 0$.

Here is an informal description of the averaged range process. At the nth iteration ARP is at level Y_n and the domain process is at some point in $f^{-1}(Y_n)$. The probability of being at each of these domain states is determined by the normalised restriction of the domain distribution at the nth iteration to $f^{-1}(Y_n)$. Mappings of the transition distributions at each candidate domain state into the range (termed *local range distributions*) are then mixed according to their domain weightings. This produces an averaged range distribution at the nth iteration. Figure 2 illustrates the way in which ARP uses this convex combination of local range distributions of the search algorithm.

We now give a formal definition of the averaged range process. The initial distribution of ARP is identical to that of RP, being the image in the range of the initial distribution of the original domain process. At the nth iteration, suppose that ARP is in state y_i. Then the probability that ARP moves to state y_j at the next iteration is found by summing all the probabilities of distinct transitions in the domain from

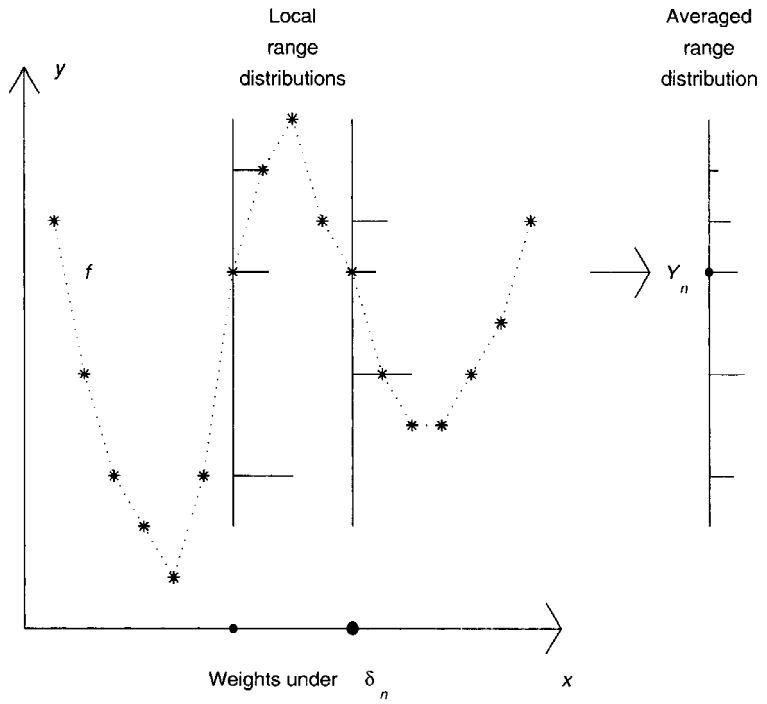

Figure 2. The Markovian approximation to the range process uses a mixture of local range distributions, with weights determined by the domain process. In this figure, the current range value may be due to either of two domain points, each of which gives rise to a probability distribution for the range value after one further iteration. These probability distributions are mixed as shown.

$f^{-1}(y_i)$ to $f^{-1}(y_j)$. That is, the ARP transition matrix at the nth iteration, R_n, is given by

$$(R_n)_{ij} = \mathrm{P}(Y_{n+1} = y_j | Y_n = y_i)$$
$$= \sum_{x_p \in f^{-1}(y_j)} \sum_{x_k \in f^{-1}(y_i), P(x_n = x_k) > 0} \mathrm{P}(X_n = x_k | Y_n = y_i) \mathrm{P}(X_{n+1} = x_p | X_n = x_k)$$

for $n \geq 0$ and $i, j \in \{1, 2, \ldots, m\}$ when $\mathrm{P}(Y_n = y_i) \neq 0$, and $(R_n)_{ij}$ is arbitrarily set to zero otherwise.

Define the domain weightings used by ARP, γ_n, based only on the current range level and iteration number, as

$$\gamma_n(i) = \mathrm{P}(X_n = x_i | Y_n = f(x_i)) = \frac{\mathrm{P}(X_n = x_i)}{\mathrm{P}(Y_n = f(x_i))} = \frac{\delta_n(i)}{\sum_{\{j : f(x_j) = f(x_i)\}} \delta_n(j)} \quad (1)$$

for $i \in \{1, 2, \ldots, l\}$ where this is defined. Otherwise, arbitrarily assign $\gamma_n(i) = 0$.

We now define an $l \times m$ matrix M mapping from domain states to objective function states by

$$M_{ij} = \begin{cases} 1 & \text{if } f(x_i) = y_j, \\ 0 & \text{otherwise.} \end{cases}$$

Then

$$R_n = M^{\mathrm{T}} \mathrm{diag}(\gamma_n) P M \tag{2}$$

where $\mathrm{diag}(\gamma_n)$ is a diagonal matrix with entries of γ_n on the diagonal.

The sequence of processes studied in this paper is designed to span the gap between, at one end, the generic stochastic global optimisation algorithm we wish to understand, and at the other end, BAS, which we expect to be able to analyse. The value of this study is determined by the quality of the approximations. The main analytic result of this paper, presented in this subsection, is that the averaged range process and the range process (and therefore the domain process) share the same expected number of iterations before termination.

Where (Y_n) goes at each iteration will in general depend on more than the last step. By defining ARP based only on the current level, therefore, we distance it somewhat from RP. Despite this, the above definition implies that, while the averaged range process and the range process can in general differ in joint distribution, they must be equal in marginal distribution at each iteration.

THEOREM 2.1 *At any given iteration, the range process and the averaged range process have the same marginal distribution.*

For the proof see the appendix. Further, the following result (similar to a result in [1]) holds true for the mean number of iterations to convergence:

COROLLARY 2.1 *Let N_d be the number of iterations to convergence for the generic stochastic global optimisation algorithm in a finite domain. For the range process let N_r be the number of iterations to convergence, while for the averaged range process let N_a be the number of iterations to convergence. Then*

$$\mathrm{E}(N_d) = \mathrm{E}(N_r) = \mathrm{E}(N_a).$$

The proof appears in the appendix. It makes use of a truncation transformation T_i, equal to the identity matrix of size i (I_i) with the first column removed. This matrix has the effect of removing the first component of a probability position vector, corresponding to the probability of being in the absorbing state. A vector of i ones is also denoted by $\mathbf{1}_i$.

The processes can differ in the variance of the number of iterations to convergence; using the notation described above, the difference in variance between RP and ARP can be written as

$$2\delta_0 \left(T_l \left((I_{l-1} - Q)^{-1} \right)^2 \mathbf{1}_{l-1} - M T_m \left(I_{m-1} + T_m^{\mathrm{T}} R_0 T_m + T_m^{\mathrm{T}} R_0 T_m T_m^{\mathrm{T}} R_1 T_m + \cdots \right)^2 \mathbf{1}_{m-1} \right).$$

Example (continued) Formulating the example problem in terms of this subsection, we find that

$$P = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ \frac{1}{2} & \frac{1}{2} & 0 \end{bmatrix}, \quad \delta_0 = \begin{bmatrix} \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{bmatrix} \text{ and } \quad M = \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 1 \end{bmatrix}.$$

For this cyclic example, explicit expressions for P^n are available as

$$P^n = 2^{-\frac{n}{2}} \begin{bmatrix} 2^{\frac{n}{2}} & 0 & 0 \\ 2^{\frac{n}{2}} - 1 & 1 & 0 \\ 2^{\frac{n}{2}} - 1 & 0 & 1 \end{bmatrix} \text{ for } n \text{ even, while}$$

$$P^n = 2^{-\frac{n+1}{2}} \begin{bmatrix} 2^{\frac{n+1}{2}} & 0 & 0 \\ 2^{\frac{n+1}{2}} - 2 & 0 & 2 \\ 2^{\frac{n+1}{2}} - 1 & 1 & 0 \end{bmatrix} \text{ for } n \text{ odd.}$$

Thus, using (1),

$$\gamma_n = \begin{bmatrix} 1 & \frac{1}{2} & \frac{1}{2} \end{bmatrix} \text{ for } n \text{ even}, \quad \text{while} \quad \gamma_n = \begin{bmatrix} 1 & \frac{1}{3} & \frac{2}{3} \end{bmatrix} \text{ for } n \text{ odd.}$$

All the ARP transition matrices can now be found explicitly by application of (2). Thus

$$R_n = \begin{bmatrix} 1 & 0 \\ \frac{1}{4} & \frac{3}{4} \end{bmatrix} \text{ for } n \text{ even}, \quad \text{while} \quad R_n = \begin{bmatrix} 1 & 0 \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix} \text{ for } n \text{ odd.}$$

Since there is only one absorbing domain state and corresponding range level, the first row and column of M are vectors of zeros with an initial one, and the first element of γ_n is a one. Also the first rows of transition matrices P and R_n are always vectors of zeros with an initial one. In this small example the transient portions of the range transition matrices R_n are single numbers, giving the probability of staying at the transient range level after one transition.

The expectation of N_a can now be found as follows:

$$\begin{aligned} \mathrm{E}(N_a) &= \sum_{n=1}^{\infty} \mathrm{P}(N_a \geq n) \\ &= \frac{2}{3} + \frac{2}{3} \cdot \frac{3}{4} + \frac{2}{3} \cdot \frac{3}{4} \cdot \frac{2}{3} + \frac{2}{3} \cdot \frac{3}{4} \cdot \frac{2}{3} \cdot \frac{3}{4} + \cdots \\ &= 2\frac{1}{3} \end{aligned}$$

This is the same result as that stated earlier, from direct analysis of the process in the domain. Thus ARP preserves the expected number of iterations to convergence of the domain process it approximates.

The variance, however, is found to be $8\frac{5}{9}$. Thus the variance of ARP is not in general the same as that of the domain process it approximates.

2.2. THE ASYMPTOTIC AVERAGED RANGE PROCESS

While ARP is a time-inhomogeneous Markov process, BAS is defined below as a time-homogeneous Markov process. To link the two, it is necessary to remove the iteration dependence of ARP. This is done by observing that the transition matrices R_n for ARP settle down in the limit to a constant matrix, or, possibly, to oscillation between multiple constant matrices. (One way to see this is via Jordan decomposition; the details are left to the interested reader.)

We define the asymptotic averaged range process (AARP) to be the time-homogeneous Markov process with initial distribution equal to that in both the range process and the averaged range process, and with transition matrix

$$R = \lim_{N\to\infty} \frac{1}{N} \sum_{n=0}^{N-1} R_n,$$

where the matrices R_n are the transition matrices describing the averaged range process. Except in special cases, this expression simplifies to $R = \lim_{n\to\infty} R_n$.

Example (continued) The example illustrates the periodic case. The limiting transition matrix may be found as follows:

$$\begin{aligned} R = \lim_{N\to\infty} \frac{1}{2N} \sum_{n=0}^{2N-1} R_n &= \lim_{N\to\infty} \frac{1}{2N} \sum_{n=0}^{N-1} \left(\begin{bmatrix} 1 & 0 \\ \frac{1}{4} & \frac{3}{4} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ \frac{1}{3} & \frac{2}{3} \end{bmatrix} \right) \\ &= \lim_{N\to\infty} \frac{1}{2N} N \begin{bmatrix} 2 & 0 \\ \frac{7}{12} & \frac{17}{12} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 \\ \frac{7}{24} & \frac{17}{24} \end{bmatrix} \end{aligned}$$

Applying standard theory to AARP using this limiting transition matrix gives an expected number of iterations to convergence of $\frac{2}{3}\left(1 - \frac{17}{24}\right)^{-1} = 2\frac{2}{7}$. This is close to the value for the domain process, $2\frac{1}{3}$. Some means of quantifying the error now introduced by the approximation to AARP is required; this remains an area for future research.

2.3. BACKTRACKING ADAPTIVE SEARCH

Over the past decade the pure adaptive search algorithm has been defined and analysed [13]. It has been extended to hesitant adaptive search, now also fully understood [12]. In this subsection we extend HAS to a new stochastic algorithm, BAS. This process is proposed in anticipation that BAS will provide the basis for a sufficiently flexible family of homogeneous Markov range processes for approximating the asymptotic averaged range distributions of stochastic global optimisation algorithms.

At each iteration, the objective function value either improves, remains at the current level, or worsens. It is assumed that the distribution of the next objective function value is then determined using a single distribution. This is acknowledged to be restrictive.

Initially BAS samples Y_0 from $f(S)$ according to the range probability measure ρ with cumulative distribution function p. At each iteration thereafter, one of three things happens. With a known probability $b(Y_n)$, the algorithm will make an improvement, sampling the next evaluation point Y_{n+1} according to the normalised restriction of ρ to the current improving set. With a second known probability $w(Y_n)$, the algorithm will backtrack, sampling the next evaluation point Y_{n+1} according to the normalised restriction of ρ to the current worsening set. Otherwise, the algorithm will hesitate, remaining at the current evaluation point. The functions b and w depend only on the current level. They are assumed to be continuous and to have bounded variation. We now present the algorithm formally.

Backtracking adaptive search

Step 1 Generate Y_0 in $f(S)$ according to ρ. Set $n = 0$.

Step 2 With probability $b(Y_n)$ choose Y_{n+1} according to the normalised restriction of ρ to $(-\infty, Y_n)$. With probability $w(Y_n)$ choose Y_{n+1} according to the normalised restriction of ρ to (Y_n, ∞). Otherwise set $Y_{n+1} = Y_n$.

Step 3 If a stopping criterion is met, stop. Otherwise, increment n and return to Step 2.

Note that HAS is the special instance of BAS occurring when $w(t) = 0$ for all $t \in f(S)$. In turn, PAS is the special instance of HAS occurring when $b(t) = 1$ for all $t \in f(S)$. Thus PAS must improve at every iterate.

Let N_h be the number of iterations before termination for HAS. The mean of N_h is

$$\mathrm{E}(N_h) = \int_y^\infty \frac{d\rho(t)}{b(t)\,p(t)}. \tag{3}$$

The variance of N_h is

$$\mathrm{Var}(N_h) = \int_y^\infty \left(\frac{2}{b(t)} - 1\right) \frac{d\rho(t)}{b(t)\,p(t)}.$$

These results can be found in [12], together with analogous results for the discrete problem. Backtracking adaptive search awaits full analysis. For the discrete case, some convergence theory for BAS has been developed in [7].

The increasingly general families of homogeneous first-order Markov chains provided by PAS, HAS and BAS can serve to approximate the asymptotic averaged range process of a stochastic optimisation algorithm.

Example (continued) The formulation of AARP in the example can be directly transferred into the framework of HAS. In this case there is one transient level and one absorbing level. The range probability measure ρ assigns weight $\frac{1}{3}$ to the absorbing level and $\frac{2}{3}$ to the transient level. Thus the cumulative distribution function p is the piecewise constant function

$$p(t) = \begin{cases} 0, & t < 1, \\ \frac{1}{3}, & 1 \leq t < 2, \\ 1, & 2 \leq t. \end{cases}$$

The bettering probability $b(2)$ is given from the transition matrix in AARP as $\frac{7}{24}$. Applying the discrete case of (3) then gives the expected number of iterations until convergence as $2\frac{2}{7}$, as before. A more complicated approximation process than this would be required for richer examples.

3. Conclusion

This paper describes a framework for approximating the convergence rate of an arbitrary Markovian optimisation algorithm, by linking it to a tractable stochastic process via a chain of intermediate stochastic processes. Each process in the chain is derived from the previous one, and can be used to approximate its convergence behaviour.

In addition, a start has been made on the detailed investigation of the chain of processes. The links from the domain process to the range process, and from there to the averaged range process, have been studied. It has been shown that these three processes all terminate in the same number of iterations in expectation. Numerical experiments suggest that the asymptotic averaged range process usually approximates the averaged range process quite closely.

The complete strategy for analysis enables us to predict how long a particular stochastic global optimisation algorithm should be run to reach a set level. The effectiveness of the stochastic global optimisation algorithm on a particular problem is thus measured. This information is also useful for tuning the algorithm, for example, through tailoring search region to landscape.

4. Appendix

Proof of Theorem 2.1. Denote the probability distribution at the nth iteration of ARP by π_n. The probability distribution of range states in RP is defined as the image of δ_n under f. The theorem may thus be stated as

$$\pi_n = \delta_n M.$$

The proof is by induction. The initial averaged range distribution is the image in the range of the initial domain distribution, so that $\pi_0 = \delta_0 M$.

Now assume the result for some integer $k \geq 0$. Then

$$\begin{aligned}\left[\delta_k M M^{\mathrm{T}} \mathrm{diag}(\gamma_k)\right](i) &= \left[\pi_k M^{\mathrm{T}} \mathrm{diag}(\gamma_k)\right](i) \qquad \text{by hypothesis} \\ &= \mathrm{P}(Y_k = f(x_i))\mathrm{P}(X_k = x_i | Y_k = f(x_i)) \\ &= \mathrm{P}(X_k = x_i) \\ &= \delta_k(i) \end{aligned} \tag{4}$$

when $P(Y_k = f(x_i)) > 0$. If $P(Y_k = f(x_i)) = 0$, $[\Pi_k M^T \mathrm{diag}(\delta_k)](i) = 0 = \delta_k(i)$ so (4) still holds. Thus,

$$\begin{aligned} \pi_{k+1} &= \pi_k R_k && \text{by definition} \\ &= \delta_k M M^{\mathrm{T}} \mathrm{diag}(\gamma_k) P M && \text{by hypothesis and using (2)} \\ &= \delta_k P M && \text{from (4)} \\ &= \delta_{k+1} M. \end{aligned}$$

Hence by induction $\pi_n = \delta_n M$ for all $n \geq 0$. ■

Proof of Corollary 2.1. The proof uses the relationship $T_l Q^n = P^n T_l$, obtained by recognising

$$\begin{aligned} T_l Q^n &= \left[\frac{\mathbf{0}}{I_{l-1}}\right] Q^n \\ &= \left[\frac{\mathbf{0}}{Q^n}\right] \end{aligned}$$

as the truncation under T_l of P^n.

Standard theory [5] then gives

$$\begin{aligned} \mathrm{E}(N_d) &= \delta_0 T_l (I_{l-1} - Q)^{-1} \mathbf{1}_{l-1} \\ &= \delta_0 T_l (I_{l-1} + Q + Q^2 + \cdots) \mathbf{1}_{l-1} \\ &= (\delta_0 T_l + \delta_0 T_l Q + \delta_0 T_l Q^2 + \cdots) \mathbf{1}_{l-1} \\ &= (\delta_0 T_l + \delta_0 P T_l + \delta_0 P^2 T_l + \cdots) \mathbf{1}_{l-1} \\ &= (\delta_0 + \delta_1 + \delta_2 + \cdots) T_l \mathbf{1}_{l-1}. \end{aligned} \tag{5}$$

The range process is defined as the image of the domain process under f. Consequently, the number of iterations to convergence for this process is stochastically equivalent to the number of iterations to convergence in the original domain algorithm, and in particular $\mathrm{E}(N_r) = \mathrm{E}(N_d)$.

Reference to RP is by way of a stepping stone to the new Markov process in the range, ARP. By analogy with (5), the expected number of iterations to convergence for ARP is

$$\begin{aligned}\mathrm{E}(N_a) =&(\pi_0 + \pi_1 + \pi_2 + \cdots)T_m\mathbf{1}_{m-1}\\ =&(\delta_0 M + \delta_1 M + \delta_2 M + \cdots)T_m\mathbf{1}_{m-1}\\ =&(\delta_0 + \delta_1 + \delta_2 + \cdots)MT_m\mathbf{1}_{m-1}.\end{aligned}$$

Since M is defined as a matrix of zeros with exactly one entry of one in each row, it follows that

$$\begin{aligned}\mathrm{E}(N_a) =&(\delta_0 + \delta_1 + \delta_2 + \cdots)T_l\mathbf{1}_{l-1}\\ =&\mathrm{E}(N_d).\end{aligned}$$

A similar development in [1] proceeds from different initial assumptions. ■

5. Acknowledgements

The authors would like to thank the Marsden Fund administered by the Royal Society of New Zealand for support of this research. Also, D. Alexander was supported by a Bright Future Scholarship administered by the Foundation for Research, Science and Technology.

References

1. Baritompa, W. P. and Steel, M. A. (1996), Bounds on absorption times of directionally biased random sequences, *Random Structures and Algorithms* 9: 279–293.
2. Bulger, D. W. and Wood, G. R. (1998), Hesitant adaptive search for global optimisation, *Mathematical Programming* 81: 89–102.
3. Glover, F. and Laguna, M. (1997), *Tabu Search*, Kluwer Academic Publishers, Boston, MA.
4. Holland, J. H. (1975), *Adaptation in natural and artificial systems*, The University of Michigan Press, Ann Arbor, MI.
5. Kemeny, J. G. and Snell, J. L. (1960), *Finite Markov Chains*, Van Nostrand, New York.
6. Kirkpatrick, S., Gelatt, C. D. Jr. and Vecchi, M. P. (1983), Optimisation by simulated annealing, *Science* 220: 671–680.
7. Kristinsdottir, B. P., Zabinsky, Z. B. and Wood, G. R., Discrete backtracking adaptive search for global optimization, in *Stochastic Methods in Global Optimization*, dedicated to the 70th anniversary of Professor J. Mockus, edited by G. Dzemyda, V. Saltenis and A. Zilinskas, Kluwer Academic Publishers, Dordrecht (to appear).
8. Locatelli, M. and Schoen, F. (2000), Convergence rates of simulated annealing, Preprint University of Florence.

9. Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H. and Teller, E. (1953), Equations of state calculations by fast computing machines, *The Journal of Chemical Physics* 21: 1087–1092.
10. Romeijn, H. E. and Smith, R. L. (1994), Simulated annealing and adaptive search in global optimisation, *Probability in the Engineering and Informational Sciences* 8: 571–590.
11. Trouvé, A. (1996), Cycle decompositions and simulated annealing, *SIAM J. Control and Optimization* 34: 966–986.
12. Wood, G. R., Zabinsky, Z. B. and Kristinsdottir, B. P. (2001), Hesitant adaptive search: the distribution of the number of iterations to convergence, *Mathematical Programming* 89: 479–486.
13. Zabinsky, Z. B. and Smith, R. L. (1992), Pure adaptive search in global optimization, *Mathematical Programming* 53: 323–338.