Journal of Global Optimization **11:** 181–191, 1997.

Aspiration Based Simulated Annealing Algorithm

M. M. ALI[1] and C. STOREY[2]
[1]*Research Fellow, Department of Computer Science, Åbo Akademi University, Turku, Finland*
[2]*Research Professor, Department of Mathematical Sciences, DeMontfort University, The Gateway, Leicester, U.K.*

(Received: 6 December 1994; accepted: 16 February 1996)

Abstract. An aspiration based simulated annealing algorithm for continuous variables has been proposed. The new algorithm is similar to the one given by Dekkers and Aarts (1991) except that a kind of memory is introduced into the procedure with a self-regulatory mechanism. The algorithm has been applied to a set of standard global optimization problems and a number of more difficult, complex, practical problems and its performance compared with that of the algorithm of Dekkers and Aarts (1991). The new algorithm appears to offer a useful alternative to some of the currently available stochastic algorithms for global optimization.

Key words: Global optimization, continuous variables, aspiration value, simulated annealing, stochastic.

1. Introduction

The global minimization problem for a function $f : \Omega \subset \mathrm{R}^{\mathrm{n}} \rightarrow \mathrm{R}$ is to find x^* such that

$$f(x^*) \leq f(x), \quad \forall x \in \Omega \, . \tag{1}$$

It is also assumed that the problem is essentially unconstrained, that is, the global minimum f^* of f is attained in the interior of Ω.

In recent years, a number of deterministic and stochastic algorithms have been proposed (Ratschek and Rokne, 1988; Törn and Žilinskas, 1989; Horst and Tuy, 1990 and Floudas and Pardalos, 1992) for solving (1). In this paper we restrict ourselves to stochastic methods, especially the simulated annealing (SA) method. More recent developments on stochastic methods are based on the combination and adaptation of multistart (Rinnooy Kan and Timmer, 1984) and clustering methods (Timmer, 1984 and Törn and Viitanen, 1992). They are single linkage (SL) (Rinnooy Kan and Timmer, 1987), multilevel single linkage (MSL) (Rinnooy Kan and Timmer, 1987a) and topographical multilevel single linkage (TMSL) (Ali and Storey, 1994). These stochastic methods are very successful and robust for problems with a small number of local minima, however, as the number of local minima increases their robustness falls off (Ali, 1994). An extensive numerical investigation of these methods using a number of very complex practical problems is carried out in Ali (1994) with the conclusion that it is possible for the simulated

annealing method to be made a reasonable alternative for problems with large numbers of local minima.

In this paper we present a new SA method. The advantage of the new method is that it can memorize the best solution during a run. In Section 2 we present our new algorithm, in Section 3 a new cooling schedule is proposed and in Section 4 comparisons and numerical results are given. Finally, some conclusions are given in Section 5.

2. The Aspiration Based SA Algorithm (ASA)

The SA algorithm sometimes accepts solutions which are worse than the current solution. It is therefore possible in any single SA run for the final solution to be worse than a solution found during the run. In fact, since the SA algorithm is a randomization device, which by means of an acceptance/rejection criterion allows some ascent steps during the optimization process, it is quite possible that at some fixed temperature level the procedure will visit the optimal solution but due to the acceptance/rejection mechanism it will leave the best solution and arrive at a worse solution. The SA procedure, therefore, is memoryless in the sense that new solutions are accepted disregarding previously obtained intermediate results. The SA algorithm therefore has the following shortcomings:

Simulated annealing does not use strategic decision rules based on knowledge of the global structure of the problem.

No learning procedure is incorporated to make effective use of information gained in previous iterations.

We have developed an aspiration based SA algorithm which attempts to take into account the above drawbacks by adopting a self regulatory mechanism. The algorithm also introduces a new acceptance criterion. As far as simulated annealing for discrete optimization is concerned many researchers have considered alternative acceptance and/or generation probabilities (Romeo and Sangiovanni-Vincentelli, 1985; Greene and Supowit, 1986 and Faigle and Schrader, 1988). In most cases theoretical results regarding asymptotic convergence have been established. In reality, however, a system of cooling particles and an optimization problem are not same and therefore the simulation has to be suitably adapted. Moreover, for the continuous case we may have extra useful information about the objective function such as, for example, its differentiability properties.

Let x_t be the starting point of the t-th Markov chain (a Markov chain in the simulated algorithm is a sequence of trials). We now define the aspiration value $f_a(x_t)$ of the objective function f by which the Metropolis acceptance probability is modified. Let x_a be the point obtained by carrying out a few steps of local search from x_t. We define $f_a(x_t)$ as $f(x_a)$ and use it, in essence, as a guide to the procedure. In our modification of the SA algorithm we use an acceptance criterion which is independent of the current function value f_x whenever Δf_{xy} is positive.

This acceptance criterion is given by

$$A^a_{xy}(T) = \min\left(1, A^*_{xy}(T)\right) \ , \tag{2}$$

where

$$A^*_{xy}(T) = \begin{cases} \exp\left(-(f_y - f_a)/T\right) & \text{if } f_y > f_x \geq f_a \ , \\ 1 & \text{otherwise} \ , \end{cases} \tag{3}$$

where T is the temperature. In their formulation of the SA algorithm for continuous variables Dekkers and Aarts (1991) used the Metropolis acceptance criterion and the generation mechanism **GM**:

$$g_{xy} = \begin{cases} LS(x), & \text{if } \omega \geq t_o, \\ \frac{1}{m(\Omega)}, & \text{if } \omega < t_o, \end{cases} \tag{4}$$

where t_o is a fixed number in $(0, 1)$, and ω a uniform random number in $(0, 1)$. $LS(x)$ denotes a local search that generates a point y in a descent direction from x. The local search from x is not a complete local search but only a few steps of some appropriate descent search. Thus $f_y < f_x$ but y is not necessarily a local minimum. In ASA we use the same generation mechanism but the acceptance probability defined by (2) and (3). This has the effect of motivating the search procedure, via the aspiration value, in a more realistic manner than the Metropolis acceptance criterion. As for convergence, Dekkers and Aarts have proved that their algorithm converges in probability to the global minimum of f. We have also established probabilistic convergence for the ASA algorithm. Our convergence proof closely follows that given by Dekkers and Aarts with the main differences arising because of the new acceptance mechanism. For brevity the proof is omitted from this paper but can be found in full, together with relevant discussion, in Ali (1994). In ASA the same method was used as was used by Dekkers and Aarts to determine the initial temperature, T_o. A brief description of how the ASA algorithm works is now given.

As mentioned earlier T_o is found using the initial temperature calculating scheme for SA. (N.B. the same T_o as in SA will be produced if we use exactly the same local search in **GM**.) A few (one or two) local descent steps are now taken from the random starting point x_o of the initial Markov chain, the resulting solution gives the aspiration value and then the initial chain begins. During the execution of the t-th Markov chain with length L_t, if a point is generated whose function value is less than or equal to the aspiration value, then no more attempt is made to generate the next point. In other words the inner loop stops, the aspiration value is updated and a new Markov chain begins. If a solution can not be found whose function value is less than $f_a(x_t)$ then the complete chain of length L_t is executed and the aspiration value is not updated at the begining of the next Markov chain. The updated aspiration value gives effective information on the objective function as the search proceeds. At the begining of each Markov chain (say, the

t-th Markov chain) the aspiration value is updated if required and the length, L_t, is then determined. The greater the difference between f_{x_t}, the starting solution of the t-th Markov chain and $f_a(x_t)$, the aspiration value, the longer will be the current Markov chain. Therefore, the aspiration value f_a plays a part in determining the length of a particular chain in a way to be clarified later. In ASA, therefore, the choice of L_t is not constant throughout the course of the algorithm, it can vary from short to long depending upon the current aspiration solution.

Central to the construction of ASA is a mechanism for keeping track of the best solution during the course of the procedure. This mechanism can also give signals to the procedure to increase the temperature if required as described in the next section.

3. An Adaptive Finite-Time Cooling Schedule

A number of cooling schedules have been reported in the literature (Otten and van Ginneken, 1984; Lam and Delosme, 1986 and Huang *et al.*, 1986). However, there has always been an open question as to how fast the simulation should be 'cooled', i.e. the question of the length of Markov chains and how much the temperature may be decreased to achieve convergence to the global minimum. Different arguments have been addressed in different cooling schedules. In our proposed adaptive cooling schedule some of the annealing parameters proposed by Dekkers and Aarts (1991) are changed based on what we believe to be plausible (though as yet possibly heuristic) reasoning. Suppose, at the start of a particular temperature level T_t the aspiration value is given by f_a. Adjustments are made to the annealing parameters in the following way.

3.1. Length of the Markov chain (L_t^a)

As mentioned above no generally acceptable solution has been presented for the 'inner-loop criterion', which decides how many 'local move-iterations' are required at each temperature. The optimal value of this parameter, which has to depend on the problem size, can not be determined in a rigorous way. Dekkers and Aarts (1991) choose the value to be

$$L_t = 10n \ , \tag{5}$$

where n is the problem dimension. Rather than allowing L_t to depend on the problem dimension only it would be more sensible to link it with the topography of f in some way to provide additional information to the procedure. Therefore, in our implementation, we determine the length of Markov chain in an adaptive way that depends on the starting solution, f_{x_t}, and the aspiration value, f_a, for the t-th Markov chain. In fact, we define

$$L_t^a = L_t + \mathbf{Int}\big[L_t F\big] \ , \tag{6}$$

where, $F = 1 - \exp\left(-(f_{x_t} - f_a)\right)$. Clearly L_t^a tends to L_t as f_{x_t} tends to f_a and tends to $2L_t$ when $f_{x_t} - f_a$ tends to ∞, i.e., $L_t^a \in (L_t, 2L_t)$. This strategy allows the possibility of increasing the number of function evaluations at a given temperature as the difference between f_a and f_{x_t} increases, but, of course, the whole of the increased chain length will not be required if a point is found with function value less than or equal to f_a.

3.2. Decrement of the control parameter

Dekkers and Aarts (1991) find the new temperature T_{t+1} by

$$T_{t+1} = T_t \left(1 + \frac{T_t \ln(1+\delta)}{3\sigma(T_t)}\right)^{-1}, \tag{7}$$

where $\sigma(T_t)$ denotes the standard deviation of the values of the cost function at the points in the Markov chain at T_t. The constant δ is known as the distance parameter and determines the rate of decrease of the control parameter. In ASA the distance parameter δ is allowed to vary. If the length of the current Markov chain is small then we decrease the temperature level for the next Markov chain by a small amount and vice versa. This is achieved by setting

$$\delta = \delta_{\min} + (\delta_{\max} - \delta_{\min})\frac{V}{2L_t}, \tag{8}$$

where V is the number of iterations carried out so far in the current Markov chain and $\delta_{\min}$ and $\delta_{\max}$ are user supplied values. Whether the aspiration criterion is satisfied or not the distance parameter δ is calculated from the above rule so that the greater the number of trials the greater the decrement in temperature with $\delta = \delta_{\max}$ if $V = 2L_t$. However caution has to be taken when V is very small because then, even if the temperature is high, the number of acceptances will be small. Moreover, if the number of solutions generated at a high temperature is small and if they are close to each other then $\sigma(T_t)$ will be very small and, from (7), the next temperature will be reduced dramatically. Therefore, to safeguard a smooth temperature decrement we use $T_{t+1} = 0.95T_t$ when the number of acceptances $\leq n_a$, where n_a is a small positive integer. The main purpose of this new way of decreasing the temperature is to introduce a controlling influence from the number of iterations in the current Markov chain which, in turn, is influenced by the aspiration value.

3.3. Stopping Criteria

The last problem to be addressed is that of finding a criterion to terminate the annealing. A simple (but perhaps not very satisfactory) way consists in stopping when the objective function has not changed significantly over a reasonable number of temperature steps. This is the condition, $C1$, used by Dekkers and Aarts. In our

implementation we combine $C1$ with a further condition $C2$ which depends on the aspiration value,

$$C1 : \left| \frac{d\bar{f}_s(T_t)}{dT_t} \frac{T_t}{\bar{f}(T_o)} \right| < \varepsilon_s ,$$
$$C2 : |f_t^* - f_a| \leq \varepsilon_r . \tag{9}$$

Here $\bar{f}(T_o)$ is the mean value of f at the points in the initial Markov chain, $\bar{f}_s(T_t)$ (see Ali, 1994) is the smoothed value of $\bar{f}$ over a number of chains in order to reduce the fluctuations of $\bar{f}(T_t)$, ε_s and ε_r are small positive numbers and f_t^* is the final solution obtained in the t-th Markov chain, at the end of which $C1$ is satisfied. Notice that the condition $C1$ is satisfied only if T_t is very small and no improved solution is found over a number of chains. For termination of the ASA procedure at the end of the t-th Markov chain we check condition $C1$ and then $C2$. If condition $C1$ is satisfied but $C2$ is not then the algorithm starts again with the aspiration point as the starting point of the $(t+1)$-th Markov chain and a new aspiration value is found by updating the old one. The temperature level T_{t+1} is now increased by setting

$$T_{t+1} = \mu T_o , \tag{10}$$

where T_o is the initial temperature and μ satisfies $0 < \mu < 1$. (This is known as re-annealing (Ingber, 1989).) If we increase the temperature as above, this may, of course, introduce cycling in the algorithm, especially when the aspiration point gets stuck on a local minimizer. This could possibly be overcome by using

$$T_o^r = \frac{T_o}{2r} , \tag{11}$$

where T_o^r is the initial temperature for the r-th re-annealing. However, in most implementations of ASA a slightly better result was obtained by using

$$T_o^{r+1} = \mu T_o^r . \tag{12}$$

Obviously, in the first re-annealing $T_o^1 = \mu T_o$. When the algorithm stops the aspiration solution is taken as the optimal solution.

4. Numerical Results

In this section we compare ASA with SA numerically. The computing was carried out on the HP9000/870 computer at Loughborough University of Technology using the programming language PASCAL. The test functions were taken from Dixon and Szegö (1978), a set of commonly used functions in global optimization, together with four more complex, practical problems.

4.1. Choice of parameters

In our numerical work initial focus was on the different parameters of ASA. Except for δ, which is allowed to vary between its maximum and minimum values, the

values of the other parameters that are common to SA and ASA are kept the same as those suggested by Dekkers and Aarts (1991). Therefore for the cooling schedules of both SA and ASA we use the following common parameters: $\chi_o = 0.9$, $\varepsilon_s = 10^{-4}$ and $t_0 = 0.75$ but $\delta = 0.1$ was chosen for SA and $\varepsilon_r = 10^{-3}$ was chosen for ASA. These values for the common parameters including the value for δ were suggested by Dekkers and Aarts (1991). The value of $\delta_{\min}$ has to be a small number and therefore we took $\delta_{\min} = 0.05$. We also tested the effect of n_a by varying it from 2 to 5 and the results indicated that $n_a = 3$ was a reasonable value and so for our comparison purposes this was the value we used. We found the initial temperature T_o for both ASA and SA by generating $m_o = 10n$ solutions. For the generation mechanism, **GM**, we used steepest descent in the early stages and limited memory BFGS (version E04DGF from NAG), implemented for two iterations, in the later stages for the SA and ASA algorithms. If the current temperature level, T_t, falls below a certain fraction of the initial temperature, T_o, i.e. if $T_t \leq 0.05T_o$ or $T_t \leq 15$, BFGS is implemented. The effects of conditions $C1$ and $C2$ and the effects of μ have been thoroughly investigated in Ali (1994) and the results suggest that a good value for μ lies in the range $[0.05, 0.30]$. The incorporation of condition $C2$ also enhanced the robustness of the algorithm in finding the global minimum, this effect was particularly noticed in the practical problems (Ali, 1994). The investigation of the effects of $\delta_{\max}$ suggested that a good value for $\delta_{\max}$ lies in $[0.1, 1.0]$ with the overall best value equal to 0.8. For a full discussion of these effects including the complete set of results see Ali (1994).

5. Comparisons

In Table I, the results of comparing SA, ASA and ASA_{met} are shown, where ASA_{met} is the ASA algorithm with the Metropolis acceptance criterion. In this Table AVE represents the average result for all test functions for which the global minima were obtained, R the number of re-annealings, t the temperature counter, FE the number of function evaluations, cpu the cpu time and T_o and T_f respectively the initial and final temperatures.

From Table I it is clear that both implementations of the ASA algorithm perform much better than SA in terms of cpu and FE and, moreover, SA failed to locate the global minimum for GP. The difference between ASA and ASA_{met} is very small but the results do indicate that the changes made to the Metropolis acceptance criterion have made some improvement. Although in the event of re-annealing in ASA the temperature is increased to facilitate the global exploration of the area of interest Ω still the average number of temperature decrements for ASA is much less than that for SA. This improvement occurs because the aspiration value not only provides information as to whether to increase the temperature in order to search for a better solution but also helps in decisions to take meaningful up-hill steps.

Table I.

		SA				
FE	cpu	t	T_o	T_f	R	
1088	0.05	42	335.34	5.24E-5	(-)	BR
1102*	0.09	49	21815.39	5.24E-5	(-)	GP
1120	0.10	26	10.87	3.49E-5	(-)	S5
1122	0.12	26	11.16	3.49E-5	(-)	S7
1179	0.12	27	11.25	3.49E-5	(-)	S10
1252	0.18	38	5.42	5.24E-5	(-)	H3
1817	0.33	27	4.77	2.62E-5	(-)	H6
1263	0.15	31				AVE
		ASA				
383	0.06	16	335.34	2.39E-5	0	BR
1602	0.10	35	21815.39	5.73E-2	3	GP
592	0.12	15	10.87	7.50E-1	0	S5
596	0.09	15	11.16	7.50E-1	0	S7
598	0.11	16	11.25	7.50E-1	0	S10
607	0.09	19	5.42	1.48E-2	0	H3
602	0.17	11	4.77	8.12E-3	0	H6
711	0.11	18				AVE
		ASA_{met}				
500	0.04	17	335.34	6.0E-1	0	BR
1417	0.13	33	21815.39	9.73E-5	2	GP
635	0.10	16	10.87	2.32E-1	0	S5
639	0.12	16	11.16	2.33E-1	0	S7
639	0.13	16	11.25	2.34E-1	0	S10
633	0.12	21	5.42	2.12E-2	0	H3
808	0.24	16	4.77	1.70E-2	0	H6
753	0.13	19				AVE

* Local minimum found; ASA_{met} is ASA with Metropolis acceptance criterion

Notice that the stopping condition $C1$ must be satisfied for both SA and ASA and $C2$ is only used in ASA when $C1$ is already satisfied. This means that the larger values for T_f in ASA are the results of the changes made to SA other than the stopping condition. The low values for T_f in SA occur because at low temperatures the generation mechanism GM can cause the standard deviation $\sigma(T_t)$ in (7) to become very small.

ASA was also compared with several other currently available stochastic algorithms using cpu times, number of function evaluations and accuracy of the global

Table II.

		SA			ASA			
Problem	n	FE	CPU	f^*	FE	CPU	f^*	
TR	13		157931	2382.10	0.141	101357	1380.90	0.144
BC	10		171543	414400	9.641	166317	391517	9.640
PL	12		268568	1396.50	59.80	89265	572.30	59.84
MP†								
$Si(B)$	12		206166	315.71	-23.12	151109	175.52	-25.98
$Si(C)$	12		272927	254.92	-10.57	66274	47.63	-13.03
As	12		205151	319.80	-18.49	10462	149.67	-18.49

† Calculated for six particles (Ali, 1994)

minima as indicators. The results, which were found to be quite encouraging, are reported in full in Ali (1994).

6. Practical Problems

In addition to the above test functions we also considered four highly complex problems from practical fields of application. They are concerned with a stirred-tank reactor (TR), a bifunctional catalyst reactor (BC), the statistics of pig-liver behaviour (PL) and many-body potentials (MP) for silicon ($Si(B)$) and ($Si(C)$) and arsenic (As). The first two are optimal control problems from chemical engineering with two and twenty five local optima and global optimum values of 0.133 and 10.09 respectively. The number of local maxima for the third problem is not known but the global maximum is known to be 59.84. The last problem is the most difficult of all and has a large number of local minima and an unknown global minimum. This problem arises in the field of physical chemistry (Ali and Smith, 1993). Full details of these problems can be found in (Ali, 1994). In Table II we briefly present the best results obtained by SA and ASA for these problems. The full set of results can be found in Ali (1994). A discretization technique was introduced to deal with the differential equations in the optimal control problems (Ali, 1994).

From the above Table it is quite clear that both in terms of the number of function evaluations and cpu time ASA is much superior to SA. Moreover, ASA exhibited superiority over SA in finding good approximations to the global minima for all of the practical problems except BC. Further results also confirmed the competiveness of ASA with other recent stochastic algorithms, see Ali, 1994.

7. Conclusions

From our numerical comparisons in this paper it is clear that ASA is considerably superior to SA. It also offers a useful alternative to some other currently available

stochastic methods (Ali, 1994). For higher dimensional problems and for problems with many local minima SA-type algorithms may be necessary because the amount of data that has to be stored is negligible and no complete local searches are needed. Moreover, if the dimension or the number of local minima is increased, this has no effect on the amount of data stored. Therefore, in many situations the ASA algorithm will be preferable since it performs better than the original SA algorithm. However, further research may yield yet more efficient SA algorithms.

References

Ali, M. M. (1994), Some modified stochastic global optimization algorithms with applications, Ph.D Thesis, Loughborough University of Technology, Loughborough, Leicestershire, England.

Ali, M. M. and Smith, R. (1993), The Structures of small Clusters Ejected by Ion Bombardment of Solids, *Vacuum* **44**, No. 3/4, 377–379.

Ali, M. M. and Storey, C. (1994), Topographical Multilevel Single Linkage, *Journal of Global Optimization* **5**, 349–358.

Dekkers, A. and Aarts, E. (1991), Global Optimization and Simulated Annealing, *Mathematical Programming* **50**, 367–393.

Dixon, L. C. W. and Szegö, G. P. (eds.) (1978), *Towards Global Optimization 2*, North-Holland, Amsterdam, Holland.

Faigle, U. and Schrader, R. (1988), On the Convergence of Stationary Distributions in Simulated Annealing Algorithms, *Information Processing Letters* **27**, 189–194.

Floudas, A. and Pardalos, M. (eds.) (1992), *Recent Advances in Global Optimization*; Princeton University Press, U.S.A.

Greene, J. and Supowit, K. (1986), Simulated Annealing Without Rejected Moves, *IEEE Transactions on CAD of IC and Systems* **5**, No 1, 221–228.

Horst, R. and Tuy, H. (1990), *Global Optimization (Deterministic Approaches)*, Springer-Verlag, Berlin.

Huang, M. D., Romeo, F., and Sangiovanni-Vincentelli, A. (1986), An Efficient General Cooling Schedule for Simulated Annealing, *Proceedings IEEE International Conference on CAD*, ICCAD-86, Santa Clara, CA, pp. 381–384.

Ingber, L. (1989), Very Fast Simulated Re-annealing, *Mathematical and Computer Modelling* **12**, 967–973.

Lam, J. and Delosme, J. M. (1986), Logic Minimization Using Simulated Annealing, *Proceedings IEEE International Conference on CAD*, ICCAD-86, Santa Clara, CA, 348–351.

Otten R. H. J. M. and van Ginneken L. P. P. P. (1984), Floorplan Design Using Simulated Annealing, *Proceedings IEEE International Conference on CAD*, ICCAD-84, Santa Clara, CA, November, 1984, pp.96-98.

Ratschek, H. and Rokne, J. (1988), *New Computer Methods for Global Optimization*, Ellis Horwood, Chichester.

Rinnooy Kan, A. H. G. and Timmer, G. T. (1984), Stochastic Methods for Global Optimization, *American Journal of Mathematical and Management Sciences* **4**, 7–40.

Rinnooy Kan, A. H. G. and Timmer, G. T. (1987), Stochastic Global Optimization Methods; Part I : Clustering Methods, *Mathematical Programming* **39**, 27–56.

Rinnooy Kan, A. H. G. and Timmer, G. T. (1987a), Stochastic Global Optimization Methods; Part II: Multilevel Methods, *Mathematical Programming* **39**, 57–78.

Romeo, F. and Sangiovanni-Vincentelli, A. (1985), Probabilistic Hill Climbing Algorithms: Properties and Applications, *Proceedings on VLSI*, Chapel Hill, NC, pp 393-418.

Timmer, G. T. (1984), Ph.D Dissertation, Econometric Institute, Erasmus University, Rotterdam, Holland.

Törn, A. and Žilinskas, A. (1989), *Global Optimization*, Springer-Verlag, Berlin.

Törn, A. and Viitanen, S. (1992), Topographical Global Optimization, in *Recent Advances in Global Optimization*, C. A. Floudas and P. M. Pardalos (eds.), Princeton University Press, Princeton USA, pp. 384–398.