

# Best-so-far vs. where-you-are: implications for optimal finite-time annealing\*

Kenneth D. Boese and Andrew B. Kahng

UCLA Computer Science Department, Los Angeles,  
CA 90024-1596, USA

Received 2 December 1992

Revised 25 April 1993

*Abstract:* The *simulated annealing* (SA) algorithm is widely used for heuristic global optimization due to its high-quality results and its ability, in theory, to yield optimal solutions with probability one. Standard SA implementations use monotone decreasing, or ‘cooling’ temperature schedules that are motivated by the algorithm’s proof of optimality as well as by an analogy with statistical thermodynamics. In this paper, we challenge this motivation. The theoretical framework under which monotone cooling schedules are ‘optimal’ fails to capture the *practical* performance of the algorithm; we therefore propose a ‘best-so-far’ (BSF) criterion that measures the practical utility of a given annealing schedule. For small instances of two classic combinatorial problems, we determine annealing schedules that are optimal in terms of expected cost of the output solution. When the goal is to optimize the cost of the *last* solution seen by the algorithm (the ‘where-you-are’ (WYA) criterion used in previous theoretical analyses), we confirm the traditional wisdom of cooling temperature schedules. However, if the goal is to optimize the cost of the *best* solution seen over the entire algorithm execution (i.e., the BSF criterion), we give evidence that optimal schedules do not decrease monotonically toward zero, and are in fact *periodic* or *warming*. These results open up many interesting research issues, including the BSF analysis of simulated annealing and how to best apply hill-climbing to difficult global optimizations.

*Keywords:* Simulated annealing; stochastic hill-climbing; global optimization; traveling salesman problem; graph bisection.

*Correspondence to:* A.B. Kahng, UCLA Computer Science Department, Los Angeles, CA 90024-1596, USA.

\* Partial support for this work was provided by a GTE Graduate Fellowship and by ARO DAAK-70-92-K-0001, ARO DAAL-03-92-G-0050, NSF MIP-9110696, and NSF Young Investigator Award MIP-9257982.

## 1. Introduction

Given a finite solution set  $S$  and a cost function  $f: S \rightarrow \mathbb{R}$ , we may formulate global optimization as the search for  $s \in S$  such that  $f(s) \leq f(s') \forall s' \in S$ . Typically,  $|S|$  is very large compared to the number of solutions that can be reasonably examined in practice. Indeed, many important global optimization formulations are known to be intractable [4], so that efficient algorithms are unlikely. Therefore, general-purpose heuristics are of interest.

One of the most successful global optimization heuristics is *simulated annealing* (SA), which was proposed independently by Kirkpatrick et al. [10] and Cerny [3] and is motivated by analogies between the solution space of an optimization instance and microstates of a statistical thermodynamic ensemble. From the solution  $s_i \in S$  at the  $i$ th time step, the SA algorithm (Figure 1) generates a ‘neighbor’ solution  $s'$  and decides whether to adopt it as  $s_{i+1}$ , based on the cost difference  $f(s') - f(s_i)$  and the value of a *temperature* parameter  $T_{i+1}$ . (For each  $s_i$ , the set of possible neighbors  $s'$  is called the *neighborhood*  $N(s_i)$ ; together the neighborhoods  $N(s)$  for all  $s \in S$  induce a topology over  $S$  called its *neighborhood structure*.) Over the  $M$  steps for which the SA algorithm is executed, a *temperature schedule*  $T_1, T_2, \dots, T_M$  guides the optimization process. Typical SA practice uses a large initial temperature and a final temperature of zero, with  $T_i$  monotonically decreasing according to a fixed schedule or in order to maintain some measure of ‘thermodynamic equilibration’.

The SA algorithm enjoys certain theoretical attractions [11]. Using Markov chain arguments and basic properties of Gibbs–Boltzmann statistics, one can show that for any finite  $S$ , SA will converge to a globally optimal solution given infinitely large  $M$  and a temperature schedule that converges to

SA Algorithm Template	
0.	$s_0 \leftarrow$ random solution in $S$
1.	For $i = 0$ to $M - 1$
2.	Choose $s' \leftarrow$ a random element from $N(s_i)$
3.	<b>if</b> $f(s') < f(s_i)$
4.	$s_{i+1} \leftarrow s'$
5.	<b>else</b>
6.	$s_{i+1} \leftarrow s'$ with probability $e^{-[f(s')-f(s_i)]/T_{i+1}}$
7.	otherwise $s_{i+1} \leftarrow s_i$
8.	Return $s_M$
8a.	Return $s_i, 0 \leq i \leq M$ , such that $f(s_i)$ is minimum.

Fig. 1. The simulated annealing algorithm for a given bound of  $M$  time steps.

zero sufficiently slowly. In other words,

$$\Pr(s_M \in R) \rightarrow 1 \quad \text{as } M \rightarrow \infty, \quad (1)$$

where  $R \subset S$  is the set of all globally optimal solutions, so that SA is ‘optimal’ in the limit of infinite time. Several groups have proved that specific temperature schedules guarantee convergence of SA to a global optimum; e.g., Hajek [6] showed the optimality of ‘logarithmic cooling’ with  $T_i = a/\log(i + 1)$  when  $a$  is sufficiently large (see also [5, 15]).

## 2. Best-so-far vs. where-you-are

Theoretical analysis of annealing has always been performed with respect to a ‘where-you-are’ (WYA) implementation, where the algorithm returns whichever solution is last visited (line 8 of Figure 1). According to the theoretical analysis, it is this single solution  $s_M$  that in the limit  $M \rightarrow \infty$  has probability 1 of being optimal. On the other hand, a practical implementation will never ignore all of the solutions  $s_0, s_1, \dots, s_{M-1}$ : certainly, one can maintain the best solution seen so far, and return it if it is better than  $s_M$  (line 8a of Figure 1). We call this more realistic variant ‘best-so-far’ (BSF) annealing. Traditional convergence proofs for WYA annealing also apply to BSF annealing, since optimality according to equation (1) trivially implies optimality of the BSF variant. However, the results of Section 4 below show that BSF-optimal temperature schedules differ markedly from the traditional ‘cooling’ that is suggested by both the physical annealing analogy and the WYA analysis.

The extensive literature on simulated annealing contains little mention of either nonmonotone cooling schedules or BSF analysis. Lasserre et al.

[12] use a BSF implementation in comparing simulated annealing to other iterative optimization heuristics, but do not explore the implications of using BSF as opposed to WYA. A more direct reference to BSF is contained in [6], which establishes necessary and sufficient conditions for the infinite-time WYA optimality of monotone decreasing temperature schedules. Hajek briefly suggests [6, p. 315] a similar analysis for BSF: ‘It would be interesting to know the behavior of  $\min_{n \leq k} V(X_n)$  rather than the behavior of  $V(X_k)$ ’.

Noncooling schedules have been investigated by Hajek and Sasaki [7], who show the existence of a special class of optimization problems for which monotone cooling schedules are suboptimal. An ancillary result of [7] is that for neighborhood structures where the costs of any two neighboring solutions differ either by zero or a constant, there exists an optimal annealing schedule where all  $T_i$  are either 0 or  $+\infty$  (cf. our results showing optimal ‘periodic’ schedules in Section 4.2). While the BSF criterion is not mentioned in [7], the authors do suggest a similar measure of schedule quality, specifically, the expected number of time steps required to first encounter a solution with cost less than or equal to some prescribed constant  $c^+$ . Finally, our study of optimal *finite-time* schedules follows in the direction established by Strenski and Kirkpatrick [18].

## 3. Computing optimal finite-time schedules

Strenski and Kirkpatrick [18] use numerical methods to estimate optimal finite-time schedules according to an exact equation for computing expected WYA cost. We now state their technique and then extend it to compute optimal schedules according to the BSF criterion.

For any given finite schedule length  $M$ , an *optimal temperature schedule* is one for which simulated annealing has the lowest expected solution cost after  $M$  steps, assuming that all initial states  $s_0$  are equally likely. We use  $P(i)$  to denote the  $1 \times |S|$  row vector whose  $j$ th element  $[P(i)]_j$  gives the probability that solution  $s^j$  is the current solution at step  $i$ .<sup>1</sup> Because each  $s^j$  has equal probability of being the initial state, we have that  $P(0) = [1/|S|, 1/|S|, \dots, 1/|S|]$ . We let  $A(T_i)$  denote the  $|S| \times |S|$

<sup>1</sup> Here, we use superscripts to denote indices of particular solutions  $s^j \in S$ . We continue to use subscripts of solutions to denote time steps, so that  $s_i$  is the current solution at step  $i$ .

transition matrix induced by temperature  $T_i$ , i.e.,  $[A(T_i)]_{jk}$  equals the probability of moving from solution  $s^j$  to solution  $s^k$  in one step at temperature  $T_i$ . Thus, we can calculate each  $P(i)$  recursively as  $P(i) = P(i-1)A(T_i)$ . Let  $C$  be the  $|S| \times 1$  column vector of costs for solutions in  $S$ . Then the expected WYA cost  $E[f(s_M)]$  is equal to

$$E[f(s_M)] = P(0) \cdot A(T_1) \cdot A(T_2) \cdot \dots \cdot A(T_M) \cdot C. \quad (2)$$

To compute the expected BSF cost of a temperature schedule, we first sort the solutions  $s^j \in S$  in order of increasing cost  $f(s^j)$ , so that  $s^1$  is the optimal solution and  $s^{|S|}$  is the solution with highest cost. For each solution  $s^j$  and temperature  $T$ , we define a new transition matrix  $B^j(T)$  such that

$$[B^j(T)]_{k\ell} = \begin{cases} 1 & \text{if } k = \ell \text{ and } k \leq j, \\ 0 & \text{if } k \neq \ell \text{ and } k \leq j, \\ [A(T)]_{k\ell} & \text{if } k > j. \end{cases}$$

In other words, transitions in  $B^j(T)$  are the same as in  $A(T)$ , except that the only transition from a solution of cost less than or equal to  $f(s^j)$  is a self-move. In this way, each solution with cost less than or equal to  $f(s^j)$  becomes a ‘sink’ in transition matrix  $B^j(T)$ . We define  $P^j(i)$ , a  $1 \times |S|$  probability vector, such that  $P^j(0) = P(0)$  and  $P^j(i) = P^j(i-1)B^j(T_i)$ ,  $\forall i > 0$ . For instance,  $P^1(i)$  contains the probability distribution of solutions at step  $i$  if the global optimum  $s^1$  has been converted to a sink in all steps up to  $i$ .

We use  $d^j(i)$  to denote the probability of ever reaching a solution with cost  $f(s^j)$  or less within the first  $i$  steps. The value of  $d^j(i)$  is computed by

$$d^j(i) = \sum_{\ell=0}^i [P^j(i)]_{\ell}.$$

Note that  $d^1(M)$  is equal to the first element in  $P^1(M)$ , and so equals the probability of ever reaching the global optimum. For  $j > 1$ , the probability that  $s^j$  is the BSF solution after  $M$  steps is simply  $d^j(M) - d^{j-1}(M)$ , i.e., the probability of ever reaching a solution of cost  $f(s^j)$  or lower, minus the probability of ever reaching a solution of cost  $f(s^{j-1})$  or lower.<sup>2</sup> Thus, the expected cost of the BSF solution is equal to

$$d^1(M)f(s^1) + \sum_{j=2}^{|S|} [d^j(M) - d^{j-1}(M)]f(s^j).$$

<sup>2</sup> In the case where  $f(s^j) = f(s^{j-1})$ , we arbitrarily force the algorithm to return  $s^{j-1}$  as the BSF solution whenever a run visits both  $s^j$  and  $s^{j-1}$  during its execution.

Because the calculation of each  $d^j(M)$  requires the same number of matrix multiplications as  $E[f(s_M)]$ , the calculation of BSF cost is  $\Theta(|S|)$  times more expensive than the calculation of WYA cost. However, we also note that the BSF formula is still linear in each  $A(T_i)$  and  $B^j(T_i)$ , so that exact partial derivatives can be computed for use in numerical optimization.

In the experiments in Section 4, we select optimal annealing schedules with respect to a discrete set of possible values of  $T_i$ : in addition to  $T_i = 0$ , we also choose from among 100 evenly spaced temperature values  $> 0$ , such that the overall range  $[0, +\infty]$  is effectively represented.<sup>3</sup> For larger values of  $M$ , it is impossible to exhaustively enumerate all possible temperature schedules, and we therefore use an iterative method to generate locally optimal schedules. At each iteration, we test all possible perturbations of a single temperature  $T_i$  ( $i = 1, \dots, M$ ) to an adjacent temperature value above or below  $T_i$ , then deterministically adopt the single perturbation which yields the greatest improvement in expected solution cost. The search terminates when a locally optimal schedule is found.<sup>4</sup> For each estimation, we begin from several different initial schedules and report the best locally optimal schedule; we have observed very few distinct local optima, with almost all being qualitatively very similar.

#### 4. Experimental results: BSF-optimal schedules

In this section, we study small instances of two prominent combinatorial optimization problems – the traveling salesman problem and graph bisection. For each instance, we compute locally BSF- and WYA-optimal temperature schedules as described above.

<sup>3</sup> Our experiments have used  $T_i \in \{1, 2, \dots, 100\}$ , as well a range of values chosen such that the transition probabilities for an ‘average’ disimproving move are 0.01, 0.02, 0.03,  $\dots$ , 0.99  $\approx$  1.00. We found that results are qualitatively the same with either range of  $T_i$  values.

<sup>4</sup> Strenski and Kirkpatrick [18] also find locally optimal, rather than globally optimal, (WYA) schedules. Their method uses the partial derivatives of the expected WYA cost  $E[f(s_M)]$  with respect to each  $T_i$  to afford a gradient-descent method. Note that our WYA-optimal schedules in Section 4.2 are essentially identical to those obtained in [18] using gradient descent.

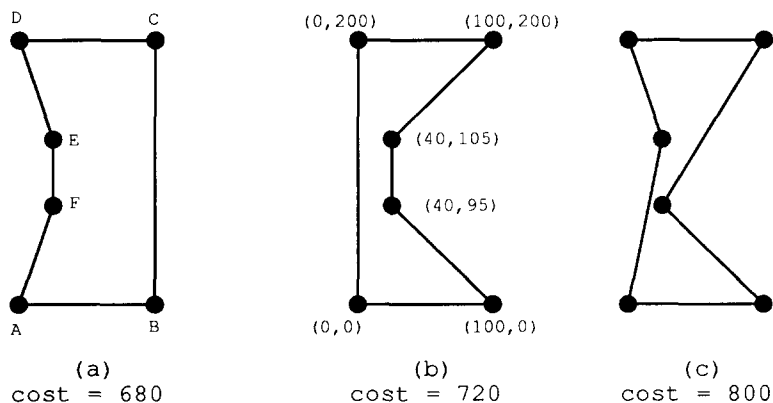


Fig. 2. Traveling salesman problem (TSP) instance with 6 cities in the Manhattan plane. The global optimum solution (a) and two local minimum solutions are depicted. The remaining local minimum solution is symmetric to (c), and is not shown.

#### 4.1. The traveling salesman problem

Our first set of experiments was performed on a small instance of the symmetric traveling salesman problem (TSP) [8, 13]. Given  $n$  cities and all  $n(n-1)/2$  intercity distances, the traveling salesman problem is to find a minimum-cost *tour*, i.e., a permutation of the cities such that the sum of the distances between adjacent cities in the tour, plus the distance between the first and last cities in the tour, is minimized.

We studied a 6-city TSP instance embedded in the Manhattan plane with city coordinates  $A = (0, 0)$ ,  $B = (100, 0)$ ,  $C = (100, 200)$ ,  $D = (0, 200)$ ,  $E = (40, 105)$  and  $F = (40, 95)$  (see Figure 2). For this instance,  $|S| = 5!/2 = 60$ , and there exists one globally optimal solution [ABCDEF(A)], along with three other locally optimal solutions [ABFECDA(A)], [ABECDF(A)], and [ABFCDE(A)]. We use the Lin 2-opt neighborhood operator that is usual in studies of the TSP [13]: a 2-opt move deletes two nonadjacent edges of the current solution  $s_i$  and then reconnects the two resulting paths into a new tour  $s'$ .

Figure 3 shows locally WYA-optimal and BSF-optimal temperature schedules for finite time bounds  $M = 40, 80$  and  $160$ . There is a clear contrast between the two criteria: the WYA-optimal schedules are monotone decreasing (as would be expected from the body of results in the current literature), while the BSF-optimal schedules are nearly *monotone increasing*, or *warming*.<sup>5</sup>

<sup>5</sup> Note that in BSF annealing, the last temperature  $T_M$  is irrelevant since an improving move  $s'$  will always be accepted no matter what the value of  $T_M$  might be.

Table 1 and Figure 4 compare the expected solution qualities of BSF- and WYA-optimal temperature schedules for a variety of schedule lengths. The difference in BSF quality between the BSF-optimal and WYA-optimal schedules is quite significant: the expected BSF quality achieved by the 80-step WYA-optimal schedule is almost matched by the BSF-optimal schedule of only 40 steps (!). The poor WYA quality of BSF-optimal schedules, particularly for larger values of  $M$ , may indicate the irrelevance in practice of optimizing  $f(s_M)$ .

We have also estimated WYA- and BSF-optimal schedules for a small six-node instance of the graph placement problem (see [2]) with qualitatively similar results. WYA-optimal schedules were monotone decreasing, while BSF-optimal schedules were monotone *increasing*.

#### 4.2. Graph bisection

We next studied a small instance of the graph bisection problem [9]. Given an edge-weighted graph  $G = (V, E)$ , the graph bisection problem is to find a partition of  $V$  into disjoint  $U$  and  $W$ , with  $|U| = |W|$ , such that sum of the weights of edges  $(u, w) \in E$  with  $u \in U, w \in W$  is minimized. We used the same highly-structured instance treated by Strenski and Kirkpatrick [18]; the instance consists of a complete graph of eight nodes, with edge weights calculated as shown in Figure 5(a). Each of the eight nodes is represented by a leaf in the height-3 binary tree shown in the figure; the edge between any two nodes has weight  $\alpha^k$ , where  $k$  is the height of the least common ancestor between the

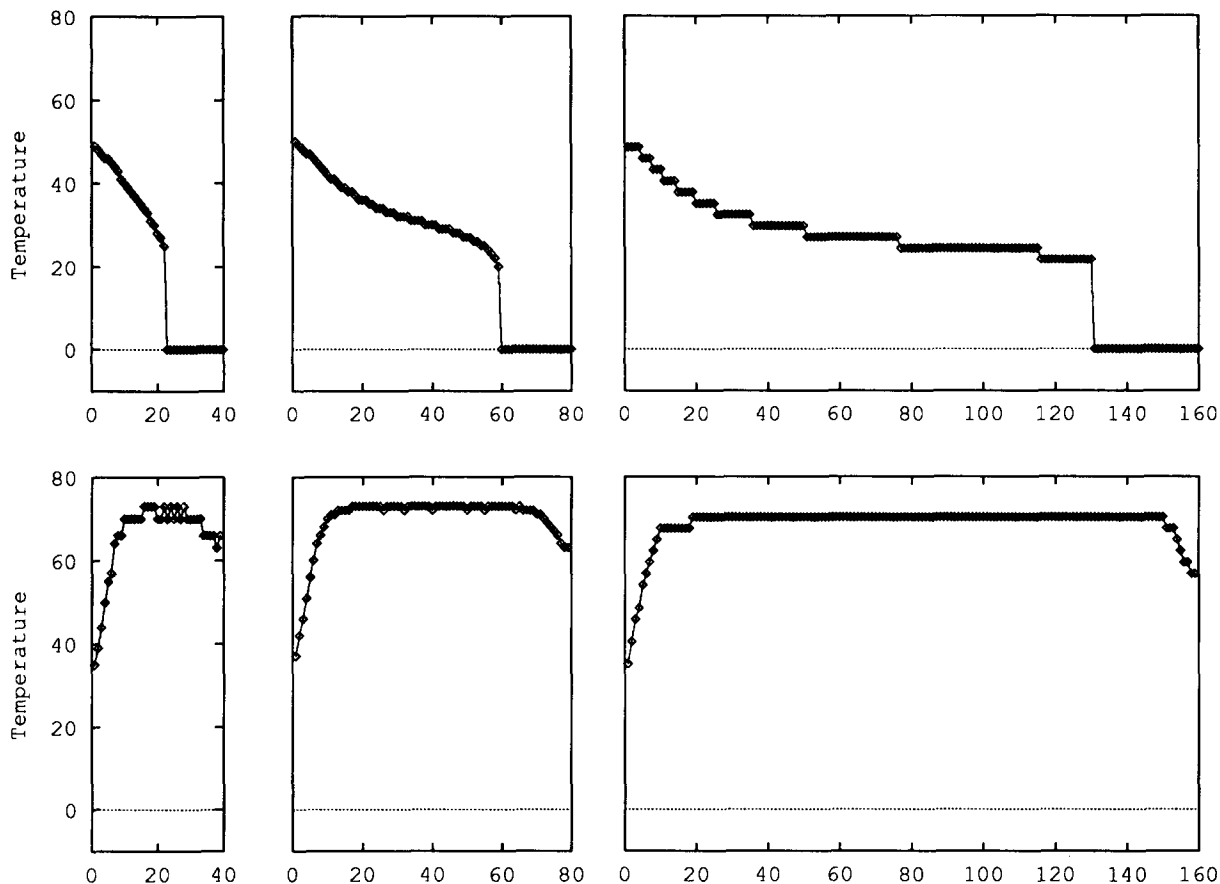


Fig. 3. Locally WYA-optimal (top) and BSF-optimal (bottom) temperature schedules of length 40, 80, and 160 steps for the 6-city TSP instance.

Table 1  
Expected BSF and WYA solution quality produced by locally optimal schedules for the 6-city TSP instance. We write the quality as a multiple of the cost of the global optimum solution

Number of steps	BSF-optimal schedule		WYA-optimal schedule	
	BSF quality	WYA quality	BSF quality	WYA quality
5	1.153	1.154	1.154	1.154
10	1.078	1.090	1.079	1.079
15	1.049	1.085	1.052	1.053
20	1.034	1.091	1.040	1.041
25	1.025	1.096	1.033	1.035
30	1.019	1.100	1.028	1.031
35	1.015	1.102	1.024	1.028
40	1.011	1.104	1.021	1.027
50	1.007	1.106	1.017	1.024
60	1.005	1.105	1.014	1.022
70	1.003	1.105	1.012	1.020
80	1.002	1.103	1.010	1.019

two nodes in the binary tree. Both our experiments and those of [18] use  $\alpha = 3$ . The globally optimum partition is  $\{1, 2, 3, 4\} \{5, 6, 7, 8\}$ , which corresponds to solution  $A$  in Figure 5(b) with cost  $f(A) = 16$ . Because of the symmetries in the edge weight construction, there are only five classes of equivalent-cost solutions; the multiplicities of these classes and their relative transition probabilities are both highly nonuniform, as seen in Figure 5(b).

Locally optimal schedules of  $M = 40, 80$  and 160 time steps are shown in Figure 6. Our WYA-optimal schedules almost exactly match the results of [18] and again confirm the traditional cooling intuition except in the first four steps, which have temperature 0. On the other hand, our locally BSF-optimal schedules are completely different, containing temperatures of only 0 and 100 (essentially  $+\infty$ ). These schedules are nearly *periodic*,

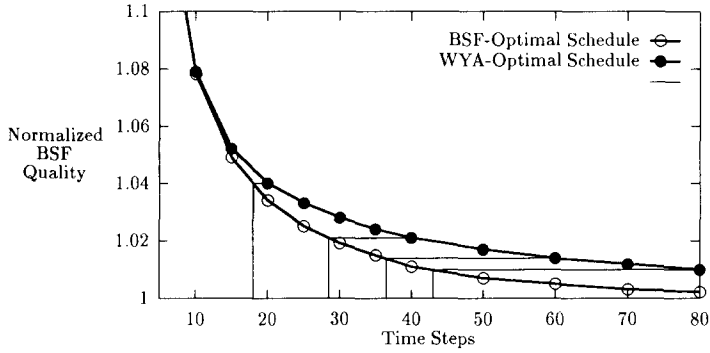


Fig. 4. Graphical representation of Table 1 data. A BSF-optimal schedule of approximately 42 time steps can achieve the same quality as a WYA-optimal schedule of length 80.

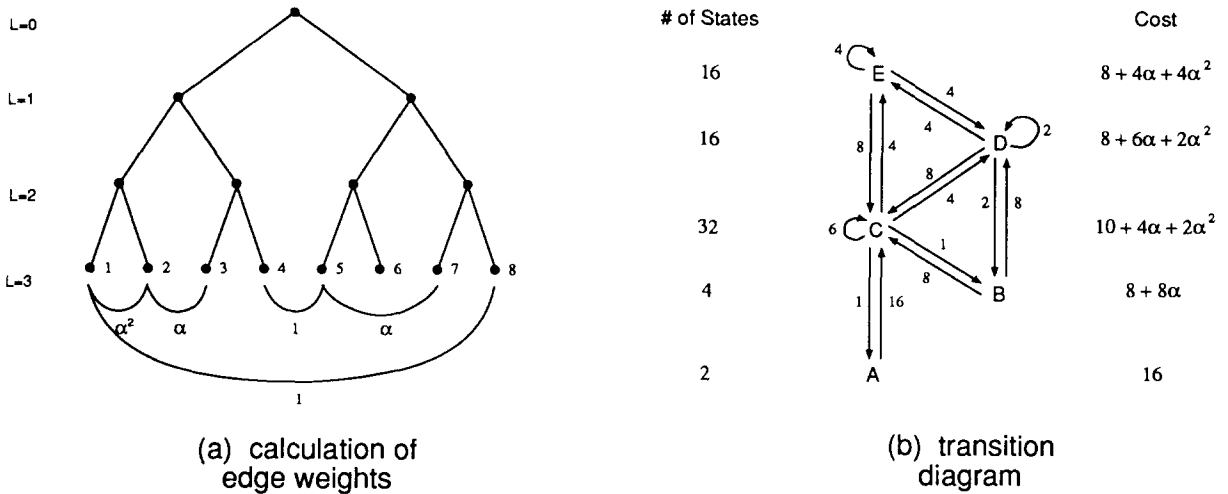


Fig. 5. Edge weight calculation and state transition diagram for the complete graph used as a bisection instance by Strenski and Kirkpatrick.

and are evocative of 'iterated descent' methodologies [1,8,12]. Table 2 compares the expected BSF and WYA solution costs for locally optimal schedules of various lengths. Again, optimal schedules in terms of the traditional WYA objective are clearly suboptimal when measured by their *practical*, BSF utility. As with the TSP instance discussed above, the BSF-optimal schedules are considerably more effective, using approximately 30% fewer steps to achieve the same expected BSF quality as the WYA-optimal schedules.

The eight-node graph bisection instance of [18] generalizes to complete graphs with  $2^k$  nodes (Strenski and Kirkpatrick find WYA-optimal schedules for only the eight-node instance). We

have also computed locally BSF-optimal and WYA-optimal temperature schedules for the 16-node instance with  $\alpha = 3$ , for which  $|S| = 28$ . While the WYA-optimal schedules are similar to those for the eight-node instance, the BSF-optimal schedules are no longer periodic, but are instead nearly monotonically increasing (similar to the BSF-optimal schedules for the TSP instance in Section 4.1).

**5. Conclusions**

The far-reaching consequences of BSF analysis are apparent even at first glance. For example, 'infinite-time optimality' holds for a much wider range of schedules under the BSF criterion: given

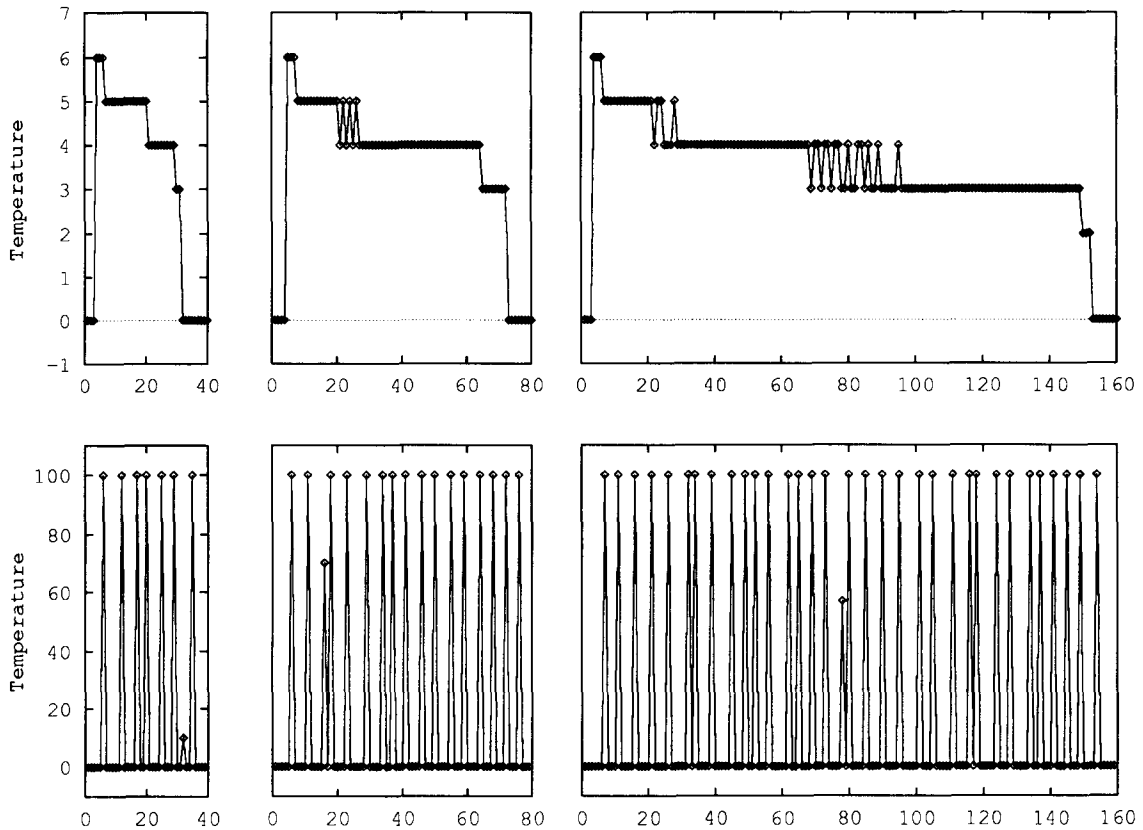


Fig. 6. Locally WYA-optimal (top) and BSF-optimal (bottom) temperature schedules of length 40, 80, and 160 steps for the eight-node graph bisection instance of Strenski and Kirkpatrick.

Table 2

Expected BSF and WYA solution quality produced by locally optimal schedules for the eight-node graph bisection instance of Strenski and Kirkpatrick. We write the quality as a multiple of the cost of the global optimum solution

Number of steps	BSF-optimal schedule		WYA-optimal schedule	
	BSF quality	WYA quality	BSF quality	WYA quality
5	2.038	2.038	2.038	2.038
10	1.780	1.981	1.797	1.797
15	1.603	1.978	1.661	1.675
20	1.475	1.920	1.548	1.586
25	1.379	2.036	1.462	1.514
30	1.306	1.967	1.396	1.453
35	1.248	2.039	1.343	1.402
40	1.202	1.969	1.298	1.359
50	1.135	1.970	1.231	1.289
60	1.090	2.032	1.181	1.236
70	1.063	2.048	1.143	1.195
80	1.041	2.044	1.112	1.164

infinite time, *any* schedule that is bounded away from zero will eventually visit a globally optimal solution, assuming that all solutions are reachable at nonzero temperatures. In this paper, we have explored the implications of finite-time, BSF analysis of the simulated annealing algorithm. The BSF analysis is more consistent with annealing practice than the traditional WYA analysis, and the study of *finite-time* annealing also reflects practical reality, since applications of the annealing algorithm are certainly limited to finite amounts of CPU time.

To assess the practical effect of the BSF criterion on finite-time annealing strategies, we have numerically estimated BSF- and WYA-optimal schedules for small instances of two classic problem formulations. Our results show striking differences between annealing schedules that optimize the two criteria. WYA-optimal schedules confirm the traditional regime of monotone cooling schedules. However,

BSF-optimal schedules are no longer monotone cooling, but are rather periodic or even warming. Moreover, our experiments show that the BSF criterion can yield tangible improvements in running times: BSF-optimal schedules required 30 to 50% fewer steps than WYA-optimal schedules to obtain the same expected BSF solution cost.

Although practitioners may argue that WYA and BSF solution qualities converge for longer schedules, we believe that such observations are due to the current experience with only cooling schedules. Our results show that while WYA-optimal schedules have very similar BSF and WYA solution qualities, the BSF-optimal schedules have comparatively poor expected WYA solution quality. The intuition behind BSF annealing is that 'reachability' and 'mobility' are critical to success; these factors are at odds with the WYA-optimal strategy, which reduces the temperature to zero in order to minimize  $f(s_M)$ . Put another way, all annealing schedules may be viewed as trading off between reachability among solutions (high  $T$ ) and a bias to lower-cost solutions (low  $T$ ). The WYA criterion forces the low-cost bias to dominate at the end of the run, while the BSF criterion may allow reachability to take precedence. It is therefore not surprising that BSF-optimal schedules may actually be 'warming' even at the end of the annealing execution (cf. our results for graph placement in [2]).

In conclusion, the BSF analysis opens the door to new hill-climbing regimes, as well as new theoretical fronts such as the Markov analysis of BSF annealing. A major area of future research lies in the application of nonconventional temperature schedules to larger problem instances. To this end, our experiments point to periodic 'iterated descent' methods [1, 8, 12] and adaptive construction of 'nonmonotone' (warming) schedules as being especially promising. We also believe that tuning BSF-optimal annealing strategies to the statistical parameters of optimization cost surfaces [17] will provide an important research direction.

### Acknowledgements

The authors wish to thank C.-W. Albert Tsao for his implementation of much of the code used in these experiments.

### References

- [1] E.B. Baum, Iterated descent: a better algorithm for local search in combinatorial optimization problems, in: D. Touretzky, ed., *Proc. Neural Information Processing Systems* (1987).
- [2] K.D. Boese, A.B. Kahng and C.W. Tsao, Best-so-far vs. Where-you-are: new directions in simulated annealing for CAD, UCLA CSD TR-920050, 1992.
- [3] V. Cerny, Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm, *J. Optim. Theory Appl.* **45**(1) (1985) 41–51.
- [4] M.R. Garey and D.S. Johnson, *Computers and Intractability: A Guide to the Theory of NP-Completeness* (W.H. Freeman, New York, 1979).
- [5] S. Geman and D. Geman, Stochastic relaxation, Gibbs distributions, and the Bayesian restoration of images, *IEEE Trans. on Pattern Analysis and Machine Intelligence* **6** (1984) 721–741.
- [6] B. Hajek, Cooling schedules for optimal annealing, *Math. Oper. Res.* **13**(2) (1985) 311–329.
- [7] B. Hajek and G. Sasaki, Simulated annealing – to cool or not, *Systems Control Lett.* **12** (1989) 443–447.
- [8] D.S. Johnson, Local optimization and the traveling salesman problem, *Proc. 17th Internat. Colloquium on Automata, Languages and Programming* (1990) 446–460.
- [9] D.S. Johnson, C.R. Aragon, L.A. McGeoch and C. Schevon, Optimization by simulated annealing: an experimental evaluation; part i, graph partitioning, *Oper. Res.* **37** (1989) 865–892.
- [10] S. Kirkpatrick, Jr. C.D. Gelatt and M. Vecchi, Optimization by simulated annealing, *Science* **220** (4598) (1983) 671–680.
- [11] P.J.M. Laarhoven and E.H.L. Aarts, *Simulated Annealing: Theory and Applications* (D. Reidel, Boston, 1987).
- [12] J.B. Lasserre, P.P. Varaiya and J. Walrand, Simulated annealing, random search, multistart or SAD?, *Systems Control Lett.* **8** (1987) 297–301.
- [13] E.L. Lawler, J.K. Lenstra, A. Rinnooy-Kan and D. Shmoys, *The Traveling Salesman Problem: A Guided Tour of Combinatorial Optimization* (Wiley, Chichester, 1985).
- [14] T. Lengauer, *Combinatorial Algorithms for Integrated Circuit Layout* (Wiley-Teubner, Berlin, 1990).
- [15] M. Lundy and A. Mees, Convergence of an annealing algorithm, *Math. Programming* **34** (1986) 111–124.
- [16] C. Sechen and A. Sangiovanni-Vincentelli, The timberwolf placement and routing package, *IEEE J. Solid-State Circuits* **20**(2) (1985) 510–522.
- [17] G.B. Sorkin, Efficient simulated annealing on fractal energy landscapes, *Algorithmica* **6** (1991) 367–418.
- [18] P. Strenski and S. Kirkpatrick, Analysis of finite length annealing schedules, *Algorithmica* **6** (1991) 346–366.