2000

Simulated annealing with an optimal fixed temperature

Mark James Fielding
University of Wollongong, mfieldin@uow.edu.au

Simulated annealing with an optimal fixed temperature

Abstract

Contrary to conventional belief, it turns out that in some problem instances of moderate size, fixed temperature simulated annealing algorithms based on a heuristic formula for determining the optimal temperature can be superior to algorithms based on cooling. Such a heuristic formula, however, often seems elusive. In practical cases considered we include instances of traveling salesman, quadratic assignment, and graph partitioning problems, where we obtain results that compare favorably to the ones known in the literature.

Keywords

temperature, annealing, fixed, simulated, optimal

Disciplines

Engineering | Science and Technology Studies

Publication Details

Fielding, M. (2000). Simulated annealing with an optimal fixed temperature. SIAM Journal On Optimization, 11 (2), 289-307.

SIAM J. OPTIM.
Vol. 11, No. 2, pp. 289–307

SIMULATED ANNEALING WITH AN OPTIMAL FIXED TEMPERATURE*

MARK FIELDING†

Abstract. Contrary to conventional belief, it turns out that in some problem instances of moderate size, fixed temperature simulated annealing algorithms based on a heuristic formula for determining the optimal temperature can be superior to algorithms based on cooling. Such a heuristic formula, however, often seems elusive. In practical cases considered we include instances of traveling salesman, quadratic assignment, and graph partitioning problems, where we obtain results that compare favorably to the ones known in the literature.

Key words. simulated annealing, fixed temperatures

AMS subject classifications. 90C27, 65K05

PII. S1052623499363955

1. Introduction.

It has been seen, in Cohn and Fielding [6], that simulated annealing with a suitably chosen fixed temperature performs well for many instances of the traveling salesman problem (TSP). Investigated here is the existence of a (problem dependent) optimal fixed temperature, T_{opt}, and how this depends on the time available and the quality of solution required. We determine optimal fixed temperatures, allowing a predetermined number of iterations, to yield near-optimal solutions, and we compare the performance of fixed temperature simulated annealing against that of Aarts' algorithm [1] based on a cooling schedule.

Optimal fixed temperatures are experimentally determined for instances of the TSP, quadratic assignment (QAP), and graph partitioning (GPP) problems. Predictability of the optimum fixed temperature is also considered in the given applications.

The potential usefulness of using a fixed temperature algorithm is mentioned in Kirkpatrick [13], where it is considered suitable for small problem instances. We find the performance of simulated annealing with a fixed temperature to deteriorate as the size of problems increases, while for some problem instances it turns out to perform better than a fast cooling schedule. It is shown in Connolly [7] that fixed temperature schedules outperform fast cooling schedules for both small and large instances of the QAP.

Let $f(x)$ denote the value of the *objective* (*cost*) *function* associated with *configuration* x (a feasible solution to the problem instance at hand). Let $f_{\min}$ be the value of the objective function associated with global minima. Let S be the set of all feasible solutions.

At each iteration of simulated annealing a random perturbation is made to the current solution, x, giving rise to a set, $N(x)$, of *neighbors*. A neighbor, $y \in N(x)$, is accepted as the next configuration, with probability $\exp(-(f(y) - f(x))/T)$, and x is to otherwise remain.

*Received by the editors November 8, 1999; accepted for publication (in revised form) April 18, 2000; published electronically September 27, 2000.
http://www.siam.org/journals/siopt/11-2/36395.html
†Department of Mathematics and Statistics, The University of Melbourne 3010, Victoria, Australia (markf@ms.unimelb.edu.au).

2. Selected applications.

2.1. The TSP. The TSP is commonly used in investigating the performance and behavior of simulated annealing. The TSP is used in two founding papers on simulated annealing, Kirkpatrick, Gellat, and Vecchi [12] and Černý [5] and is more thoroughly followed up in Kirkpatrick [13] and Aarts, Korst, and van Laarhoven [3].

In the TSP we are given n cities and our task is to find the shortest path through each and all of the cities, returning to the city where the path arbitrarily commenced. Between each pair of cities is given a distance, although this could be, for example, time or cost of travel. We will consider symmetric cases of the TSP, where the distance from one city to another is equal to the distance in the reverse. Distances may be, among others, Euclidean (calculated from Cartesian coordinates of the cities), geographic (calculated from coordinates on the earth's surface), or (specified) road distances.

Let $p(i)$ denote the ith city to be visited, and let $d_{c_1\,c_2}$ denote the distance between cities c_1 and c_2. The objective function is the length of a given path and is given by

$$f = \sum_{i=1}^{n-1} d_{p(i)p(i+1)} \; + \; d_{p(n)p(1)}.$$

2.1.1. Implementation details. In applying simulated annealing to the TSP, the following method will be used. The cities are numbered $1, \ldots, n$ arbitrarily. A feasible solution is a given path and is stored as an n-entry array, representing the order in which the cities are to be visited (with multiple array assignments corresponding to the same path). The total number of possible paths is

$$|S| = \frac{(n-1)!}{2}.$$

An initial path is given by an arbitrary sequence of the numbers 1 to n, and for our purposes it is chosen randomly.

A neighbor of a given path, for a symmetric TSP, is generated by choosing two cities randomly, say, the ith and the jth city visited, and reversing the order in which the cities $i+1$ (modulo n) up to j are visited. This is called a 2-*opt* move.

The 2-opt move leads to a neighborhood size of

$$|N(x)| = \frac{n(n-3)}{2},$$

if only neighbors that differ from the existing solution are allowed. (Choosing j as following or preceding i would result in no change to the path.) The following pseudocode gives a possible way of generating i and j:

```
k = integer(random()*n*(n-3));
i = (k mod n)+1;
j = ((i+1+(k div n)) mod n)+1;
```

it requires only one random number to generate each pair.

The distances, either explicitly given or calculated, are stored as a 2-dimensional array and are assigned integer values as instructed in TSPLIB [18]. The change in the objective function, resulting from a 2-opt move, is given by

$$\Delta f = -d_{p(i)p(i+1)} - d_{p(j)p(j+1)} + d_{p(i)p(j)} + d_{p(i+1)p(j+1)}.$$

TABLE 2.1
Selected instances of the TSP.

Instance	n	$f_{\min}$	Source
`gr48`	48	5046	[9], [19]
`kt57`	57	12955	[11]
`eil76`	76	538	[17], [19]
`kroA100`	100	21282	[14], [19]
`gr120`	120	6942	[8], [19]
`pr152`	152	73682	[17], [19]
`kroA200`	200	29368	[14], [19]
`pr264`	264	49135	[17], [19]
`lin318`	318	42029	[15], [19]
`gr442`	442	5069	[9], [19]

2.1.2. Selected TSP instances. Ten TSP instances have been selected, ranging from 48 to 442 cities. Among the instances chosen are those examined in Aarts and van Laarhoven [2]. The selected problem instances are shown in Table 2.1, along with the source of each problem. Except for `kt57`, these instances are available from the website TSPLIB [19]. Note that `lin318` is treated as a TSP rather than a Hamiltonian circuit. Also, `gr442` is taken in the form given in Grötschel and Holland [9] rather than as it appears in TSPLIB as `pcb442` (where it differs in scale and precision).

2.2. The QAP. The QAP occurs in many contexts, often considered as assigning n facilities to n locations. Let $p(i)$ denote the location to which facility i is to be assigned. Between each pair of facilities, i and j, there is given a *flow*, a_{ij}, and between each pair of locations, k and l, there is given a distance, b_{kl}. The objective function is given by

$$f = \sum_i \sum_j a_{ij}\, b_{p(i)p(j)}.$$

There may also be considered an additional cost $c_{i\,p(i)}$ of assigning facility i to location $p(i)$. This cost shall not be considered here. We shall also restrict our attention to symmetric cases of the QAP, that is, where the distance matrix is symmetric.

2.2.1. Implementation details. A feasible solution to the QAP is a given allocation of facilities to locations. This is stored as an n-entry array where the ith entry represents the location to which facility i is located, i.e., $p(i)$. The flow between facilities, and the distances between locations, are stored as 2-dimensional arrays. An initial allocation of facilities may be arbitrarily chosen, and for our purposes it is chosen randomly. The number of possible allocations totals

$$|S| = n!.$$

A neighbor, of a given solution x, is generated by randomly choosing two facilities, i and j, and interchanging their locations. This yields a neighborhood size of

$$|N(x)| = \frac{n(n-1)}{2}.$$

The change in the objective function simplifies to

$$\Delta f = 2 \sum_{k \neq i,j} (a_{kj} - a_{ki})\,(b_{p(k)p(i)} - b_{p(k)p(j)}).$$

TABLE 2.2
Selected instances of the QAP.

Instance	n	$f_{\min}$	Source
nug15	15	1150	[16], [4]
rou15	15	354210	[4]
nug20	20	2570	[16], [4]
nug30	30	(6124)	[16],[4]
kra30a	30	(88900)	[4]
wil50	50	(48816)	[21],[4]
wil100	100	(273038)	[21], [4]
sko100a	100	(152002)	[20],[4]

2.2.2. Selected QAP instances. Eight QAP instances have been gained from the website QAPLIB [4]. The instances are given in Table 2.2. The value given for $f_{\min}$ is given in brackets when it is only the best found solution in the literature, not a proven global minimum.

2.3. The GPP. In the GPP, a graph (V, E) is given, where V is a set of n vertices and E is the set of edges. With each edge (i, j) is assigned a weight. The aim is to allocate the vertices of the graph to a given number of equally sized groups in such a way that the sum of the weights of all edges crossing between groups is minimized.

A simple case of the GPP, considered here, is when a graph is to be partitioned into two equally sized groups, V_1 and V_2, and the objective function is given by the total number of edges crossing between the two groups.

In the application of simulated annealing to such GPPs, Johnson et al. [10] find it advantageous to extend the set of feasible solutions to allow unequal partitions while adding a penalty to the objective function to encourage equally sized groups. The updated objective function is given by

$$f = |\{(i,j) \in E : \ i \in V_1 \text{ and } j \in V_2\}| + \alpha(|V_1| - |V_2|)^2, \tag{2.1}$$

where α is a suitably chosen constant. This means that a solution which is near optimal according to the updated objective function need not be a (true) feasible solution. If the final solution is not feasible, then, repeatedly, a vertex in the larger group is to be transferred to the smaller such that the true objective function increases as little as possible, until the graph is equally partitioned.

Our investigation into the application of simulated annealing to the GPP shall be regarding its ability to locate a partition, equal or otherwise, that minimizes the updated objective function (2.1). We shall also fix α at 0.05, the value found suitable in [10].

2.3.1. Implementation details. The vertices of a graph are arbitrarily numbered $1, \ldots, n$. We have stored a graph as a 2-dimensional array, where the ith row lists the vertices that are connected by an edge to the ith vertex. The degree of a vertex is the number of edges joined to it. A given configuration is represented by an n-entry array with the ith entry denoting to which group vertex i has been allocated. The number of possible configurations is

$$|S| = 2^n,$$

which differs from the number of equally partitioned solutions.

TABLE 2.3
Generated instances of the GPP.

Instance	n	Average degree		$f_{\min}$
		Expected	Actual	
rand124	124	5	4.9	(58.8)
rand250	250	5	5.2	(126.0)
geom250	250	30	25.8	(248.8)
rand500	500	5	5.0	(223.0)
rand500b	500	10	10.2	(695.0)
geom500	500	20	19.0	(174.8)
geom500b	500	40	34.8	(578.8)
rand1000	1000	5	5.2	(483.0)
rand1000b	1000	20	20.3	(3410.0)
geom1000	1000	10	9.3	(30.0)

A neighbor of a given configuration is generated by randomly selecting a vertex, and swapping to which group it belongs. This yields a neighborhood size of

$$|N(x)| = n.$$

The resulting change in the objective function, moving vertex i from V_1 to V_2, is given by

$$\begin{aligned}\Delta f = &|\{j \in V_1 : \ (i,j) \in E\}| - |\{j \in V_2 : \ (i,j) \in E\}| \\ &- 4\,\alpha\,(|V_1| - |V_2| - 1).\end{aligned}$$

Note that $((|V_1| - 1) - (|V_2| + 1))^2 = (|V_1| - |V_2|)^2 - 4(|V_1| - |V_2| - 1)$.

2.3.2. Selected GPP instances. Ten GPP instances have been generated according to the method described in Johnson et al. [10], where two types of graphs are considered, *geometric* and *random*. In the geometric case, points (vertices) are randomly generated in a unit square, and any two distinct vertices within distance d from each other are allocated an edge. In the random case, edges are added between each pair of distinct vertices independently with probability p. In each case, d or p are chosen to achieve a specified expected average degree D,

$$p = \frac{D}{n-1},$$

and

$$d = \sqrt{\frac{D}{n\,\pi}}.$$

The generated problems are listed in Table 2.3. For these instances the global minima are not known, and the value for $f_{\min}$ for each problem instance is given in brackets to show that it is merely the best solution obtained here.

3. Fast cooling schedules. In practice, cooling schedules tend to 0 too quickly to allow reaching global minima solutions with probability even close to 1. Two fast cooling schedules are utilized here, Aarts' cooling schedule [1] and the geometric cooling schedule [12]. Aarts' cooling schedule is used to determine the number of iterations appropriate for reaching near optimal solutions. The geometric schedule has been selected as a simple, easy-to-implement cooling schedule, potentially to assist in determining suitable fixed temperatures.

Aarts [1]: Temperature is held fixed during each loop of $R = |N(x)|$ iterations. At the end of each loop the temperature is dropped according to the rule

$$T_{k+1} = T_k \Big/ \left(1 + \frac{T_k \log(1+\delta)}{3\sigma_k}\right)$$

for some small real number δ ($\delta = 0.1$ is recommended in [1]). Also, σ_k is the standard deviation of the values of the cost function observed during the kth loop of the algorithm.

Geometric [12]: Temperature is again held fixed during each loop of $R = |N(x)|$ iterations. At the end of each loop the temperature is dropped according to the rule

$$T_{k+1} = \alpha\, T_k$$

for some $\alpha < 1$ (say, $0.8 \leq \alpha < 1$). We found the number of iterations performed at each loop had little effect on the algorithm's performance, provided the value of α is adjusted appropriately to give the same overall rate of cooling.

An initial temperature, T_0, is typically determined to yield a specified acceptance of proposed moves, say, 95%. In this paper, T_0 is determined by trial and error and is specified for each problem instance considered.

4. Determining N. The number of iterations to be allowed in the fixed temperature algorithms is denoted by N.[1] We wish to choose an N, as in [6], for each problem instance, appropriately large for simulated annealing to find near optimal solutions.

We have chosen Aarts' algorithm to determine N, although alternative algorithms could have been used. Aarts' algorithm does not require the determining of the cooling parameter δ for each individual problem instance. 100 runs of Aarts' algorithm are performed with the parameter setting ($\delta = 0.1$) recommended by Aarts and others [1, 2, 3]. In choosing N, we consider the number of iterations taken until first visiting the best solution found in each run. The maximum of these is taken, after removing outliers. An outlier is taken as a value more than 1.5 times the interquartile range greater than the third quartile.

5. Searching for an optimal fixed temperature.

5.1. Which optimal fixed temperature? As discussed in [6], determining an optimal temperature schedule may depend on which optimality criterion is adopted. We discuss here various optimality criteria in determining an optimal fixed temperature schedule. The influence of these criteria is then investigated experimentally with a 100-city TSP instance.

An optimal fixed temperature may be chosen experimentally, by running simulated annealing a number of times at each of a number of fixed temperatures and determining which temperature is best, according to an appropriate optimality criterion.

If it is our goal to find a global minimum in the shortest possible time, then an optimal fixed temperature might be considered one yielding a stopping time with

$$\mathbb{E}[\tau(T_{\text{opt}})] = \min_{T \in [0,\infty]} \mathbb{E}[\tau(T)], \tag{5.1}$$

[1] Note that N has been used in denoting the set of neighbors of x, $N(x)$, so N is used here.

where $\tau(T)$ is the time until reaching a global minimum using a fixed temperature T and a random initial state and $[0, \infty]$ is the set of nonnegative reals in union with positive infinity. Note that $T = 0$ relates to *iterative improvement*, where only transitions to improved solutions are accepted, yielding $\mathbb{E}[\tau] = \infty$ (providing local minima that are not global minima exist), and $T = \infty$ is the case where all proposed transitions are accepted.

If a global minimum is unlikely to be reached in a reasonable time frame, then we may simply consider reaching near-optimal solutions. In such a case, τ in (5.1) might represent the time until reaching a configuration x with objective function $f(x) \leq f_{\min} + \epsilon$. The temperature that is deemed optimal might then depend on ϵ.

In Connolly [7] is demonstrated the existence of optimal fixed temperatures for a number of QAP instances. Such a temperature is identified by considering the best solution in a given number of iterations, N, and taking the fixed temperature that on average yields the best result. Connolly determines the fixed temperature to yield the optimal value of

$$\mathbb{E}\left[\min\{f(X_1), \ldots, f(X_{\mathrm{N}})\}\right], \tag{5.2}$$

where X_t is the configuration visited in the tth iteration. The temperature deemed optimal under such a scheme may then depend on the choice of N.

An optimal fixed temperature may alternatively be considered one which yields the maximum probability of reaching a global (or near global) minimum within N iterations. That is,

$$P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) = \max_{T \in [0,\infty]} P(\tau(T) \leq \mathrm{N}). \tag{5.3}$$

If after N iterations the temperature is to be set to zero, allowing the algorithm to quickly settle in a local minimum, then this may also influence which temperature is deemed optimal.

5.2. A 100-city TSP example. We now investigate the influence of the above-mentioned criteria in determining an optimal fixed temperature for the 100-city TSP, `kroA100`. We determine optimal fixed temperatures according to (5.1), with respect to reaching global minima, and within 1 and 2 percent of global minima. We estimate the three corresponding values for $\mathbb{E}[\tau(T_{\mathrm{opt}})]$. With N set to each of these values we then go on to determine optimal fixed temperatures according to (5.2) and (5.3). These results are obtained by running 100 runs of simulated annealing at each of a number of fixed temperatures, under each scheme. The range of temperatures to be considered is determined experimentally, with temperatures outside this range yielding increasingly inferior results. The results of these runs are shown graphically in Figure 5.1.

In determining optimal fixed temperatures according to (5.1), the results are summarized as follows:

Reaching global minimum,

$$T_{\mathrm{opt}} \in (38, 41), \quad \mathbb{E}[\tau(T_{\mathrm{opt}})] \approx 100{,}000{,}000.$$

Reaching within 1% of global minimum:

$$T_{\mathrm{opt}} \in (45, 50), \quad \mathbb{E}[\tau(T_{\mathrm{opt}})] \approx 2{,}000{,}000.$$

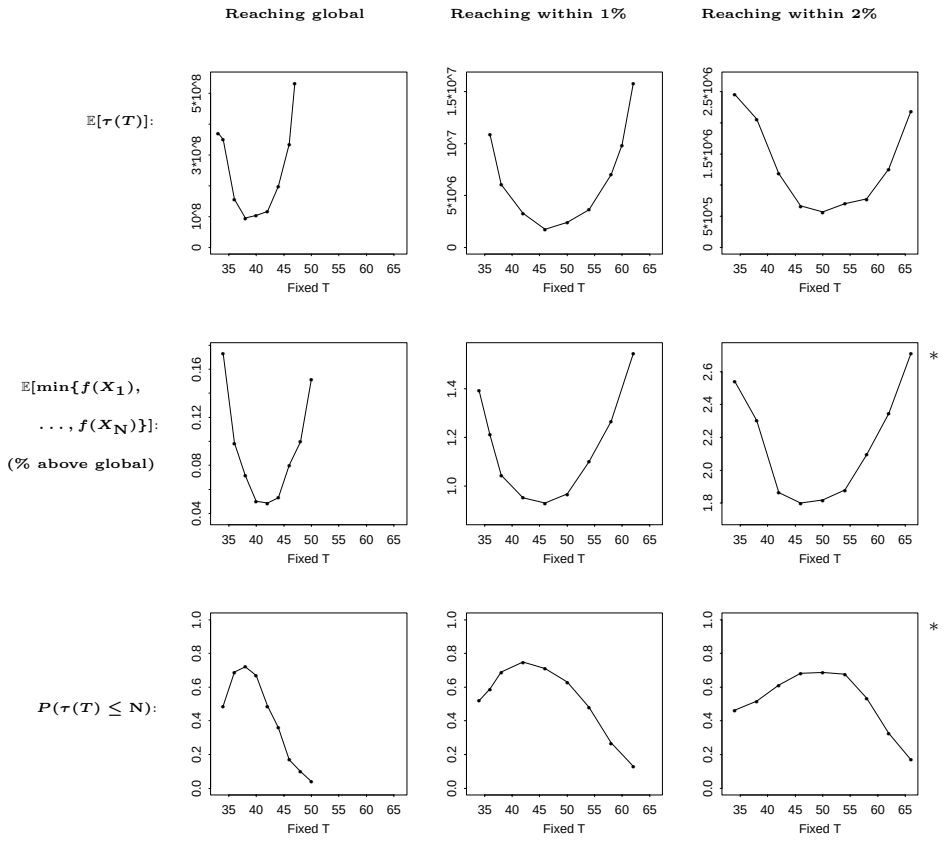

FIG. 5.1. *Locating the optimal fixed temperature for* `kroA100` *under various criteria, with respect to times until reaching global minima and within* 1% *and* 2% *of global minima. Graphs show estimates based on* 100 *runs (*500 *runs) of simulated annealing at each of the various fixed temperatures.*

Reaching within 2% of global minimum:

$$T_{\text{opt}} \in (48, 53), \quad \mathbb{E}[\tau(T_{\text{opt}})] \approx 600{,}000.$$

It is apparent that the temperature found optimal according to (5.1) depends on the required quality of solution.

In determining optimal fixed temperatures according to (5.2), the runs of simulated annealing are repeated, each terminating after first setting the temperature to zero after N iterations, for the respective values of N. The results are summarized as follows:

For N = 100,000,000,

$$T_{\text{opt}} \in (40, 43), \quad \mathbb{E}\left[\min\{f(X_1), \ldots, f(X_N)\}\right] \approx 1.0005\, f_{\min}.$$

For N = 2,000,000,

$$T_{\text{opt}} \in (43, 46), \quad \mathbb{E}\left[\min\{f(X_1), \ldots, f(X_N)\}\right] \approx 1.009\, f_{\min}.$$

For N = 600,000,

$$T_{\text{opt}} \in (45, 50), \quad \mathbb{E}\left[\min\{f(X_1), \ldots, f(X_N)\}\right] \approx 1.018\, f_{\min}.$$

It is apparent that the optimal temperature according to (5.2) depends on the choice of N.

Figure 5.1 also shows estimates for $P(\tau(T) < \mathrm{N})$, where $\tau(T)$ is the time until reaching the global minimum and within 1 and 2 percent of the global minimum, for N set to 100,000,000, 2,000,000, and 600,000,000, respectively. The results are summarized as follows:

For N = 100,000,000 and τ the time until reaching the global minimum,

$$T_{\mathrm{opt}} \in (36, 38), \quad P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) \approx 0.71.$$

For N = 2,000,000 and τ the time until reaching within 1% of the global minimum,

$$T_{\mathrm{opt}} \in (40, 44), \quad P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) \approx 0.77.$$

For N = 600,000 and τ the time until reaching within 2% of the global minimum,

$$T_{\mathrm{opt}} \in (45, 50), \quad P(\tau(T_{\mathrm{opt}}) \leq \mathrm{N}) \approx 0.69.$$

It is apparent that the temperature found optimal according to (5.3) depends on the required quality of solution.

6. Predicting the optimal fixed temperature.

6.1. The TSP. For the TSP, values for T_{opt} are given in Table 6.1, along with various parameters which may assist in predicting T_{opt}. The optimal fixed temperatures are determined experimentally by running 100 trials of simulated annealing at each of a number of fixed temperatures, using a fixed number of iterations, N, and noting, or interpolating to determine, the temperature which yields the best solution on average. Aarts' algorithm is used to determine N, as described in section 4, using an initial temperature T_0.

TABLE 6.1

Parameters to help predict T_{opt}, for various instances of the TSP. Values for T_{opt} are experimentally determined allowing N *iterations.*

Instance	T_0	N	T_{opt}	n	$f_{\min}$
gr48	2800	509760	20	48	5046
kt57	6000	857223	40	57	12955
eil76	200	1795441	1.4	76	538
kroA100	11700	4243750	46	100	21282
gr120	2900	7104240	11	120	6942
pr152	44500	14640064	75	152	73682
kroA200	11800	29509991	34	200	29368
pr264	32500	67095121	37.5	264	49135
lin318	11800	102173400	26	318	42029
gr442	2420	242935584	2.2	442	5069

	$\epsilon\%$	χ	f_{best}	T_{best}
gr48	3.5	0.0104	5088	14.2
kt57	3.9	0.0082	13021	18.2
eil76	4.2	0.0040	559	0.8
kroA100	3.7	0.0043	21449	16.5
gr120	3.6	0.0030	7156	2.6
pr152	3.0	0.0031	74983	18.3
kroA200	4.6	0.0025	29890	7.7
pr264	2.4	0.0007	50156	15.6
lin318	4.4	0.0013	43220	3.2
gr442	3.8	0.0005	5223	1.0

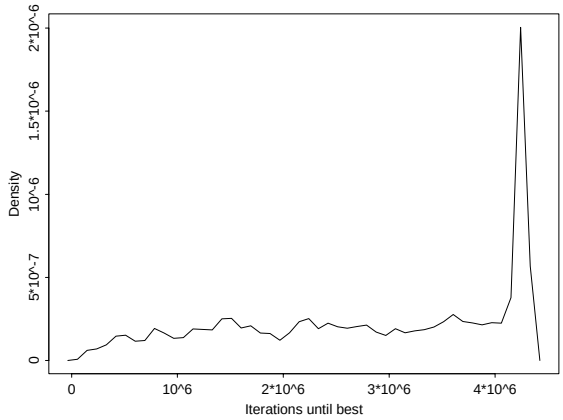

FIG. 6.1. *Iterations until reaching the best solution in each run of* 100 *runs performed of fixed temperature simulated annealing using* T_{opt}*, for* `kroA100`*.*

For each problem instance, estimates for the acceptance ratio, $\chi = \chi(T_{\text{opt}})$, are given at the optimal fixed temperature, along with $\epsilon\%$, the average value of the objective function (as percentage above global minima) observed at this temperature.

There is some uncertainty as to how estimates for χ and $\epsilon\%$ would best be gained. If (quasi) equilibrium[2] is not easily obtained at the optimal fixed temperature, then differing methods may yield differing estimates. For example, running a cooling schedule before continuing with a fixed temperature, and then observing average behavior, may yield different results to a usual fixed-temperature algorithm. By looking at the occurrences of the best solution found in each run it is seen that this is, roughly, consistently likely to occur throughout the course of a fixed temperature algorithm, suggesting that equilibrium at T_{opt} is easily obtained. This is seen in Figure 6.1, with a smoothed histogram of the iterations taken until reaching the best found solution in each fixed temperature run for the problem instance `kroA100`, using $T = T_{\text{opt}}$. A peak at N = 4243750 iterations corresponds to the temperature being set to zero. It is apparent that a *burn-in* period of up to 1 million iterations is required, to allow the algorithm to exhibit equilibrium.

Results for $\epsilon\%$ and χ are obtained after allowing a burn-in period of $1,000$ loops of $R = |N(x)|$ iterations. Upon reaching this many iterations the algorithms are continued for $1,000$ further loops in gaining estimates. These values are merely obtained in order to investigate the potential for such parameters in determining an optimal fixed temperature. In practice, more efficient means of estimation may be used.

From the 10 problem instances considered, the average value of the objective function is 3.7% above the global minimum. The average acceptance ratio is 0.0038, with the desirable value for χ tending to decrease as the size of the TSP instances increases.

Also given in Table 6.1 is the temperature, T_{best}, at which the best visited solution, f_{best}, occurred in one run of simulated annealing based on a geometric cooling schedule, with cooling parameter $\alpha = 0.95$ and initial temperature T_0. It was deter-

[2]It is described in Černý [5] how, at a fixed temperature, simulated annealing will tend toward an equilibrium, where the average of observed values of the objective function tends toward an equilibrium value.

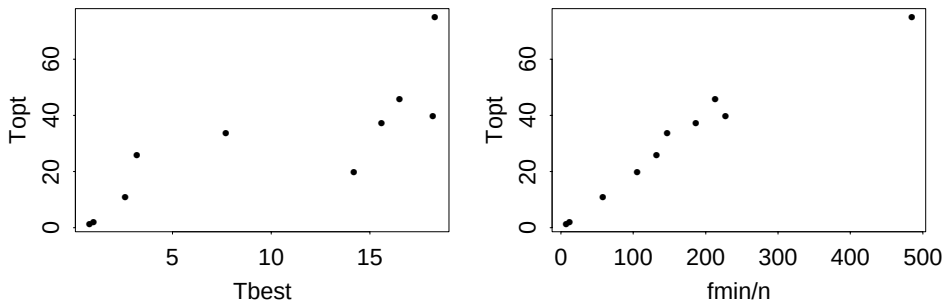

FIG. 6.2. *Relationship between T_{opt} and T_{best}, as well as between T_{opt} and f_{min}, for the TSP.*

mined by Connolly [7] that a suitable fixed temperature may be chosen by running simulated annealing with a fast cooling schedule and observing the temperature at which the best solution found first occurred. Indeed, while Connolly suggests using his values for T_{best} exactly, we find only that there exists a weak relationship between T_{opt} and T_{best} (see Figure 6.2).

In using T_{best} to make predictions of T_{opt}, the results of the problem instances yield on average

$$\hat{T}_{\text{opt}} = 3.4 \times T_{\text{best}}.$$

A reliable way to predict the optimal fixed temperature in the TSP is by a formula developed next (and noted in [6]). From Figure 6.2, it is apparent that a strong relationship exists between T_{opt} and the value of

$$\frac{f_{\text{min}}}{n}.$$

The ratios between T_{opt} and f_{min}/n are

$$0.190,\ 0.176,\ 0.198,\ 0.216,\ 0.190,\ 0.155,\ 0.232,\ 0.201,\ 0.197,\ 0.192,$$

and on average we have

$$\hat{T}_{\text{opt}} = 0.19 \times \frac{f_{\text{min}}}{n}. \tag{6.1}$$

In practice, the optimal solution to a given problem instance need not be known, and Table 6.1 also yields on average

$$\hat{T}_{\text{opt}} = 0.19 \times \frac{f_{\text{best}}}{n}.$$

It is interesting to note that the relationship in (6.1) describes a link between the optimal fixed temperature and the average link length in the global minimum solution, with

$$f = \sum_{i=1}^{n-1} d_{p(i)p(i+1)} \ + \ d_{p(n)p(1)}.$$

TABLE 6.2
Predictions of the optimal fixed temperature for various instances of the TSP.

Instance	T_{opt}	by (6.1)	by T_{best}	by $\epsilon\%$	by χ
gr48	20	20	48	21	8
kt57	40	43	62	40	28
eil76	1.4	1.3	2.7	1.3	1.3
kroA100	46	40	56	47	42
gr120	11	11	8.8	11	12
pr152	75	92	62	85	85
kroA200	34	22	26	32	44
pr264	37.5	35	53	42	60
lin318	26	25	11	21	44
gr442	2.2	2.2	3.4	2.2	5.3

Note that the change in objective function due to a 2-opt move is given by the subtraction and addition of links,

$$\Delta f = -d_{p(i)p(i+1)} - d_{p(j)p(j+1)} + d_{p(i)p(j)} + d_{p(i+1)p(j+1)},$$

and the optimal fixed temperature appears proportional to the average link length in the optimal path.

The transition probabilities between solutions depend on the ratio of the change in path length to the temperature:

$$P(\text{transition accepted}) = \exp(-\Delta f/T).$$

The long-term probability of being found in particular configurations, by the theory of Markov chains, is determined by such transition probabilities. This may give insight into developing approximate formulae in determining suitable fixed temperatures in other applications.

6.2. Performance of T_{opt} predictions in the TSP. To assess the merits of T_{opt} predictions, given in Table 6.2, a χ^2 *goodness-of-fit* test will be employed,

$$\sum \frac{(o-e)^2}{e} \stackrel{\text{d}}{=} \chi^2_{k-1},$$

where o are the observed values of T_{opt} and e are the expected values under a given method of prediction. The value obtained is to be compared against a χ^2 probability distribution with $k-1$ degrees of freedom, with k the number of values considered. The parameters χ and ϵ are used to predict T_{opt} by experimentally determining the temperatures yielding the same values that were observed on average.

For the four methods of prediction considered, (6.1), and by T_{best}, $\epsilon\%$, and χ, the χ^2 values obtained are 5.76, 58.6, 3.05, 44.7, respectively. The 5% upper tale of the χ^2_9 distribution occurs after 16.9, so that only (6.1), or by using $\epsilon\%$, yields satisfactory results.

It is apparent that predictions based on χ tend to be too small for the smaller instances while too large with the larger instances. The desirable value for χ is seen to decrease as the size of problems increases, so that using only an average value for χ would be expected to lead to such a result. Indeed, we shall see that different applications, as well as different problem structures, have different desirable values for χ. This will also be seen to be the case with $\epsilon\%$, which suggests that monitoring χ and $\epsilon\%$ would not generally be suitable in determining a suitable fixed temperature.

TABLE 6.3

Performance of the optimal fixed temperature for various instances of the TSP, against Aarts' algorithm. Percentage above global minimum of best solution on average in 100 *runs of the algorithms. Parentheses are used with repeated values due to accurate predictions.*

	% above global		
Instance	Aarts	with T_{opt}	by (6.1)
gr48	0.93	0.20	(0.20)
kt57	0.99	0.59	0.68
eil76	2.26	0.39	0.39
kroA100	0.78	0.55	0.60
gr120	1.83	0.85	(0.85)
pr152	0.73	0.59	0.68
kroA200	1.40	1.58	1.66
pr264	1.11	0.84	0.86
lin318	1.73	2.20	2.28
gr442	1.66	1.99	(1.99)

We gauge the performance of of fixed-temperature simulated annealing, and (6.1), against Aarts' algorithm. Table 6.3 gives the results of 100 runs of each algorithm. The fixed temperature algorithms are allowed N iterations, as determined with Aarts' algorithm. The given values are subject to variation, and of interest is the difference between using T_{opt} and (6.1) compared with using Aarts' algorithm.

It can be seen that as the size of problem instances increases, the relative performance of fixed-temperature algorithms decreases. For the smaller instances, fixed-temperature algorithms show to perform far better. Using (6.1) to predict values for T_{opt} yields consistently favorable results.

Although (6.1) closely describes the optimal temperatures for the given instances, it need not perform as well for TSP instances in general. It is the experience of the author, however, that (6.1) performs well in determining suitable fixed temperatures for TSPs in general.

For the TSP, fixed-temperature simulated annealing shows to outperform fast cooling in problems of less than 150 cities.

6.3. The QAP. For the QAP, values for T_{opt}, along with various parameters to assist in predicting T_{opt}, are given in Table 6.4, as carried out with the TSP.

The average value for $\epsilon\%$ is 3% and for χ is 0.03. Again, the value for χ tends to decrease as the size of problems increases, but this tends to be the case here also with $\epsilon\%$. Predictions of T_{opt} based on $\epsilon\%$ and χ will therefore not be further considered for the QAP.

In Figure 6.3, a strong relationship between T_{opt} and $f_{\min}/n$ is not found with the QAP. We consider instead insight gained from the case with the TSP. For the QAP, the objective function is given by

$$f = \sum_i \sum_j a_{ij}\, b_{p(i)p(j)},$$

which relates the average value of $a_{ij}\, b_{p(i)p(j)}$ times the square of the size of the problem instance. The change in objective function relates to the addition and subtraction of terms $a_{ij}\, b_{p(i)p(j)}$, with, when interchanging facilities i and j,

$$\Delta f = 2 \sum_{k \neq i,j} \left[-a_{ki}\, b_{p(k)p(i)} - a_{kj}\, b_{p(k)p(j)} + a_{ki}\, b_{p(k)p(j)} + a_{kj}\, b_{p(k)p(i)} \right].$$

TABLE 6.4

Parameters to help predict T_{opt}, for various instances of the QAP. Values for T_{opt} are experimentally determined allowing N *iterations.*

Instance	T_0	N	T_{opt}	n	$f_{\min}$
nug15	360	15691	8.0	15	1150
rou15	96000	13627	2700	15	354210
nug20	525	35360	9.5	20	2570
nug30	780	121313	10.5	30	(6124)
kra30a	16500	122621	300	30	(88900)
wil50	1550	568395	12	50	(48816)
wil100	2700	3894148	24	100	(273038)
sko100a	2550	3824669	18	100	(152002)

	$\epsilon\%$	χ	f_{best}	T_{best}
nug15	4.1	0.048	1174	6.6
rou15	7.0	0.049	371986	999.2
nug20	3.3	0.033	2604	3.3
nug30	1.9	0.022	6180	4.2
kra30a	5.6	0.048	91000	126.2
wil50	0.5	0.015	49014	3.0
wil100	0.6	0.018	273674	6.0
sko100a	0.6	0.009	152532	1.6

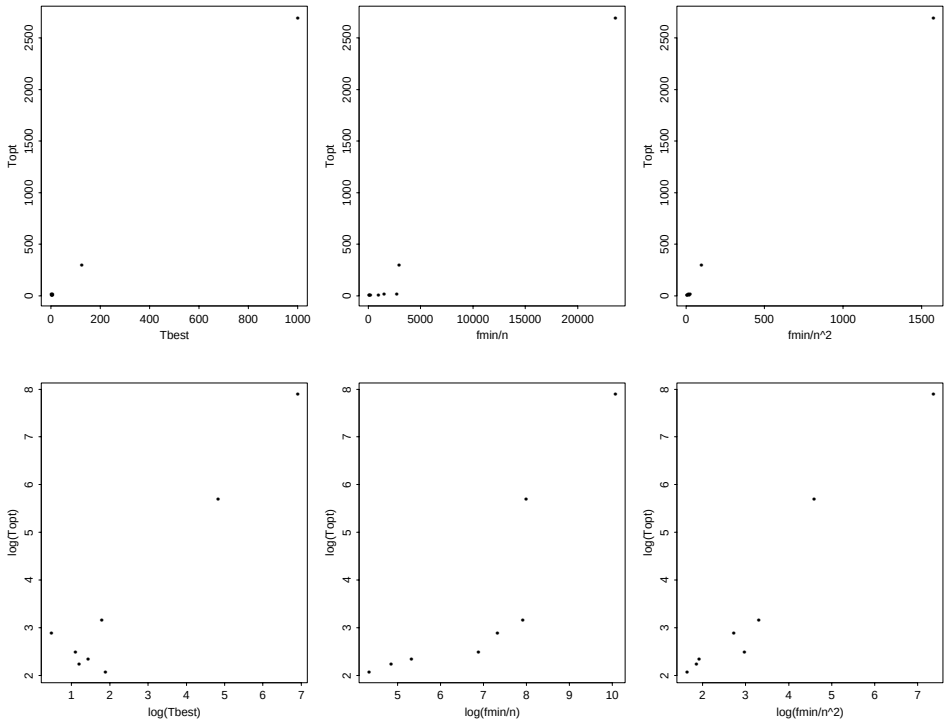

FIG. 6.3. *Relationship between T_{opt} and T_{best}, as well as between T_{opt} and the optimum value of the objective function, for the QAP.*

TABLE 6.5
Predictions of the optimal fixed temperature for various instances of the QAP.

Instance	T_{opt}	by (6.2)	by T_{best}
`nug15`	8.0	7.7	18.5
`rou15`	2700	2361	2798
`nug20`	9.5	9.6	9.2
`nug30`	10.5	10.2	11.8
`kra30a`	300	148	353
`wil50`	12	29	8.4
`wil100`	24	41	17
`sko100a`	18	23	4.5

We therefore examine, in analogy with the case for the TSP, T_{opt} relating to

$$\frac{f_{\min}}{n^2}.$$

Figure 6.3 considers graphically the relationship between T_{opt} and $f_{\min}/n^2$, as well as those between T_{opt} and $f_{\min}/n$ and between T_{opt} and T_{best}. There appears a relationship between T_{opt} and $f_{\min}/n^2$, although it is not as strong as the analogous relationship seen with the TSP. Only a weak relationship is seen between T_{opt} and T_{best}.

Looking at the average ratio between T_{opt} and $f_{\min}/n^2$, we get

$$\hat{T}_{\text{opt}} = 1.5 \times \frac{f_{\min}}{n^2}. \tag{6.2}$$

Predictions may be gained by the average ratio between T_{opt} and T_{best},

$$\hat{T}_{\text{opt}} = 2.8\, T_{\text{best}},$$

found after excluding (outlying) `sko100a`.

6.4. Performance of T_{opt} predictions in the QAP. Table 6.5 shows predictions for T_{opt} that would result, with various methods of prediction.

Although a relationship is apparent between T_{opt} and $f_{\min}/n^2$, this does not lead to a reliable method of predicting T_{opt}. Predictions that would result give a χ^2 value of 222.9. Between $\log(T_{\text{opt}})$ and $\log(f_{\min}/n^2)$, we have the ratios

$$1.275,\ 1.073,\ 1.210,\ 1.226,\ 1.242,\ 0.836,\ 0.961,\ 1.062,$$

with an average of 1.1. By this log-relationship we would get the predictions

$$6.0,\ 3287,\ 7.7,\ 8.2,\ 156,\ 26,\ 38,\ 20,$$

with a χ^2 value of 252.4. Neither method gives a satisfactory explanation for the variation in T_{opt}, with a 5% critical value for a χ^2_7 distribution of 14.1.

The relationship given with T_{best} yields a χ^2 value of 62.4, which is also deemed not satisfactory.

With the QAP from Table 6.6, fixed-temperature simulated annealing appears to outperform the fast cooling schedule for instances of size less than 50 facilities to be allocated.

TABLE 6.6

Performance of the optimal fixed temperature for various instances of the QAP, against Aarts' algorithm. Percentage above global minimum of best solution on average in 100 *runs of the algorithms.*

	% above global	
Instance	Aarts	with T_{opt}
nug15	1.30	0.38
rou15	3.41	1.81
nug20	1.48	0.45
nug30	1.01	0.49
kra30a	2.46	1.94
wil50	0.18	0.27
wil100	0.12	0.28
sko100a	0.22	0.37

TABLE 6.7

Parameters to help predict T_{opt}*, for various instances of the GPP. Values for* T_{opt} *are experimentally determined allowing* N *iterations.*

Instance	T_0	N	T_{opt}	n	degree	$f_{\min}$
rand124	22.0	489499	0.50	124	4.9	(58.8)
rand250	25.5	1440837	0.45	250	5.2	(126.0)
geom250	44.0	1028574	6.0	250	25.8	(248.8)
rand500	28.0	4229417	0.40	500	5.0	(223.0)
rand500b	33.5	4269441	0.50	500	10.2	(695.0)
geom500	43.0	3245665	5.0	500	19.0	(174.8)
geom500b	55.0	3104347	8.0	500	34.8	(578.8)
rand1000	31.0	12064888	0.35	1000	5.2	(483.0)
rand1000b	46.0	12460153	0.70	1000	20.3	(3410.0)
geom1000	36.0	10232663	2.0	1000	9.3	(30.0)

	$\epsilon\%$	χ	f_{best}	T_{best}
rand124	10.0	0.10	63.0	0.23
rand250	6.7	0.08	136.0	0.11
geom250	84.2	0.16	248.8	2.25
rand500	6.0	0.05	250.0	0.17
rand500b	2.4	0.04	712.0	0.17
geom500	214.3	0.16	273.8	1.31
geom500b	82.9	0.13	585.8	1.12
rand1000	5.8	0.04	531.0	0.09
rand1000b	1.3	0.04	3461.2	0.22
geom1000	900.2	0.10	200.0	0.36

6.5. The graph partitioning problem. Table 6.7 shows values for T_{opt} and values for various parameters to help in its prediction (as previously described) for the 10 GPP instances considered.

Figure 6.4 considers a possible relationship between T_{opt} and $f_{\min}/n$ and a possible relationship between T_{opt} and T_{best}. For the GPP instances considered, the average ratio between T_{opt} and T_{best} gives

$$\hat{T}_{\text{opt}} = 3.8 \times T_{\text{best}}.$$

The average acceptance ratio for the 10 problem instances is 0.09, though there is a tendency for the geometric graph instances to relate to larger acceptance ratios, at T_{opt}, than the random instances. The acceptance ratio therefore will not be further considered toward predicting T_{opt} with the GPP.

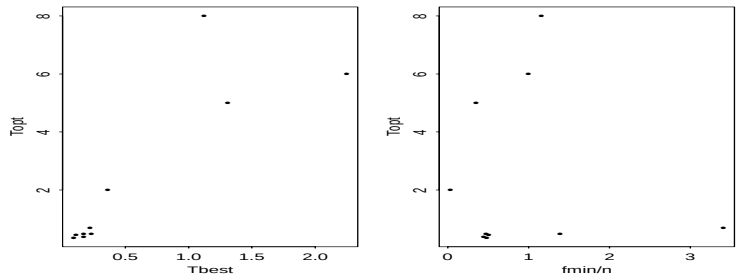

FIG. 6.4. *Investigating possible relationships toward helping predict T_{opt} with the GPP.*

TABLE 6.8
Predictions of the optimal fixed temperature for various instances of the GPP.

Instance	T_{opt}	by T_{best}
rand124	0.50	0.87
rand250	0.45	0.42
geom250	6.0	8.6
rand500	0.40	0.65
rand500b	0.50	0.65
geom500	5.0	5.0
geom500b	8.0	4.3
rand1000	0.35	0.34
rand1000b	0.70	0.84
geom1000	2.0	1.4

TABLE 6.9
Performance of the optimal fixed temperature for various instances of the GPP, against Aarts' algorithm. Best solution on average for 100 *runs of each algorithm, given as the percentage above the best found solution.*

	% above best	
Instance	Aarts	with T_{opt}
rand124	0.76	0.14
rand250	0.40	0.02
geom250	0.55	0.89
rand500	0.97	1.25
rand500b	0.53	0.50
geom500	1.11	7.88
geom500b	0.82	1.94
rand1000	1.58	2.05
rand1000b	0.16	0.56
geom1000	132.55	237.43

The average percentage above global minima, at T_{opt}, varies greatly in the GPP instances considered, with the geometric instances yielding far larger values than the random graph instances, so that $\epsilon\%$ is also not further considered in predicting T_{opt}.

The variation in T_{opt} for the GPP instances may largely be explained by classifying the instances as geometric and random graphs, and the optimal fixed temperature with GPP instances shows to depend largely on problem structure.

6.6. Performance of T_{opt} predictions in the GPP. For predicting T_{opt} by T_{best}, predictions that would result, for the instances considered, are given in Table 6.8. The χ^2 goodness-of-fit value obtained is 62.4. With a critical χ^2_9 value of 19.9,

using T_{best} does not give a satisfactory explanation for the observed variation in T_{opt}.

Consider comparing the performance of fixed-temperature simulated annealing against Aarts' algorithm for the GPP instances considered. Again (see Table 6.9), we see a tendency for fixed-temperature simulated annealing to outperform the fast cooling schedule for smaller sized problems, while its relative performance deteriorates as the size of problems increases.

For the GPP, the performance of fixed-temperature simulated annealing shows to outperform fast cooling for problem instances with less than 500 vertices.

7. Concluding remarks. In using a fixed-temperature simulated annealing algorithm, we have investigated the importance of determining a suitable fixed temperature. We have seen that the temperature deemed most suitable depends on criteria used. Optimal fixed temperatures have been determined experimentally, and investigated also is the predictability of such a temperature.

A formula for fixed temperature prediction has been developed for the TSP. Such did not show to follow directly with other applications considered, although insight gained has also been seen to follow in the QAP, while not yielding satisfactory results.

We have seen that fixed-temperature simulated annealing tends to outperform the use of a fast cooling schedule, for TSP instances of size less than 150 cities, for QAP instances of size less than 50 facilities, and for GPP instances of size less than 500 vertices.

REFERENCES

[1] E. H. L. Aarts and P. J. M. van Laarhoven, *Statistical cooling: A general approach to combinatorial optimization problems*, Philips J. Res., 40 (1985), pp. 193–226.
[2] E. H. L. Aarts and P. J. M. van Laarhoven, *Simulated Annealing: Theory and Applications*, Reidel, Dordrecht, the Netherlands, 1987.
[3] E. H. L. Aarts, J. H. M. Korst, and P. J. M. van Laarhoven, *A quantitative analysis of the simulated annealing algorithm: A case study for the traveling salesman problem*, J. Statist. Phys., 50 (1988), pp. 187–206.
[4] R. E. Burkard, S. E. Karisch, and F. Rendl, *QAPLIB—A Quadratic Assignment Problem Library*, available online from http://fmatbhp1.tu-graz.ac.at/~karisch/qaplib/
[5] V. Černý, *Thermodynamical approach to travelling salesman problem: An efficient simulation algorithm*, J. Optim. Theory Appl., 45 (1985), pp. 41–51.
[6] H. Cohn and M. Fielding, *Simulated annealing: Searching for an optimal temperature schedule*, SIAM J. Optim., 9 (1999), pp. 779–802.
[7] D. T. Connolly, *An improved annealing scheme for the QAP*, European J. Oper. Res., 46 (1990), pp. 93–100.
[8] M. Grötschel, *Polyedrische Charakterisierungen Kombinatorischer Optimie-rungsprobleme*, Hain, Meisenheim am Glan, 1977.
[9] M. Grötschel and O. Holland, *Solution of large-scale symmetric traveling salesman problems*, Math. Programming, 51 (1991), pp. 141–202.
[10] D. S. Johnson, C. R. Aragon, L. A. McGeoch, and C. Schevon, *Optimization by simulated annealing: An experimental evaluation; Part* 1, *Graph Partitioning*, Oper. Res., 37 (1989), pp. 865–892.
[11] R. L. Karg and G. L. Thompson, *A heuristic approach to solving traveling salesman problems*, J. Management Sci., 10 (1964), pp. 225–248.
[12] S. Kirkpatrick, C. D. Gellat, and M. P. Vecchi, *Optimization by simulated annealing*, Science, 220 (1983), pp. 671–680.
[13] S. Kirkpatrick, *Optimization by simulated annealing: Quantitative studies*, J. Statist. Phys., 34 (1983), pp. 975–986.
[14] P. D. Krolak, W. Felts, and G. Marble, *A man-machine approach toward solving the traveling salesman problem*, Commun. ACM, 14 (1971), pp. 327–334.
[15] S. Lin and B. W. Kernighan, *An effective heuristic algorithm for the traveling salesman problem*, J. Oper. Res., 21 (1973), pp. 498–516.

[16] C. E. Nugent, R. F. Vollman, and J. Ruml, *An experimental comparison of techniques for the assignment of facilities to locations*, Oper. Res., 16 (1968), pp. 150–173.
[17] M. Padberg and G. Rinaldi, *A branch-and-cut algorithm for the resolution of large-scale symmetric traveling salesman problems*, SIAM Rev., 33 (1991), pp. 60–100.
[18] G. Reinelt, *TSPLIB—A traveling salesman problem library*, ORSA J. Comp., 3 (1991), pp. 376–384. Also available from online http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95/TSPLIB.html
[19] G. Reinelt, *TSPLIB*, available online from http://www.iwr.uni-heidelberg.de/iwr/comopt/soft/TSPLIB95/TSPLIB.html
[20] J. Skorin-Kapov, *Tabu search applied to the quadratic assignment problem*, ORSA J. Comp., 2 (1990), pp. 33–45.
[21] M. R. Wilhelm and T. L. Ward, *Solving quadratic assignment problems by simulated annealing*, IIE Trans., 19 (1987), pp. 107–119.