PERGAMON

Computers & Operations Research 29 (2002) 67–86

computers & operations research

www.elsevier.com/locate/dsw

Defining tabu list size and aspiration criterion within tabu search methods☆

S. Salhi*

Management Mathematics Group, School of Mathematics and Statistics, University of Birmingham, UK

Received 1 November 1998; received in revised form 1 January 2000

Abstract

An investigation to explicitly define two key elements in tabu search methods is attempted. In this study a functional representation of the tabu list size is presented and a softer aspiration criterion is put forward. Experiments are conducted on a set of p-median problems.

Scope and purpose

Tabu search is a metaheuristic that proved successful in finding good solutions to difficult combinatorial problems that were hard to find otherwise. In this study, we attempt to help the user in the choice of some of the parameters used in this type of heuristics. We based our analysis on the tabu list size and on an implementation on how to define the aspiration criterion. This added information can be valuable to those users who apply these methods in a near systematic manner without relying heavily on experimentations. As an example we used a simple location problem to test the usefulness of these ideas. © 2001 Elsevier Science Ltd. All rights reserved.

Keywords: Tabu search; Tabu duration; Aspiration level; Facility location; p-median problem

1. Introduction

Many problems can be solved optimally by one of the well-known optimization techniques which are widely discussed in most OR/MS textbooks. However, when the problems are combinatorial in nature such as facility location, vehicle routing, job shop scheduling, etc., heuristics appear to be the best way forward in tackling this sort of problems which are known as NP hard. Basic

☆Earlier versions of this paper were presented at IFORS, Vancouver, 1996 and INFORMS, San Diego, 1997.

* Tel.: + 44-121-414-6602; fax: + 44-121-414-3389.

E-mail address: salhis@for.mat.bham.ac.uk (S. Salhi).

PII: S0305-0548(00)00062-9

greedy heuristics, though applicable in practice, did not attract so many academics in the past, as they were considered mainly problem specific and the heuristics were not equipped with mechanisms that attempted to avoid local optimality. In the last 15 years, a new philosophy for enhancing such methods was launched and the idea was to accept nonimproving solutions as well as improving ones. The techniques that take advantage of this flexibility include better constructive/destructive heuristics, simulated annealing, tabu search, genetic algorithms among others. For a general discussion and an overview on these topics, see Reeves [1], and Salhi [2].

In this study, we concentrate on one of these techniques namely tabu search (*TS*) and we investigate some elements within this approach. TS concepts were initially proposed in 1986 by Glover [3], and since then these *TS*-based methods have proved to be successful in several areas of optimization, see Glover et al. [4]. For further discussion see Dammeyer and Voss [5], Glover [6,7], Osman and Salhi [8], Battiti and Tecchiolli [9], Thomas and Salhi [10], and Voss [11,12]. For further references till 1996, see Osman and Laporte [13] and for more recent ones see Glover and Laguna [14].

The main ideas of *TS* methods look easy to understand and to implement. Unfortunately such a simplistic view can be misleading. The success of these *TS* methods depends on several control parameters: These include the size of the tabu list, how a tabu status can be bypassed, the definition of the neighborhood, how diversification schemes are developed and employed, the way previous solutions are identified, the efficiency of the computer program, and above all a good understanding of the problem. In this study, we concentrate on two key factors of the *T*S method namely (i) the tabu list size of the attribute producing the selected solution and (ii) the way the aspiration criterion is defined.

The tabu list size (*LS*) is an important tool in guiding the search in the short term, given the determination of an effective set of attributes for defining tabu status. There are usually at least three approaches for defining *LS*: fixed to a predetermined value, randomly chosen from a specific range, or dynamically changing by adjusting the value. In this study, we examine closely the third approach which we believe is more challenging and more informative. One commonly used idea is to increase or decrease the value of *LS* if the solution has been improved or not during a certain number of iterations. Here, we extend this idea by explicitly defining the tabu list size for each attribute involved at each iteration. In other terms, we are trying to link the contribution a particular attribute may have on the solution to its corresponding tabu list size. The principle of attaching the same *LS* value to all attributes is, in our view, not convincing although it has in many applications produced good results. The relevance of distinguishing between different kinds of attributes and treating them differently has been highlighted by Glover [6] and re-emphasized in the recent book by Glover and Laguna [14]. Unfortunately these important messages have simply been either ignored or overlooked by most TS papers. There are only a few studies that considered these useful concepts. For instance, the use of frequency-based memory gives a different emphasis to different attributes by considering the quality of the solutions. This has the effect of changing the tenure of these attributes. TS memory for ejection chain approaches also typically varies the tenure according to the quality of attributes. However, there are several kinds of dynamic tabu list strategies (cancellation sequence, reverse elimination method, moving gap, etc.) which do not fall into the above category, see Glover [6], Voss [12], and Glover and Laguna [14].

The tabu status of an attribute is usually overriden if the new solution is better than the best solution found. Although such an aspiration level is useful and easy to implement, it can be too restrictive in the search. One of the drawbacks is that all those attributes that are tabu are considered on the same footing. Some ideas on how to relax such a scheme were mentioned by Glover [6] and recently stressed by Glover and Laguna [14]. However, no significant effort seems to be devoted to this challenging modelling issue. For instance, the idea of introducing penalties for tabu status, rather than strict pre-emption, where penalties decline with age, is a good one but does not seem to have been widely exploited by researchers. In this study, we shall propose one possible implementation of this concept which, to our knowledge, has not been specifically addressed in the past.

The purpose of the study is two-fold:

- to provide a logical grounding which may help in defining explicitly the tabu list size. In this study, we put forward a functional representation which is based on the attribute at that time of the move and the change in the objective function such an attribute generates.
- to develop a softer aspiration criterion which considers simultaneously the tabu status of an attribute, how far a nonimproving solution is from the best solution found so far (in terms of cost), and the current change in cost such an attribute generates.

We have chosen the p-median problem as a platform to test our ideas. In Section 2 we present our methodology. In Section 3 we propose a functional representation of the tabu list size. The criterion used for representing aspiration level is given in Section 4. Our computational results are summarized in Section 5 and we complete the paper with a conclusion and some research avenues in Section 6.

2. Methodology

In this section we first present a brief outline of the p-median problem, the descent method we adopted and the tabu search heuristic. The determination of the tabu list size and the way the aspiration level is defined are given in the next two sections.

2.1. A brief outline of the p-median problem

The p-median problem is probably the most commonly researched problem in the area of locational analysis. In the p-median problem the aim is to find the optimal location of p facilities among m potential sites in a graph or network which minimizes the weighted distance (or cost) from all the nodes (customer' sites, say n) assuming each node is assigned exclusively to its nearest facility. The values for the fixed cost of establishing a facility is considered the same for all facilities. This is usually set to zero for clarity. In addition, there is no restriction on the capacity of the facilities. In this paper, we assume that all customer' sites are potential sites (i.e., $m = n$).

There exist several algorithms for solving this location problem. The most recognized approaches include Lagrangian relaxation heuristics (see Beasley [15], and Agar and Salhi [16]) and linear-programming-based approaches (see Erlenkotter [17]). Although these methods have proved to be successful, they have the handicap of not being easily extendable to handle efficiently more complex problems such as capacitated and single source location problems especially if computing time is a scarce resource. To overcome such potential shortcomings, heuristic methods are devised, see Voss [11], Rolland, Schilling and Current [18], Rosing and ReVelle [19] and Salhi

[20]. The first two are based on tabu search and the last two use heuristic concentration and perturbation methods, respectively.

2.1.1. Descent method

A descent (or greedy) method is a local search method that proceeds by examining some neighborhoods of the current solution. The process is repeated until there is no further improvement and the final best solution is considered as the solution of the problem. In our example, the neighborhood is represented by a drop move (remove from the current set of open facilities, one facility at a time) and when the number of open facilities reaches p, a swap move is activated (simultaneously open and close one facility). The process terminates when there is no possible cost improvement. The main steps of the descent method we used are briefly given below.

The method:

(a) Apply any suitable location heuristic to generate p facilities.
(b) Use a node exchange procedure (swap move) to improve on the current solution.
(c) If there is an improvement go to (b), otherwise record the final solution as the local optimal solution.

In (a) we applied the modified drop heuristic of Salhi and Atkinson [21]. This heuristic refers mainly to the idea of dropping one facility at a time until there is no further improvement. This concept is due to Feldman et al. [22] except that Salhi and Atkinson used the dropping scheme several times, each one starting from a subset rather than from the entire set. This simple but efficient implementation led to better solutions with a drastic reduction in computational effort when compared to the usual implementation of the drop heuristic. This modified drop heuristic does not have the weakness of the original drop heuristic in which the solution with p facilities is exactly the solution with $p + 1$ facilities with the exception of one. In (b) the commonly used method of Kuehn and Hamburger [23] is adopted except that small changes in the implementation are introduced. The cost of the swap move will be defined in the next section. Note that the drop and add moves are now commonly used in the tabu search literature, see Rolland et al. [18].

In the rest of the paper, one iteration of the *TS* method refers to step 3 and 4. We call a tabu solution a solution which is formed by a swap move where one of the attribute (facility) happens to be tabu.

The TS method

step 1. Use the final solution obtained by the descent method as the starting solution, and set all attributes (facilities) to be nontabu.

step 2. Select the best neighboring solution using the swap move. This can be either
- the nontabu solution having the least cost
- or a tabu solution that passes an aspiration level (this will be described in Section 4).

step 3. Make the reversal move tabu for a certain number of iterations (this will be described in Section 3).

step 4. If the maximum number of iterations is reached, or no improvement has been found after a certain number of successive iterations, stop;
Else go to step 2.

2.1.2. Elements describing the TS heuristic (steps 2 & 3)
Neighborhood structure:

A neighboring facility configuration is obtained by removing one open facility and replacing it with a currently closed facility. This is done formally as follows:

$$E' \in N(E) \text{ if } E' = E - \{i\} \cup \{j\},\ i \in E \quad \text{and} \quad j \in \{M_{i1}, \ldots, M_{iK}\}.$$

where E denotes the current set of open facilities, $|E| = p$ the cardinality of E, and $N(E)$ the neighborhood of E. K the number of nearest facilities which are not in the current solution to any given facility in the solution. M_{ik} the k^{th} nearest closed facility to facility i ($i \in E$ and $M_{ik} \notin E$), $k = 1, \ldots, K$ ($K \leqslant m - p$).

Tabu definition

When a given facility is involved in obtaining the new solution (such a facility is either removed or added to the set of open facilities), say facility i, such a facility is not allowed to be part of any new solutions for a certain number of iterations, say $LS(i)$, except if it passes a certain aspiration criterion. The computation of $LS(i)$ and the definition of the new aspiration criterion will be given in the next two sections.

Approximated cost of the swap move

To speed up the process, we used an approximation for the total cost. It is only when the move is selected do we compute the real cost based on those selected facilities.

Let

C_{ik} :the cost of serving customer k ($k = 1, \ldots, n$) from facility i ($i = 1, \ldots, m$).

G_i :the set of customers served from the open facility $i \in E$, and $G_i = \{k = 1, \ldots, n \text{ st } C_{ik} = \text{Min}(C_{sk}, s \in E)\}$.

$\zeta(i)$:the sum of the allocation costs between all customers in G_i and the open facility $i \in E$. In other words, $\zeta(i) = \sum_{k \in G_i} C_{ik}$

$\tilde{\zeta}(j_i)$:the sum of the allocation costs between all customers in G_i and the temporarily open facility $j \notin E, i \in E$. In other words, $\tilde{\zeta}(j) = \sum_{k \in G_i} C_{jk}$

$\hat{\Delta}_{ij}(\Delta_{ij})$:approximate (real) change in total cost by removing facility i and adding facility j ($i \in E$; $j \notin E$).

Computation of the approximated cost $\hat{\Delta}_{ij}$:

For each $i \in E$ ($j \notin E$)
compute $\hat{\Delta}_{ij} = \zeta(i) - \tilde{\zeta}(j), \forall j \in \{M_{i1}, \ldots, M_{iK}\}$

Selection of the best move

- Find the admissible swap moves (i, j).
  A move is said admissible if either (a) facility i and facility j are both nontabu or (b) the associated approximate cost is less than the least cost so far (e.g., $\Phi(E) - \hat{\Delta}_{ij} < \Phi(S_{\text{best}})$) where S_{best} denotes the facility configuration (the solution) that produced the current least cost so far, and $\Phi(E)$ is the total cost incurred using the set of open facilities E (e.g., $\Phi(E) = \sum_{i \in E} \zeta(i)$). If the move (i, j) is found admissible then we store $\hat{\Delta}_{ij}$.

  A different aspiration criterion will be discussed in Section 4.

- Compute
  $\hat{\Delta}_{rs} = \text{Max}(\hat{\Delta}_{ij};\ i \in E;\ j \in \{M_{i1}, \dots, M_{iK}\}$ and the move (i,j) is admissible)

Computation of the true cost Δ_{rs}
Let N_{ik} denote the ith nearest facility ($N_{ik} \in E$) to serve customer k, $i = 1, \dots, p$ and $k = 1, \dots, n$.

- Set $E' = E - \{r\} \cup \{s\}$ and $\Phi(E') = \Phi(E)$
- For each customer $k = 1, \dots, n$ do
  Set $i_1 = N_{1k}$ and $i_2 = N_{2k}$.
  If ($i_1 = r$) then
    if $C_{i_2 k} < C_{sk}$ then $\Phi(E') = \Phi(E') - C_{i_1 k} + C_{i_2 k}$ and $N_{1k} = N_{2k}$
    else $\Phi(E') = \Phi(E') - C_{i_1 k} + C_{sk}$ and $N_{1k} = s$
  Else
    if $C_{i_1 k} < C_{sk}$ then keep $\Phi(E')$ unchanged,
    else $\Phi(E') = \Phi(E') - C_{i_1 k} + C_{sk}$
  End if
  Update $N_{ik}, i = 2, \dots, p$.
  End do
- Set $\Delta_{rs} = \Phi(E') - \Phi(E)$

2.1.3. Some comments

We considered the simple neighborhood (the swap move) in this work. This can obviously be very restrictive and hence limit the search to finding better solutions. One way is to introduce in combination with the current one, other neighborhoods such as the add and the drop moves as commonly used in the literature. The use of these additional neighborhoods will obviously improve the quality of the solutions at the expense of extra computational effort.

Also, a diversification scheme can be used to guide the search in exploring other regions which might not be reached otherwise. Several useful diversification strategies can be found in the recent book by Glover and Laguna [14]. Such schemes are widely used in TS based methods.

In this study, we did not introduce such enhancements as these may mask the effect of what we aim to illustrate. Obviously in practice, these schemes need to be added as these have proved to be successful in several applications. It is therefore worth noting that the obtained solutions in this work can be easily improved, but this is not the subject of this investigation.

3. A functional representation of the tabu list size

We relate the tabu list size for the ith attribute, say $LS(i)$, to the change in cost of the objective function it produces. For instance in the p-median problem, swapping the open facility i with the closed facility j results in a change in cost Δ_{ij}. Developing a mapping between Δ_{ij} and $LS(i)$ or $LS(j)$ is not straightforward and requires a careful investigation. One way to construct this mapping is to base it on the following observation

More emphasis may need to be put on the attribute which has just left the solution by penalizing it according to the change in the cost function it produced. In other words, a higher value for the tabu size *is assigned to an attribute for returning to the solution if its removal yielded a larger improvement than one which produced a smaller or negative improvement. Besides, the attribute leaving the solution needs to have a higher value than the one entering the solution as these two attributes need to be treated differently.*

Such an observation is mostly based on the p-median problem and may not be valid in other problems though in many circumstances this may be the case. The functional representation which we put forward here is an attempt towards explicitly representing $LS(i)$. Note that our mapping is not unique and any suitable one is worth the investigation. In addition, we restricted the domain of $LS(i)$ to be between $LS_{\min}$ and $LS_{\max}$. This is used to guarantee that the values of $LS(i)$ remain within a certain range.

$$g: R \to R^{+} \cap [LS_{\min}, LS_{\max}]$$

$$\Delta_{ij} \to LS(i) = g(\Delta_{ij}, \Omega),$$

where $LS_{\min}$ and $LS_{\max}$ are prespecified but can be dependent on the neighborhood and the problem size. Ω is the set of other contributing factors that may be useful to incorporate. These may include the neighborhood size of attribute i, its frequency of occurrence in the previous solutions, the current number of iterations, among others. $g(\Delta_{ij}, \Omega)$ is a continuous and positive nondecreasing function of Δ_{ij}. The continuity of the function is used for simplicity of construction. In this study we did not include Ω. Similarly we can construct a mapping $h(\Delta_{ij})$ linking Δ_{ij} to $LS(j)$ for the attribute, say facility j, entering the solution.

A typical plot of $LS(i)$ vs Δ_{ij} is sketched in Fig. 1. The values of $LS(i)$ are taken as the integer part of $LS(i)$.

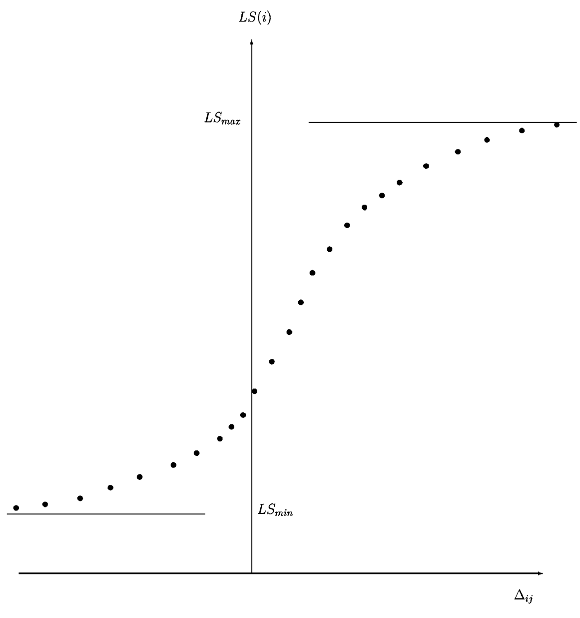

Fig. 1. A possible pattern of $LS(i)$ vs. Δ_{ij} (case of dropping facility i).

For instance, in the case of the p-median problem where simultaneously facility i is removed and replaced by facility j, the following mapping is used.

(a) facility i leaving the solution:

$$LS(i) = g(\Delta_{ij}) = \begin{cases} \mathrm{Min}(LS_{\max}, LS_{\min} + e + \mathrm{Log}(\Delta_{ij} + 1)) & \text{if } \Delta_{ij} \geqslant 0, \\ LS_{\min} + e^{(1+\Delta_{ij})} & \text{else,} \end{cases}$$

(b) facility j getting into the solution:

$$LS(j) = h(\Delta_{ij}) = \begin{cases} (1+\lambda)T_0 & \text{if } \Delta_{ij} \geqslant 0, \\ T_0 & \text{else,} \end{cases}$$

where $\lambda \in [0, 1]$, and T_0 is set to $LS_{\min}$ in this study.

Note that in case (b) since the range can be relatively small (bounded by $p - 1$) it was felt that it is unnecessary to construct a complex mapping for $LS(j)$.

4. A softer aspiration criterion

The level for which aspiration overrides tabu status is a sensitive key factor in the search as this defines the degree of flexibility of the method. One common-sense-based approach is to relax the tabu restriction if a solution happens to produce a better result than the currently best-known solution. This is obviously correct as the search would look really blind if it was too rigid by not taking such a better solution. Consider the likely found scenario where all the top solutions are both tabu and nonimproving. The question is whether it is appropriate to choose the first nontabu solution or to select one of those top tabu solutions. In case all the moves happen to be tabu, one usual implementation is to free the attribute with the least tabu tenure.

The elements which will constitute the decision need to include at least the following:

- the tabu status of the attribute for that solution,
- how much does a solution differ from the best one in terms of cost,
- and the change in the objective function.

A flexible concept of aspiration level which takes into account those attributes with active tabu status is, in our view, a good way forward. In this study, we propose a selection rule, which combines both the tabu tenure of a given attribute and the change in the objective function value it produces. In other terms, the tabu restriction of an attribute which is nearly nontabu and which has produced a solution nearly as good as the best solution (but not good enough to permit overriding its current tabu status), should, in our view, be treated differently and hence may be worth relaxing. Such a scheme can be written formally as follows:

Consider that, at a given iteration, we have M solutions. These can be put in a list of ascending order of $\tilde{\Delta}$ where $\tilde{\Delta}_j = Cost(S_j) - Cost(S_{\text{best}}), j = 1, \ldots, M$; and S_j is the jth possible solution.

We define the tabu tenure of a solution j as $t_j = \max\{0, LS(j_1)\} + \max\{0, LS(j_2)\}/2$, where j_1 and j_2 are the two facilities involved in the swap move.

In other words, there are t_j iterations left for solution j to become nontabu. Note that this could be defined differently by putting more emphasis on the leaving or the entering variable.

$$\begin{pmatrix} S_1 & \tilde{\Delta}_1 & t_1 > 0 \\ S_2 & \tilde{\Delta}_2 & t_2 > 0 \\ - & - & - \\ S_{q-1} & \tilde{\Delta}_{q-1} & t_{q-1} > 0 \\ S_q & \tilde{\Delta}_q & t_q = 0 \\ - & - & 0 \\ S_M & \tilde{\Delta}_M & t_M = 0 \end{pmatrix}$$

Assume that $S_1, \ldots, S_{q-1}$ are tabu (i.e; $t_j > 0, j = 1, \ldots, q-1$) and S_q is the first nontabu solution (i.e; $t_q = 0$).

In general, we have the following scheme:

- If $\tilde{\Delta}_1 < 0 \Rightarrow S_1$ is an improving solution and S_1 will be chosen according to the usual aspiration criterion (using the objective function value) to override the tabu status.
- If $\tilde{\Delta}_1 \geqslant 0 \Rightarrow$ all moves are nonimproving and therefore the choice is either

(i) to take the first move in the list that is not tabu, e.g., S_q.
(ii) or to use a criterion for choosing an S_v from $\{S_1, \ldots, S_q\}$.
Note that S_q is also included in the choice as given by (ii). In (i) the usual way of using the tabu restriction rule is strictly employed whereas in (ii) more flexibility is introduced into the decision. The latter, though it is mentioned in the recent literature (see Glover and Laguna [14]) does not seem to have been explored by other researchers. This concept, in our view, is both promising and challenging and therefore attempts to model such an issue are encouraged.

4.1. *A softer aspiration level*

This is based on the concept of criticality of the tabu status. For instance, an attribute which has a tabu tenure of 10 can be considered more 'tabu' than the one associated with a value of 1 or 2. In addition, a solution may be closer to the best current solution while it is still nonimproving. A flexible aspiration criterion which combines these characteristics can be employed in handling such a problem. In this study, we put forward the following procedure.

- Form a set of promising tabu moves, say

  $$PTM = \{S_j; j = 1, \ldots, q-1 \text{ st: } \tilde{\Delta}_j \leqslant \beta\tilde{\Delta}_q \ \& t_j \leqslant LS_{\min}\}$$
  where $\beta \in [0, 1]$

- If $PTM = \emptyset$ choose the solution S_q.

- Else
  (i) For all $S_j \in PTM$ compute $\delta_j = (\tilde{\Delta}_q - \tilde{\Delta}_j)/t_j^c$ where c is a parameter in [0, 1] describing the importance of penalizing a move with a large tabu tenure.
  (ii) Find $\delta_{j^*} = \mathrm{Max}(\delta_j, S_j \in PTM)$ and choose the solution S_{j^*}

In the circumstances where we have $q = M$ (i.e., all these nonimproving solutions happen to be tabu), the tabu tenures are adjusted using the following basic transformation

$t_j = t_j - t_k$, $j = 1, \ldots, M$ where t_k denotes the tabu tenure of the solution with the least tabu tenure. The above rule can then be applied using the adjusted t_j values.

5. Computational results

The proposed heuristics are implemented in Fortran 90 on a Sun Sparc 20 workstation (712MP dual processor, 70/75 MHz with integer SPECrate 5726 and floating point SPECrate 5439). We tested our approaches on two scenarios. In scenario 1, five data sets ranging in size from 200 to 796 customers are used. These are constructed from the problem sets given by Christofides et al. [24] and which are originated for the vehicle routing problem. For our purpose, we have transformed these sets into problems which are four times larger. In scenario 2, we considered the four data sets used in the continuous location problems. These have 50, 287, 654 and 1060 customers, see Brimberg et al. [25]. In each data set we run the heuristics for $p = 4$ to $\max(10\%n, 30)$ with a step size of 2, totalling 114 instances in scenario 1 and 111 instances in scenario 2.

Our analysis is based on the descent method, and the three variants of the tabu search approach namely (i) the one using $LS(j)$ values randomly generated within a specified range, (ii) the functional representation of $LS(j)$ and (iii) the new aspiration criterion with and without (ii).

The solution quality (cost) and the computing time required to obtain these solutions (in secs) are reported in Appendices A and B. A sign test for median is conducted to show the number of solutions which are better or worse than those found by the random TS. The summary results for scenarios 1 and 2 are given in Tables 1 and 2, respectively. A Wilcoxon signed rank test was also performed and showed that there is not a significant difference between the methods at a significant level $\alpha = 5\%$. Such a result is usually common in heuristic search where one particular approach does not systematically outperform all the others with a statistically significant margin, though one approach may dominate others.

In these experiments the following values of the parameters are used:

The maximum number of iterations and the number of iterations between successive nonimproving moves are set to $10n$ and n, respectively.

The number of diversifications is set to 5 and $q = [p/3]$.

$m = n$ and $K = \mathrm{Min}(K_{\max}, \alpha m/p)$ with $K_{\max}$ and α set to 50 and 1.50, respectively.

$LS_{\min} = \mathrm{Max}(3, p/3)$ and $LS_{\max} = \mathrm{Max}(p, (n-p)/10)$,

$\lambda = 0.20$, $\beta = 0.8$ and $c = 0.50$.

We used the same range $[LS_{\min}, LS_{\max}]$ for all the three TS methods. In the first variant, the tabu list size for the facilities entering the solution is set to $LS_{\min}$.

Table 1
Sign test results: Scenario 1

N (# Instances)	# Best solutions	Random TS (1)	Fractional TS (2)	Soft aspiration with (1)	Soft aspiration with (2)
200	Worse than random TS (incl ties)		4(7)	3(8)	6(11)
(14)	Better than random TS (incl ties)		7(10)	6(11)	3(8)
	Overall best (incl ties)	1(4)	4(9)	3(9)	0(5)
300	Worse than random TS (incl ties)		4(10)	6(11)	7(12)
(14)	Better than random TS (incl ties)		4(10)	3(8)	2(7)
	Overall best (incl ties)	4(10)	2(9)	1(7)	0(6)
400	Worse than random TS (incl ties)		8(11)	7(17)	5(15)
(19)	Better than random TS (incl ties)		8(11)	2(12)	4(14)
	Overall best (incl ties)	1(9)	6(10)	2(7)	2(7)
600	Worse than random TS (incl ties)		9(19)	5(17)	5(18)
(29)	Better than random TS (incl ties)		10(20)	12(24)	11(24)
	Overall best (incl ties)	3(14)	7(16)	10(19)	8(19)
796	Worse than random TS (incl ties)		15(18)	24(27)	26(28)
(38)	Better than random TS (incl ties)		20(23)	11(14)	10(12)
	Overall best (incl ties)	10(13)	15(18)	7(8)	2(2)
All	Worse than random TS (incl ties)		40(65)	44(79)	51(84)
(114)	Better than random TS (incl ties)		49(74)	35(70)	30(63)
	Overall best (incl ties)	19(50)	34(62)	23(50)	12(38)

The tabu-search-based heuristics improved on the descent method as expected, see Appendices 1 and 2. In scenario 1, the TS variant using the well-structured definition of $LS(i)$ (functional representation) appears to dominate the one using the less informative definition of $LS(i)$ (random picks) although the results were not significantly different. For instance, over 54% of the best solutions were found by the functional TS compared to just 44% obtained by both the random TS, and the TS method using the soft aspiration criterion. In addition, the fractional TS generated 49 better solutions, excluding ties, than the random TS while the latter outperformed the former in 40 instances, see Table 1. In scenario 2, the TS variants with and without the soft aspiration are the most dominant as the best results are mainly generated from these two variants, see Table 2 for details. Unfortunately in this scenario the functional representation may have been too restrictive given the magnitudes of the cost values. We believe that some enhancements may be needed to reflect better the original purpose of introducing this representation. In summary, we can conclude that though there was not always an improvement when using these modifications, better results

Table 2
Sign test results: Scenario 2

N (# Instances)	# Best solutions	Random TS (1)	Fractional TS (2)	Soft aspiration with (1)	Soft aspiration with (2)
50	Worse than random TS (incl ties)		0(14)	0(14)	0(14)
(14)	Better than random TS (incl ties)		0(14)	0(14)	0(14)
	Overall best (incl ties)	0(14)	0(14)	0(14)	0(14)
287	Worse than random TS (incl ties)		6(13)	4(12)	11(14)
(14)	Better than random TS (incl ties)		1(8)	2(10)	0(3)
	Overall best (incl ties)	4(12)	0(8)	1(10)	0(3)
654	Worse than random TS (incl ties)		21(30)	13(27)	22(31)
(31)	Better than random TS (incl ties)		1(10)	4(18)	0(9)
	Overall best (incl ties)	11(26)	1(10)	3(18)	0(9)
1060	Worse than random TS (incl ties)		19(49)	15(45)	24(51)
(52)	Better than random TS (incl ties)		3(33)	8(36)	1(28)
	Overall best (incl ties)	13(42)	1(29)	7(35)	0(28)
All	Worse than random TS (incl ties)		46(106)	32(98)	57(110)
(111)	Better than random TS (incl ties)		5(65)	14(78)	1(54)
	Overall best (incl ties)	28(94)	2(61)	11(77)	0(54)

were found for some instances, and hence one way is to test these variants in combination while more work is pursued to better reflect these two key factors we raised in this paper.

6. Conclusion and possible research directions

In this study, we have tried to have an insight into tabu search methods. The benefit of using a functional representation of the tabu list size proved to be challenging. This additional information is useful to the user as he or she may like to construct a suitable functional representation for the tabu list size to approach his or her particular application. A flexible concept of aspiration level which accepts some attributes with active tabu status was also shown to be a good way forward. The proposed selection rule is based on both the tabu tenure of a given attribute and how far a trial solution is from the overall best when measured in terms of cost.

The results obtained by both approaches are encouraging though the difference in the results is not statistically significant. The fractional TS approach was slightly superior to the other TS variants when tested on data sets from the first scenario, as it produces over 20% more best solutions than the random TS. Unfortunately a similar outcome was not obtained in the second scenario. The implementation of the softer aspiration level unfortunately did not seem to enhance the results obtained by the functional $LS(i)$ in some of the experiments, though some new results were obtained for some instances.

It is worth noting that this is the first investigation into such conceptual aspects and some further insight is necessary in future to better describe these two key factors and hence improve on the effectiveness of the *TS* method in general. It should be stressed that the aim of this study is to raise these methodological issues, which hopefully can be pursued further to render the proposed approaches more powerful. For instance, other input variables such as neighborhood size, frequency of occurrence, scaling, among others could also be embedded into the functional representation for $LS(i)$. Similarly, the aspiration criterion could be extended to include some of this information.

It may be argued that in this approach there are more parameters to use than in the classical one. Though such a remark is true, the parameters used here are of different type as the user has more control on how to measure these parameters. In the simplistic and classical setting these values can be subjective and mainly found by experiments. Such an approach, though it proves successful in testing academic problems, can be of little help to those users who are interested in getting acceptable solutions within a short time period.

These ideas can undoubtedly be adapted for many combinatorial problems where basic definitions of *TS* elements have already proved to be successful. We believe that the next development stage in heuristic search design is toward a more informative and explanatory modelling aspect of all those critical parameters existing in heuristic search in general and in tabu search in particular. It may not be a unique answer to this problem but at least the window will be open to both those practitioners and academics who do not believe in *trial search* but who appreciate very much the philosophy as well as the success of *heuristic search*.

Acknowledgements

The author is grateful to both referees for their constructive comments which improved both the content as well as the presentation of the paper. The author would also like to thank Professors S Voss and S Hanafi for their useful comments on an earlier version of this paper.

Appendix A

Total cost and time for scenario 1 is given in Table 3.

Appendix B

Total Cost and time for scenario 2 is given Table 4.

Table 3
Total cost and time (s) for scenario 1 (no diversification)

N	*p*	Descent method		Random *TS* (1)		Fractional *TS* (2)		Soft aspiration with (1)		Soft aspiration with (2)	
		Cost	Time	Cost	Time	Cost	Time	Cost	Time	Cost	Time
200	4	4723	0.15	4682	1.15	4668	1.34	4682	1.15	4682	1.18
	6	4017	0.15	3947	1.77	3872	2.37	4010	1.18	4009	1.20
	8	3318	0.14	3305	1.10	3305	2.02	3305	1.08	3305	1.08
	10	2986	0.14	2963	1.49	2942	1.53	2962	1.04	2986	1.03
	12	2775	0.14	2732	1.63	2767	1.00	2684	2.09	2775	0.98
	14	2554	0.14	2503	1.47	2516	1.49	2496	1.52	2554	0.97
	16	2399	0.14	2313	2.43	2363	1.80	2365	2.25	2363	2.21
	18	2197	0.13	2197	0.96	2166	2.27	2197	0.99	2197	0.95
	20	2130	0.15	2082	1.00	2057	1.71	2109	1.42	2115	1.84
	22	1993	0.21	1987	1.90	1988	1.48	1978	1.48	1981	1.07
	24	1870	0.29	1853	1.16	1849	1.56	1844	1.58	1844	2.01
	26	1782	0.29	1774	1.14	1765	1.13	1765	1.12	1765	1.16
	28	1665	0.29	1657	1.16	1657	1.17	1657	1.16	1657	1.17
	30	1592	0.29	1584	1.17	1584	1.17	1584	1.18	1584	1.18
	AVER	2571	0.19	2541	1.39	2535	1.57	2545	1.37	2558	1.29
	# Best	0		4		9		9		5	
300	4	7202	0.44	7202	5.49	7202	5.52	7202	5.48	7202	5.56
	6	6283	0.42	6187	4.52	6187	4.63	6283	4.42	6187	4.59
	8	5364	0.43	5358	3.15	5358	3.17	5358	3.56	5358	3.19
	10	4886	0.42	4837	6.02	4884	3.13	4867	6.06	4869	5.84
	12	4424	0.42	4391	3.06	4414	4.20	4410	4.27	4410	2.97
	14	4080	0.41	4006	4.04	4023	4.08	4018	5.31	4018	4.13
	16	3802	0.41	3749	4.04	3737	5.20	3741	2.79	3768	2.73
	18	3523	0.41	3493	7.25	3510	6.08	3522	2.76	3516	3.94
	20	3325	0.42	3325	2.76	3325	2.75	3325	2.72	3325	2.72
	22	3185	0.41	3185	3.91	3182	2.71	3182	3.90	3182	2.74
	24	3090	0.40	3087	5.10	3087	2.72	3087	2.74	3090	2.80
	26	2970	0.34	2954	2.68	2950	3.86	2935	3.90	2941	2.74
	28	2831	0.40	2831	2.75	2831	2.74	2831	2.78	2831	2.81
	30	2741	0.44	2732	4.09	2724	5.31	2733	4.17	2736	2.93
	AVER	4121	0.41	4095	4.20	4101	4.01	4106	3.92	4102	3.55
	# Best	3		10		9		7		6	
400	4	9931	0.97	9931	10.91	9926	11.14	9931	11.05	9931	11.03
	6	8422	0.94	8422	10.71	8407	10.84	8422	10.89	8407	10.98
	8	6973	0.96	6965	12.29	6965	8.46	6965	8.61	6965	12.28
	10	6259	0.94	6255	9.33	6259	6.53	6255	9.52	6255	9.47
	12	5643	0.95	5631	9.55	5623	9.50	5639	6.65	5639	6.56

Table 3 (*continued*)

N	*p*	Descent method		Random *TS* (1)		Fractional *TS* (2)		Soft aspiration with (1)		Soft aspiration with (2)	
		Cost	Time	Cost	Time	Cost	Time	Cost	Time	Cost	Time
	14	5157	0.92	5121	8.91	5121	9.55	5132	9.12	5157	6.26
	16	4792	0.93	4715	8.79	4726	6.21	4715	8.95	4723	11.81
	18	4548	0.91	4524	8.68	4515	8.70	4509	11.49	4516	11.41
	20	4369	0.91	4307	8.63	4307	8.69	4307	11.38	4327	8.88
	22	4209	0.91	4114	13.69	4151	13.63	4126	11.39	4134	13.75
	24	4059	0.90	4059	5.97	4054	8.32	4059	6.09	4059	6.10
	26	3896	0.91	3884	5.95	3885	5.90	3884	5.96	3884	5.97
	28	3753	0.92	3751	6.03	3730	11.35	3751	6.03	3751	6.09
	30	3638	0.90	3620	8.73	3625	8.63	3611	8.64	3620	8.46
	32	3552	0.89	3541	8.57	3548	6.07	3546	6.02	3541	8.49
	34	3452	0.75	3398	5.80	3454	5.77	3418	5.78	3398	5.76
	36	3334	0.83	3313	6.04	3298	6.08	3326	5.98	3254	11.17
	38	3225	0.92	3230	6.13	3226	6.11	3234	8.83	3230	6.16
	40	3154	0.99	3155	6.33	3166	6.34	3155	9.09	3150	6.29
	AVER	4861	0.91	4838	8.48	4841	8.31	4841	8.50	4839	8.79
	# Best	0		9		10		7		7	
600	4	14666	3.10	14666	27.38	14666	27.31	14666	27.24	14666	27.61
	6	12463	3.08	12463	28.60	12463	28.63	12463	28.59	12463	29.58
	8	10458	3.07	10458	29.54	10458	29.50	10458	29.52	10458	29.77
	10	9375	3.21	9375	29.60	9362	29.54	9375	29.63	9362	29.61
	12	8596	3.08	8541	50.98	8537	38.83	8505	52.86	8522	27.14
	14	7894	3.04	7841	33.08	7840	32.82	7841	33.99	7840	22.98
	16	7287	3.04	7287	19.80	7287	19.76	7287	19.63	7287	19.60
	18	6841	3.08	6798	28.05	6798	27.94	6798	29.85	6798	28.69
	20	6617	3.01	6536	27.21	6582	28.86	6518	36.02	6529	34.44
	22	6254	3.04	6216	43.45	6253	18.76	6253	19.15	6253	19.17
	24	6007	3.02	5982	26.55	5993	19.22	5982	27.30	5995	18.99
	26	5760	3.00	5760	17.96	5755	18.20	5755	18.21	5755	18.43
	28	5592	2.99	5578	17.96	5575	17.96	5576	17.82	5578	18.04
	30	5393	3.00	5393	17.55	5384	25.85	5377	26.05	5377	25.83
	32	5224	2.99	5171	40.23	5191	25.44	5176	25.56	5176	25.70
	34	5053	2.98	5026	25.23	5022	32.99	5025	25.26	5002	40.61
	36	4914	2.97	4913	17.57	4871	40.82	4869	40.76	4882	40.57
	38	4774	2.98	4774	18.00	4765	18.11	4765	25.70	4774	18.27
	40	4659	2.95	4659	18.32	4659	18.57	4659	18.45	4659	17.88
	42	4545	2.95	4538	18.27	4538	18.42	4533	26.23	4538	18.28
	44	4439	2.99	4439	18.26	4439	18.36	4439	18.47	4439	18.21
	46	4346	2.94	4340	18.63	4312	18.60	4312	18.50	4346	18.51
	48	4246	2.93	4246	18.47	4246	18.83	4246	18.64	4246	18.68
	50	4156	2.93	4156	19.04	4156	19.01	4156	19.22	4156	18.55
	52	4035	2.73	4034	18.42	4034	18.35	4035	18.40	4034	18.43
	54	3943	3.15	3936	27.26	3935	35.29	3931	35.64	3931	43.25

Table 3 (*continued*)

N	*p*	Descent method		Random *TS* (1)		Fractional *TS* (2)		Soft aspiration with (1)		Soft aspiration with (2)	
		Cost	Time	Cost	Time	Cost	Time	Cost	Time	Cost	Time
	56	3831	3.64	3818	27.81	3819	27.52	3829	35.79	3831	19.73
	58	3781	4.17	3772	20.43	3777	20.47	3771	20.30	3752	44.10
	60	3746	4.68	3746	20.93	3747	21.32	3747	20.95	3738	21.07
	AVER	6168	3.13	6153	24.99	6153	24.66	6149	26.34	6151	25.23
	# Best	8		14		16		19		18	
796	4	19390	11.51	19358	91.60	19319	63.99	19319	63.89	19358	92.38
	6	16186	11.53	15833	99.42	15781	126.57	15901	67.28	15890	67.60
	8	13299	11.51	13217	75.73	13224	103.48	13224	72.39	13225	100.54
	10	11729	11.63	11598	104.95	11626	104.30	11641	134.14	11696	103.93
	12	11015	11.53	10661	135.51	10661	131.97	10658	116.31	10692	127.44
	14	10167	11.43	9941	136.24	9948	138.88	10046	135.19	9969	131.20
	16	9465	11.49	9290	157.67	9296	157.21	9290	130.74	9291	159.59
	18	8787	11.48	8688	159.26	8667	175.17	8707	132.80	8678	132.97
	20	8408	11.50	8325	120.87	8338	122.46	8355	119.35	8330	117.76
	22	8004	11.47	7865	142.12	7888	119.32	7878	144.15	7897	142.01
	24	7529	11.55	7447	86.16	7447	87.85	7478	87.13	7462	89.51
	26	7225	11.53	7154	131.02	7146	82.75	7126	131.43	7127	85.23
	28	6892	11.54	6881	80.53	6892	56.29	6842	127.62	6834	104.43
	30	6683	11.49	6649	54.95	6660	55.89	6650	55.81	6664	55.40
	32	6468	11.56	6432	76.18	6457	76.09	6396	119.51	6406	117.48
	34	6271	11.36	6269	54.69	6201	116.96	6234	118.17	6217	118.04
	36	6027	11.43	6038	76.26	6009	53.97	6038	76.94	6038	74.48
	38	5875	11.43	5863	74.33	5871	75.33	5820	118.66	5849	95.82
	40	5710	11.49	5657	94.45	5633	93.05	5612	95.07	5616	93.90
	42	5546	11.56	5505	95.06	5484	94.64	5507	96.91	5520	74.89
	44	5394	11.50	5304	96.80	5300	116.79	5328	77.50	5352	97.82
	46	5272	11.50	5272	53.54	5198	118.06	5272	54.18	5212	122.61
	48	5150	11.43	5092	97.88	5108	77.82	5084	121.48	5101	119.00
	50	5063	11.38	5066	53.03	5068	53.33	5075	53.74	5071	54.07
	52	4944	11.29	4944	76.55	4926	76.12	4949	55.69	4950	55.22
	54	4838	12.13	4853	55.77	4850	56.06	4859	57.17	4859	55.92
	56	4735	12.09	4758	56.32	4748	56.17	4767	56.86	4767	56.93
	58	4657	11.83	4674	55.85	4670	102.39	4680	56.28	4680	56.31
	60	4590	11.73	4579	58.15	4575	79.88	4590	58.04	4590	56.53
	62	4486	11.79	4492	56.74	4499	56.40	4503	57.40	4503	57.95
	64	4424	11.91	4408	80.07	4420	80.10	4424	57.16	4424	58.85
	66	4343	11.76	4328	105.66	4328	105.02	4344	58.99	4308	131.91
	68	4255	10.36	4239	55.46	4241	101.67	4243	57.84	4246	55.63
	70	4157	11.38	4175	57.19	4153	58.09	4177	58.41	4177	58.31
	72	4083	12.15	4096	58.77	4077	82.40	4101	59.74	4102	59.12
	74	4010	13.10	3996	130.63	4010	60.59	3993	109.04	4013	61.46
	76	3963	14.23	3951	61.79	3941	83.72	3942	87.71	3956	88.26
	78	3877	15.15	3878	90.16	3851	111.68	3859	138.47	3869	138.43
	AVER	6918	11.76	6862	88.09	6855	92.43	6866	89.98	6866	91.29
	# Best	0		13		18		8		1	

Table 4
Total cost and time (s) for scenario 2 (no diversification)

N	*p*	Descent method		Random *TS* (1)		Fractional *TS* (2)		Soft aspiration with (1)		Soft aspiration with (2)	
		Cost	Time	Cost	Time	Cost	Time	Cost	Time	Cost	Time
50	4	84	0.01	84	0.04	84	0.04	84	0.04	84	0.04
	6	62	0.01	62	0.03	62	0.03	62	0.03	62	0.03
	8	51	0.01	51	0.03	51	0.03	51	0.03	51	0.03
	10	43	0.01	43	0.03	43	0.03	43	0.03	43	0.03
	12	36	0.01	36	0.03	36	0.03	36	0.03	36	0.03
	14	30	0.01	30	0.03	30	0.03	30	0.03	30	0.03
	16	26	0.01	26	0.03	26	0.03	26	0.03	26	0.03
	18	22	0.01	22	0.03	22	0.03	22	0.03	22	0.03
	20	19	0.01	19	0.03	19	0.03	19	0.03	19	0.03
	22	17	0.01	17	0.04	17	0.03	17	0.04	17	0.04
	24	14	0.01	14	0.04	14	0.03	14	0.04	14	0.04
	26	12	0.01	12	0.03	12	0.03	12	0.03	12	0.03
	28	10	0.01	10	0.03	10	0.03	10	0.03	10	0.03
	30	9	0.01	9	0.04	9	0.04	9	0.04	9	0.04
	AVER	31	0.01	31	0.03	31	0.03	31	0.03	31	0.03
	# Best	0		14		14		14		14	
287	4	1450	0.47	1450	7.65	1450	7.66	1450	7.71	1450	7.71
	6	1189	0.46	1189	7.69	1189	7.67	1189	7.79	1189	7.88
	8	1023	0.45	1019	10.43	1019	10.44	1019	7.12	1022	6.66
	10	930	0.44	906	5.91	906	8.43	906	5.86	906	10.70
	12	832	0.44	823	3.28	823	4.44	823	3.09	828	3.58
	14	760	0.45	758	4.13	758	2.94	758	2.95	760	2.91
	16	723	0.44	710	4.91	716	4.47	718	3.92	721	2.77
	18	680	0.43	668	4.54	672	2.87	666	5.53	678	2.74
	20	641	0.44	622	5.97	630	6.31	622	6.45	637	2.76
	22	609	0.44	591	6.88	599	4.01	593	4.79	609	2.73
	24	574	0.38	561	3.77	560	5.17	560	4.01	570	2.66
	26	543	0.42	538	5.95	541	5.29	539	5.10	542	2.62
	28	517	0.54	504	6.78	511	3.90	507	4.37	516	2.73
	30	496	0.86	487	7.15	487	4.38	487	6.07	496	2.99
	AVER	783	0.48	773	6.07	775	5.57	774	5.34	780	4.39
	# Best	0		4		0		1		0	
654	4	288220	12.57	288220	53.64	288220	53.24	288220	53.81	288220	53.44
	6	180614	12.60	180614	54.59	180614	54.45	180614	54.36	180614	54.61
	8	149304	12.55	149304	54.82	149304	54.85	149304	55.30	149304	54.72
	10	117312	12.68	117312	54.24	117312	54.33	117312	54.93	117312	54.94
	12	96032	12.73	96032	54.41	96032	54.48	96032	55.27	96032	55.07

Table 4 (*continued*)

N	p	Descent method		Random TS (1)		Fractional TS (2)		Soft aspiration with (1)		Soft aspiration with (2)	
		Cost	Time	Cost	Time	Cost	Time	Cost	Time	Cost	Time
	14	87130	12.67	87074	53.73	87074	53.85	87074	55.02	87074	53.88
	16	77920	12.77	77805	56.26	77804	53.89	77804	54.37	77804	53.58
	18	70327	12.73	70212	80.90	70212	71.84	70212	55.33	70212	51.93
	20	63899	12.87	63892	48.85	63892	48.82	63892	49.33	63892	48.85
	22	59601	12.82	59557	65.12	59555	64.48	59595	47.61	59595	47.86
	24	55353	12.81	55298	81.44	55331	73.33	55331	46.45	55336	44.15
	26	51791	12.77	51735	57.43	51768	45.09	51768	43.04	51768	43.45
	28	48617	12.73	48567	68.93	48589	56.21	48589	41.68	48590	42.36
	30	46454	12.83	46409	83.57	46424	43.31	46423	38.67	46424	38.68
	32	42705	12.70	42476	68.55	42520	44.73	42476	52.98	42520	45.43
	34	41268	12.73	41062	63.24	41126	45.48	41051	51.45	41126	46.32
	36	39831	12.69	39644	70.70	39713	38.69	39644	58.16	39713	39.42
	38	38506	12.76	38374	74.68	38387	51.03	38374	57.59	38387	51.77
	40	37433	12.68	37084	71.86	37132	50.27	37086	59.31	37132	37.74
	42	36043	12.66	35686	57.72	35750	36.38	35689	49.58	35744	44.38
	44	34855	12.72	34599	65.37	34628	36.32	34594	60.10	34633	36.11
	46	33758	12.71	33521	61.88	33528	38.51	33504	65.35	33528	37.06
	48	32842	12.72	32542	60.36	32601	36.20	32544	59.15	32601	36.51
	50	31877	12.64	31547	63.17	31634	51.24	31547	50.96	31634	50.58
	52	31312	12.64	30923	59.82	31100	36.77	30936	66.05	31100	36.74
	54	30510	12.62	30192	51.07	30300	37.78	30199	52.98	30301	37.84
	56	31673	10.99	31256	83.48	31589	35.93	31266	67.68	31589	36.11
	58	30446	11.87	29790	63.37	29850	43.45	29808	50.54	29850	43.29
	60	29586	12.89	28956	53.26	29133	40.36	28956	53.54	29133	39.89
	62	28692	13.79	28391	52.99	28403	44.29	28392	75.82	28403	55.53
	64	29267	15.00	28676	53.45	28699	42.07	28676	53.82	28699	48.76
	AVER	63650	12.74	63443	62.67	63491	48.12	63448	54.52	63492	45.84
	# Best	0		11		1		3		0	
1060	4	2067915	62.29	2067188	183.29	2067188	182.59	2067915	185.20	2067915	183.56
	6	1720884	62.97	1720884	183.94	1720605	183.07	1720884	185.73	1720605	183.84
	8	1452490	62.90	1442747	186.79	1450977	245.96	1452490	187.79	1450977	184.88
	10	1276508	62.42	1276508	186.82	1276508	185.77	1276508	189.77	1276508	187.03
	12	1130486	62.75	1130486	188.20	1130486	187.82	1130486	192.02	1130486	188.77
	14	1032095	63.03	1029573	302.40	1030976	191.37	1029615	274.41	1030976	192.18
	16	964134	63.22	962538	267.12	962538	193.06	962353	232.14	963083	194.02
	18	902127	62.87	901425	273.43	901915	197.11	901755	198.44	901915	194.03
	20	843886	63.33	842940	199.57	843454	196.28	842940	200.91	843835	193.99
	22	802113	63.57	798905	357.07	798725	322.04	798535	407.00	802113	195.52
	24	751458	63.53	751378	202.16	751378	197.45	751378	200.14	751378	197.78
	26	712668	62.66	711272	299.38	712032	270.91	711620	334.28	712256	196.16
	28	686106	62.69	680158	295.90	682390	281.31	681898	291.04	685850	189.25

Table 4 (*continued*)

N	p	Descent method		Random TS (1)		Fractional TS (2)		Soft aspiration with (1)		Soft aspiration with (2)	
		Cost	Time	Cost	Time	Cost	Time	Cost	Time	Cost	Time
	30	650652	62.94	647874	379.23	649554	215.82	647934	409.95	650156	188.15
	32	623666	62.99	618708	403.44	622340	249.59	621756	375.61	622340	195.31
	34	600760	63.10	596094	356.78	599070	343.78	599162	279.18	599638	218.81
	36	579228	62.92	577316	377.76	578036	356.96	577252	395.86	579228	175.02
	38	562286	62.98	559896	332.35	561958	309.66	559794	343.45	562286	170.36
	40	546608	62.69	543610	310.45	544978	283.69	543610	290.24	546608	168.00
	42	526442	62.23	523524	330.49	525360	217.01	523996	291.30	526442	164.46
	44	510752	62.04	506950	278.04	508298	222.84	506522	233.79	510752	166.89
	46	492132	62.31	490376	341.35	490568	240.84	490524	234.72	492132	166.51
	48	477726	62.42	476390	314.63	476438	174.66	477554	172.76	477726	167.86
	50	463212	62.74	461832	219.85	463212	166.82	463212	171.23	463212	167.31
	52	448636	62.69	447928	225.13	447892	242.07	447852	301.01	448636	169.46
	54	438044	62.91	436884	226.60	436884	231.66	436872	209.54	438044	169.41
	56	425364	62.78	423880	226.30	423960	228.20	424000	236.08	425364	169.41
	58	415532	62.57	415532	170.30	415532	170.91	415532	173.71	415532	171.63
	60	407564	62.27	407564	170.97	407564	170.98	407564	174.58	407564	171.34
	62	402772	62.24	402772	226.04	402772	227.43	402772	231.16	402772	227.54
	64	391252	62.45	391252	172.96	391252	173.20	391252	175.18	391252	173.04
	66	381528	62.30	381528	173.69	381528	175.36	381528	178.66	381528	175.07
	68	375020	62.11	375020	229.49	375020	231.20	375020	234.18	375020	230.54
	70	366520	62.42	366520	231.45	366520	234.14	366520	237.69	366520	232.75
	72	359808	62.56	359808	176.39	359808	178.73	359808	179.76	359808	177.55
	74	354064	62.58	354064	235.41	354064	237.88	354064	239.36	354064	236.79
	76	348624	62.62	348624	178.95	348624	181.00	348624	181.83	348624	180.36
	78	344296	62.39	344296	180.71	344296	182.34	344296	182.45	344296	181.77
	80	339840	62.51	339840	181.32	339840	183.31	339840	182.83	339840	182.28
	82	336004	62.12	336004	181.63	336004	184.27	336004	183.89	336004	182.59
	84	331536	62.39	331536	182.53	331536	185.49	331536	184.80	331536	183.45
	86	327012	61.66	327012	183.12	327012	186.47	327012	184.58	327012	183.04
	88	323104	61.71	323104	183.15	323104	186.87	323104	185.04	323104	183.49
	90	318488	51.59	313816	351.90	315040	275.84	312180	402.45	315064	234.95
	92	308980	53.96	308980	172.96	308980	176.63	308980	174.33	308980	172.57
	94	307596	56.11	307596	177.98	307596	181.18	307596	178.57	307596	177.18
	96	304100	58.74	304100	245.08	304100	250.21	304100	248.55	304100	244.98
	98	297724	61.11	297724	244.49	297724	248.26	297724	246.47	297724	243.88
	100	293816	63.64	293816	247.58	293816	252.17	293816	249.82	293816	247.18
	102	290576	66.52	290576	252.38	290576	256.91	290576	254.99	290576	252.52
	104	287368	68.76	287368	257.12	287368	262.07	287368	259.34	287368	257.05
	106	287488	71.24	287488	197.71	287488	202.48	287488	199.52	287488	197.58
	AVER	580557	62.39	579292	243.34	579901	221.42	579667	237.45	580339	193.06
	# Best	0		28		2		11		0	

References

[1] Reeves CR. Modern heuristic techniques for combinatorial problems. Oxford: Blackwell Scientific Publications, 1993.
[2] Salhi S. Heuristic search methods. In: Marcoulides G, editor. Modern methods in business research. New Jersey: Lawrence Erlbaum Associates. 1998. p. 147–76 [Chapter 6].
[3] Glover F. Future paths for integer programming and links to artificial intelligence. Computers and Operations Research 1986;13:533–49.
[4] Glover F, Laguna M, Taillard E, de Werra D, editors. Tabu search. Annals of Operations Research 1993; 41.
[5] Dammeyer F, Voss S. Dynamic tabu list management using the reverse elimination technique. Annals of Operations Research 1993;41:31–46.
[6] Glover F. Tabu search-Part I. ORSA Journal on Computing 1989;1:190–206.
[7] Glover F. Tabu thresholding: improved search for nonmonotonic trajectories. ORSA Journal on Computing 1995;7:426–42.
[8] Osman IH, Salhi S. Local search strategies for the vehicle fleet mix problem. In: Rayward-Smith VJ, Osman IH, Reeves CR, Smith GD, editors. Modern heuristic search techniques. New York: Wiley, 1996. p. 131–54.
[9] Battiti R, Tecchiolli G. The reactive tabu search. ORSA Journal on Computing 1994;6:126–40.
[10] Thomas P, Salhi S. A tabu search heuristic for the resource constrained project scheduling problem. Journal of Heuristics 1998;4:123–39.
[11] Voss S. A reverse elimination approach for the p-median problem. Studies in Locational Analysis 1996;8:49–58.
[12] Voss S. Dynamic tabu search strategies for the traveling purchaser problem. Annals of Operations Research 1996;63:253–75.
[13] Osman IH, Laporte G. Metaheuristics: a bibliography. Annals of Operations Research 1996;63:513–623.
[14] Glover F, Laguna M. Tabu search. Boston: Kluwer Academic Publishers, 1997.
[15] Beasley J. Lagrangean heuristics for location problem. European Journal Operational Research 1993;65:383–99.
[16] Agar M, Salhi S. Lagrangean heuristics applied to a variety of large capacitated plant location problems. Journal of the Operational Research Society 1998;49:1072–84.
[17] Erlenkotter D. A dual-based procedure for uncapacitaed facility location. Operations Research 1978;26:992–1009.
[18] Rolland E, Schilling DA, Current JR. An efficient tabu search procedure for the p-median problem. European Journal of Operational Research 1996;96:329–42.
[19] Rosing KE, ReVelle CS. Heuristic concentration: two stage solution construction. European Journal of Operational Research 1997;97:75–86.
[20] Salhi S. A perturbation heuristic for a class of location problems. Journal of the Operational Research Society 1997;48:1233–40.
[21] Salhi S, Atkinson RA. SUBDROP: a modified drop heuristic for location problems. Location Science 1995;3:267–73.
[22] Feldman E, Lehrer FA, Ray TL. Warehouse location under continuous economies of scale. Management Science 1966;12:670–84.
[23] Kuehn AA, Hamburger MJ. A heuristic program for locating warehouses. Management Science 1963;9:643–66.
[24] Christofides N, Mingozzi A, Toth P. The vehicle routing problem. In: Christofides N, Mingozzi A, Toth P, Sandi C, editors. Combinatorial Optimization. London: Wiley, 1978. p. 315–38.
[25] Brimberg J, Hansen P, Mladenovic N, Taillard ED. Improvements and comparison of Heuristics for solving the uncapacitated multisource Weber Problem. Operations Research 2000;48:444–60.

Said SALHI is a Senior Lecturer in Management Mathematics at the University of Birmingham, UK. His research interests include distribution management (routing, location), scheduling, heuristic search, applied mathematical programming to problems in transportation, and he has published numerous articles in these fields. Dr. Salhi gained his BSc in Mathematics from Algiers. Later he earned his MSc and Ph.D. in Operations Research from Southampton and Lancaster respectively.