A simulated annealing heuristic for the capacitated location routing problem

Article *in* Computers & Industrial Engineering · March 2010

DOI: 10.1016/j.cie.2009.10.007 · Source: DBLP

Vincent F. Yu

National Taiwan University of Science and Te…

Shih-Wei Lin

Chang Gung University

Computers & Industrial Engineering 58 (2010) 288–299

Contents lists available at ScienceDirect

Computers & Industrial Engineering

journal homepage: www.elsevier.com/locate/caie

A simulated annealing heuristic for the capacitated location routing problem

Vincent F. Yu [a], Shih-Wei Lin [b,*], Wenyih Lee [c], Ching-Jung Ting [d]

[a] Department of Industrial Management, National Taiwan University of Science and Technology, 43, Section 4, Keelung Road, Taipei 106, Taiwan
[b] Department of Information Management, Chang Gung University, 259 Wen-Hwa 1st Road, Kwei-Shan, Tao-Yuan 333, Taiwan
[c] Department of Business Administration, Chang Gung University, 259 Wen-Hwa 1st Road, Kwei-Shan, Tao-Yuan 333, Taiwan
[d] Department of Industrial Engineering and Management, Yuan Ze University, 135 Yuan-Tung Road, Chung-Li 32003, Taiwan

ARTICLE INFO

Article history:
Received 2 March 2009
Received in revised form 23 October 2009
Accepted 27 October 2009
Available online 1 November 2009

Keywords:
Location routing problem
Simulated annealing
Heuristic

ABSTRACT

The location routing problem (LRP) is a relatively new research direction within location analysis that takes into account vehicle routing aspects. The goal of LRP is to solve a facility location problem and a vehicle routing problem simultaneously. We propose a simulated annealing (SA) based heuristic for solving the LRP. The proposed SALRP heuristic is tested on three sets of well-known benchmark instances and the results are compared with other heuristics in the literature. The computational study indicates that the proposed SALRP heuristic is competitive with other well-known algorithms.

1. Introduction

The location routing problem (LRP) is a relatively new field which takes into account two key components of a logistics system, namely the facility location and vehicle routing. In various settings, these components are interdependent; therefore it is beneficial to consider the two components simultaneously. According to Min, Jayaraman, and Srivastava (1998), sequential methods that treat these two components separately have their limitation. Significant productivity gains can be achieved through the design of location routing models as these models can determine true least-cost solutions to a logistic problem taking into account both strategic policy (facility location) and operational decisions (vehicle routing). In fact, the effect of ignoring routes when locating depots has been discussed earlier by Salhi and Rand (1989). As a result, the location routing models have gained increasing popularity.

LRP is applicable to a wide variety of fields such as food and drink distribution (Dohrn, 1973), newspapers delivery (Jacobsen & Madsen, 1980; Madsen, 1983), waste collection (Kulcar, 1996), bill delivery (Lin, Chow, & Chen, 2002), military applications (Murty & Djang, 1999), parcel delivery (Bruns, Klose, & Stähly, 2000; Wasner & Zäpfel, 2004), and various consumer goods distribution (Aksen & Altinkemer, 2008; Bednar & Strohmeier, 1979). In the following, we discuss some of these applications. For a more detailed treatment, readers are referred to the original articles; most of them and other applications are summarized in a survey paper by Nagy and Salhi (2007).

Jacobsen and Madsen (1980) and Madsen (1983) described a practical LRP model for a newspaper delivery system. They introduced three heuristics to solve the three-layer LRP for the system, and obtained a slight improvement over the current distribution strategy. More recently, Lin et al. (2002) discussed an application of LRP models to the bill delivery services of a telecommunication service company in Hong Kong. The problem concerned the delivery of the company's monthly bills to densely populated housing estates by a company-owned delivery team. Aksen and Altinkemer (2008) formulated and solved an LRP model for the static conversion from traditional brick-and-mortar retailing to the hybrid click-and-mortar business model, from the perspective of distribution logistics.

The LRP belongs to the class of NP-hard problems. As the problem size increases, heuristic approaches become the only viable alternative. Therefore, we propose a simulated annealing based heuristic that features a special solution representation scheme for the LRP. The computational results suggest that the proposed approach competes well against existing approaches for the LRP.

The remainder of this paper is structured as follows. In the next section, we define the LRP and summarize the related literature. Section 3 details the proposed SALRP heuristic for the LRP. In Section 4, we discuss the computational study. Finally, conclusions are drawn and suggestions for future research are given in Section 5.

2. Problem definition and literature review

The location routing problem (LRP) can be stated as follows: Given a set of customers with known demand and a set of potential depot sites, determine the location of the depots and the vehicle routes from the depots to the customers to minimize the sum of

* Corresponding author. Tel.: +866 3 2118800; fax: +866 3 2118020.
E-mail addresses: swlin@mail.cgu.edu.tw, swlin@mail2000.com.tw (S.-W. Lin).

0360-8352/$ - see front matter © 2009 Elsevier Ltd. All rights reserved.
doi:10.1016/j.cie.2009.10.007

the costs associated with locating depots and distribution to the customers. There is a fixed cost associated with opening a depot at each potential site, and a distribution cost associated with the routing of vehicles which includes the route setup cost and the transportation cost which is linear in the total distance traveled by the vehicles. Each customer is assigned to a depot which will dispatch a vehicle to fulfill its demand. A vehicle route must start from and end at the same depot.

There may be other practical constraints, such as limitations on the total travel time of a route, the total distance traveled by a vehicle, the capacity of a vehicle, or the capacity of a depot. Most early work on LRP considers either capacitated routes or capacitated depots, but not both (Chien, 1993; Laporte, Nobert, & Taillefer, 1988; Srivastava, 1993). Recently, a number of studies have been devoted to the case with capacitated depots and routes (Prins, Prodhon, Ruiz, Soriano, & Wolfler Calvo, 2007; Prins, Prodhon, & Wolfler Calvo, 2006b; Wu, Low, & Bai, 2002).

In this paper, we focus on solving the LRP with capacitated depots and routes. Prins et al. (2007) gives the following formal mathematical model for the problem.

Let $G = (V, E)$ be an undirected network where V is a set of nodes comprised of a subset I of m potential depot sites and a subset $J = V/I$ of n customers. E is a set of edges connecting each pair of nodes in V. Associated with each edge $(i,j) \in E$ is a traveling cost c_{ij}. Each depot site $i \in I$ has a capacity W_i and an opening cost O_i. Each customer $j \in J$ has a demand d_j which must be fulfilled by a single vehicle. A set K of identical vehicles with capacity Q is available. Each vehicle, when used by a depot i, incurs a depot dependent fixed cost F_i and performs a single route. Each route must start and terminate at the same depot, and its total load must not exceed vehicle capacity. The total load of the routes assigned to a depot must fit the capacity of the depot. The objective is to determine which depots should be opened and which routes should be constructed to minimize the total cost.

Define binary variables $y_i = 1$ *iff* depot i is opened, $f_{ij} = 1$ *iff* customer j is assigned to depot i, and $x_{jlk} = 1$ *iff* edge (j, l) is traversed from j to l in the route performed by vehicle $k \in K$. Then the problem can be formulated as the following binary integer program.

$$\min z = \sum_{i\in I} O_i y_i + \sum_{i\in V}\sum_{j\in V}\sum_{k\in K} c_{ij}x_{ijk} + \sum_{k\in K}\sum_{i\in I}\sum_{j\in J} F_i x_{ijk} \qquad (1)$$

subject to

$$\sum_{k\in K}\sum_{i\in V} x_{ijk} = 1 \quad \forall j \in J \qquad (2)$$

$$\sum_{j\in J}\sum_{i\in V} d_j x_{ijk} \leqslant Q \quad \forall k \in K \qquad (3)$$

$$\sum_{j\in J} d_j f_{ij} \leqslant W_i y_i \quad \forall i \in I \qquad (4)$$

$$\sum_{j\in V} x_{ijk} - \sum_{j\in V} x_{jik} = 0 \quad \forall i \in V, \ k \in K \qquad (5)$$

$$\sum_{i\in I}\sum_{j\in J} x_{ijk} \leqslant 1 \quad \forall k \in K \qquad (6)$$

$$\sum_{i\in S}\sum_{j\in S} x_{ijk} \leqslant |S| - 1 \quad \forall S \subseteq J, \ k \in K \qquad (7)$$

$$\sum_{u\in J} x_{iuk} + \sum_{u\in V\setminus\{j\}} x_{ujk} \leqslant 1 + f_{ij} \quad \forall i \in I, \ j \in J, \ k \in K \qquad (8)$$

$$x_{ijk} \in \{0,1\}, \quad \forall i \in I, \ j \in V, \ k \in K \qquad (9)$$

$$y_i \in \{0,1\} \quad \forall i \in I \qquad (10)$$

$$f_{ij} \in \{0,1\} \quad \forall i \in I, \ j \in V \qquad (11)$$

The objective function (1) is the sum of depot opening costs and the routing costs, including the travel costs and the fixed costs associated with vehicle uses. Constraints (2) ensure that each customer belongs to exactly one route, and that each customer has only one predecessor in the route. Constraints (3) and (4) are capacity constraints associated with routes and depots, respectively. Constraints (5) and (6) guarantee the continuity of each route, and that each route terminates at the depot where the route starts. Constraints (7) are sub-tour elimination constraints. Constraints (8) ensure that a customer must be assigned to a depot if there is a route connecting them. Finally, constraints (9), (10), and (11) specify the binary variables used in the formulation.

Obviously, the LRP belongs to the class of NP-hard problems since it combines two difficult sub-problems: the facility location problem (FLP) and the vehicle routing problem (VRP), both of them are shown to be NP-hard (Cornuéjols, Fisher, & Wolsey, 1977; Karp, 1972; Lenstra & Rinnooy Kan, 1981). Due to its complexity, only limited exact methods that were usually based on a mathematical programming formulation were proposed in early studies. They often use the relaxation and reintroduction of constraints such as sub-tour elimination, chain barring, and integrity.

Laporte and Nobert (1981) proposed an exact algorithm for the single facility fixed fleet size LRP without tour length restrictions. They formulated the problem as an integer linear program and solved it by first relaxing integrality constraints, and then using a branch and bound technique to achieve integrality. They recognized that the optimal depot location rarely coincides with the node closest to the center of gravity. Similar approach was applied to uncapacitated multi-facility LRP by Laporte, Nobert, and Pelletier (1983) and capacitated multi-facility LRP by Laporte et al. (Laporte, Nobert, & Arpin, 1986). The latter applied a branching procedure where sub-tour elimination and chain barring constraints were reintroduced.

Bookbinder and Reece (1988) formulated a three-layer multi-commodity, capacitated distribution system as a nonlinear mixed integer program, and decompose the problem into its location and routing components by Benders' decomposition.

Due to the exponential growth in the problem size, exact approaches for the LRP have been limited to small and medium size instances with 20–50 customers. For this reason, heuristics and meta-heuristics are often used to solve realistic sized LRP instances in more recent studies.

Tuzun and Burke (1999) developed a two-phase tabu search (TS), but for the LRP with capacitated routes and uncapacitated depots so a depot may have as many routes as desired. The two phases of their TS algorithm are also dedicated to routing and location. The algorithm iteratively adds a depot to the current solution until it degrades the solution. The authors also reported results for instances with up to 200 clients.

Su (1999) applied a genetic algorithm to the design of a physical distribution system where both the location of facilities and the routing of vehicles were considered. In addition to determining the number and locations of distribution centers, the author also developed a methodology to estimate the required number of vehicles and corresponding routing.

Wu et al. (2002) studied a simulated annealing based decomposition approach for the multi-depot LRP with heterogeneous fleet types with limited number of vehicles and capacity on both depots and routes. Their SA algorithm incorporated a tabu list to avoid cycling.

Barreto (2004) developed a class of three-phase heuristics based on clustering techniques. In the first phase, customers were aggregated into clusters fitting vehicle capacity. The second phase solved a traveling salesman problem (TSP) for each cluster. Finally in the third phase, the depots to be opened were determined by solving an FLP, where the TSP cycles were combined to form supernodes. The author also reported lower bounds for some small-scaled instances with up to 5 depots and 36 clients.

Albareda-Sambola, Díaz, and Fernández (2005) proposed another two-phase TS heuristic for the LRP with one single route per capacitated open depot. The intensification phase optimized the routes and the diversification phase modified the set of open depots. The TS heuristic was tested on small instances with at most 30 customers.

Prins et al. (2006b) solved the LRP with capacitated depots and routes by combining greedy randomized adaptive search procedure (GRASP) with a learning process and a path relinking mechanism. They first used a GRASP, based on an extended and randomized version of Clarke and Wright algorithm and implemented with a learning process on the choice of depots. After that, path relinking was used as a post-optimization procedure to generate new solutions.

Most recently, Prins et al. (2007) combined the Lagrangian relaxation technique with Granular tabu search (GTS) to develop another iterative two-phase approach to solve the LRP and obtained promising results. The algorithm alternated between a depot location phase and a routing phase, exchanging information on the most promising edges. In the first phase, routes and their customers were combined to form supercustomers to transform the original LRP into a facility location problem. The Lagrangian relaxation of the assignment constraints were used to solve the resulting FLP. In the second phase, GTS was used to improve the multi-depot VRP solution obtained in the first phase.

Other popular heuristics and meta-heuristics were applied to the LRP as well. Prins, Prodhon, and Wolfler Calvo (2004) proposed a multi-start approach for the LRP. Bouhafs, Hajjam, and Koukam (2006) combined simulated annealing and ant colony system to solve the capacitated LRP. Prins, Prodhon, and Wolfler Calvo (2006a) developed a memetic algorithm with population management (MA|PM) for the capacitated location routing problem. More recently, Marinakis and Marinaki (2008) introduced a hybrid metaheuristic for the LRP based on particle swarm optimization and path relinking. Duhamel, Lacomme, and Prodhon (2008) proposed a memetic algorithm (genetic algorithm hybridized with a local search procedure; GAHLS) to the LRP. The experimental result showed that their approach competes well for small and medium scale instances with the best existing methods.

There are a few LRP surveys in the literature (Berman, Jaillet, & Simchi-Levi, 1995; Laporte, 1988; Min et al., 1998; Nagy & Salhi, 2007). The most recent literature review of the LRP and its extensions is due to Nagy and Salhi (2007). They proposed a classification scheme for LRP variants and surveyed exact and heuristic algorithms for the LRP. They also pointed out some future research directions for LRP study. Earlier, Min et al. (1998) gave a synthesis and survey of the LRP. They defined the location routing model as solving the joint problem of determining the optimal number, capacity and location of facilities serving more than one customer/supplier (demand node), and finding the optimal set of vehicle schedules and routes. The authors stressed that the main difference between the LRP and the classical location/allocation problem is that, once the facility is located, LRP requires a visitation of demand nodes through tours, whereas the latter assumes a straight-line or radial trip from the facility to the clients.

Note that although the LRP is normally formulated as a deterministic node routing problem, a few authors have proposed arc routing versions of the LRP (Ghiani & Laporte, 2001; Labadi, 2003) while others studied the stochastic versions of the LRP (Chan, Carter, & Burnes, 2001; Laporte, Louveaux, & Mercure, 1989). Interested readers are referred to the survey by Berman et al. (1995) for stochastic LRP.

3. Simulated annealing heuristic for the LRP

We propose an SALRP heuristic for the LRP based on the popular simulated annealing heuristic. Simulated annealing is a local search-based heuristic that is capable of escaping from being trapped into a local optimum by accepting, with small probability, worse solutions during its iterations. It has been applied successfully to a wide variety of highly complicated combinatorial optimization problems as well as various real-world problems. SA was introduced by Metropolis, Rosenbluth, Rosenbluth, Teller, and Teller (1953) and popularized by Kirkpatrick, Gelatt, and Vecchi (1983). The concept of the method is adopted from the "annealing" process used in the metallurgical industry. Annealing is the process by which slow cooling is applied to metals to produce better aligned, low energy-state crystallization. The optimization procedure of SA searches for a (near) global minimum mimicking the slow cooling procedure in the physical annealing process. It starts from a random initial solution. At each iteration, a new solution is taken from the predefined neighborhood of the current solution. The objective function value of this new solution is then compared with that of the current best solution in order to determine if an improvement has been achieved. If the objective function value of the new solution is better, that is, being smaller in the case of minimization, the new solution becomes the current solution from which the search continues by proceeding with a new iteration. A new solution with a degraded (larger) objective function value may also be accepted as the new current solution, with a small probability determined by the Boltzmann function, $\exp(-\Delta/kT)$, where Δ is the difference of objective function values between the current solution and the new solution, k is a predetermined constant and T is the current temperature. The basic idea is not to restrict the search to those solutions that decrease the objective function value, but also allow moves that increase the objective function value. This mechanism may avoid the procedure being trapped prematurely in a local minimum.

In the following subsections, we discuss the proposed SALRP heuristic in detail, including the solution representation, the generation of the initial solution, the calculation of the objective function value, various types of neighborhood, the parameters used, and the SALRP procedure.

3.1. Solution representation

Although the proposed SALRP heuristic is fairly standard, it features a specially designed solution representation scheme for the LRP. Using this scheme, the routes may be randomly terminated by the dummy zeros, or by the route capacity constraints. The latter condition is used in many heuristic approaches for VRP related problems. By allowing a route to be terminated randomly, the solution space may be enlarged, and a better solution may be found, at the expense of computational time. Thus a random neighborhood structure may be necessary to speed up the algorithm (see Section 3.4). Two sections are devoted to this new solution representation scheme due to the complex decoding process involved.

A solution is represented by a string of numbers consisting of a permutation of n customers denoted by the set $\{1, 2, \ldots, n\}$, m potential depots denoted by the set $\{n+1, n+2, \ldots, n+m\}$, and N_{dummy} zeros which are used to separate routes, in addition to the vehicle capacity constraints. The ith number in $\{1, 2, \ldots, n\}$ denotes the ith customer to be serviced. The first number in a solution is always in $\{n+1, n+2, \ldots, n+m\}$ indicating the first depot under consideration. The parameter N_{dummy} is calculated as $\lceil \sum_i \frac{d_i}{Q} \rceil$, where d_i is the demand of customer i, Q is the capacity of vehicle, and $\lceil \bullet \rceil$ denotes the smallest integer which is larger than or equal to the enclosed number.

The solution representation is further explained as following. Each depot services customers between the depot and the next depot in the solution representation. The first route of this depot starts by servicing the first customer after the depot. Other customers for this depot are added to the current route one at a time. If adding a customer will exceed the vehicle's capacity, the current route is terminated. If the next number in the solution representation is a dummy zero, the current route will also be terminated. A new route will be started to service remaining customers assigned to this depot.

It can be verified that this solution representation always gives a LRP solution without violating the capacity constraint of the vehicle. However, the capacity constraint of the depot may be violated.

3.2. Illustration of solution representation

Tables 1 and 2 together give a small LRP instance with 20 customers and 5 potential depots. The location (X, Y) and demand (W) of customers are listed in Table 1. The location (X, Y) and setup cost for opening a depot are given in Table 2. For each depot, the depot capacity is 140, the vehicle capacity is 70, and the fixed cost for setting up a route is 1000. A randomly generated sample solution for this instance is shown in Fig. 1. Five dummy zeros are present in the solution. Fig. 2 gives a visual illustration of the sample solution.

In this example, the first number is depot 24, followed by two zeros and depot 25. Since there are no actual customers between depot 24 and depot 25, depot 24 is closed. Customers (10, 9, 17, and 2) between depot 25 and depot 23 are real customers, thus depot 25 is opened to service these customers. The first route of depot 25 services customers 10, 9, 17 and 2. Customer 2 is followed by depot 23, thus the first route of depot 5 is terminated. No other routes are needed for depot 25.

The first route of depot 23 services customers 14, 15, 16 and 19. Customer 19 is followed by a zero, so the first route is terminated. The second route of depot 23 then starts to service customers 8, 11 and 6. Customer 6 is followed by depot 22, so the second route is terminated.

Depot 22 is followed by two consecutive zeros. Since there are no real customers between the zeros and the depot, these two zeros can be ignored. The first route of depot 22 services customers 4, 1, 12 and 18. Because adding customer 20 exceeds the vehicle capacity, the first route is terminated. The second route services customers 20, 13, 5, 7, and 3. Customer 3 is followed by depot 21, so the second route is terminated.

At this point, all customers are serviced so no other depots need to be opened and the decoding process can be terminated. In this solution, depots 22, 23, and 25 are open with two, two and one routes, respectively.

The solution representation has determined which depot is open and the customers on each route. Once this is done, it is easy to calculate the objective function value, obj(X, P), of a given solution X. Note that the capacity of depots is not taken into consideration during the decoding process so a per unit penalty cost P is added to the objective function value whenever the total demand serviced by a depot exceeds its capacity.

Table 1
Customer location and demand.

Customer no.	X	Y	W
1	20	35	17
2	8	31	18
3	29	43	13
4	18	39	19
5	19	47	12
6	31	24	18
7	38	50	13
8	33	21	13
9	2	27	17
10	1	12	20
11	26	20	16
12	20	33	18
13	15	46	15
14	20	26	11
15	17	19	18
16	15	12	16
17	5	30	15
18	13	40	15
19	38	5	15
20	9	40	16

Table 2
Depot location and opening cost.

Depot no.	X	Y	Depot opening cost
21	6	7	10,841
22	19	44	11,961
23	37	23	6091
24	35	6	7570
25	5	8	7497

3.3. Initial solution

The initial solution is constructed by the following greedy method with the hope that a good initial solution can be found within a reasonable time.

Step 1. Let U be the set of unused depots. For each depot i in U, let $cc(i)$ be the number of unassigned customers whose closest depot in U is depot i. Choose the depot in U with the highest cc value. If there is a tie, select the depot with the highest capacity.
Step 2. For all unassigned customers, assign them to the chosen depot one by one in the increasing order of the distance between the customer and the chosen depot. Stop when the capacity of the depot is violated.
Step 3. Construct a TSP route which starts from and ends at the depot using the Lin and Kernighan's heuristic (1973), which is a fast and effective heuristic for TSP problems.
Step 4. Split the TSP route constructed by the Lin and Kernighan's heuristic into several routes so that the route capacity constraint is not violated.
Step 5. If there are still unassigned customers, go to Step 1; otherwise, terminate the procedure and encode the current solution using the solution representation described in Sections 3.1 and 3.2.

The current solution is converted into the solution representation used in SALRP algorithm as following. The first selected open depot is placed in the first position of the solution representation followed by customers on the first route of this depot, customers on the second route of this depot, and so forth. Customers are added to the solution representation according to their order of appearance on the route. When all customers assigned to the first open depot are included in the solution representation, the open depot selected second is added to the solution representation. Continue in this fashion until all open depots and customers are included in the solution representation. Then closed depots are appended to the solution representation. Finally, N_{dummy} zeros are included at the end of the solution representation.

3.4. Neighborhood

We use a standard SA procedure with a random neighborhood structure that features various types of moves, including insertion

24	0	0	25	10	9	17	2	23	14	15	16	19	0	8	11	6	22	0	0	4	1	12	18	20	13	5	7	3	21

Fig. 1. An example of solution representation.

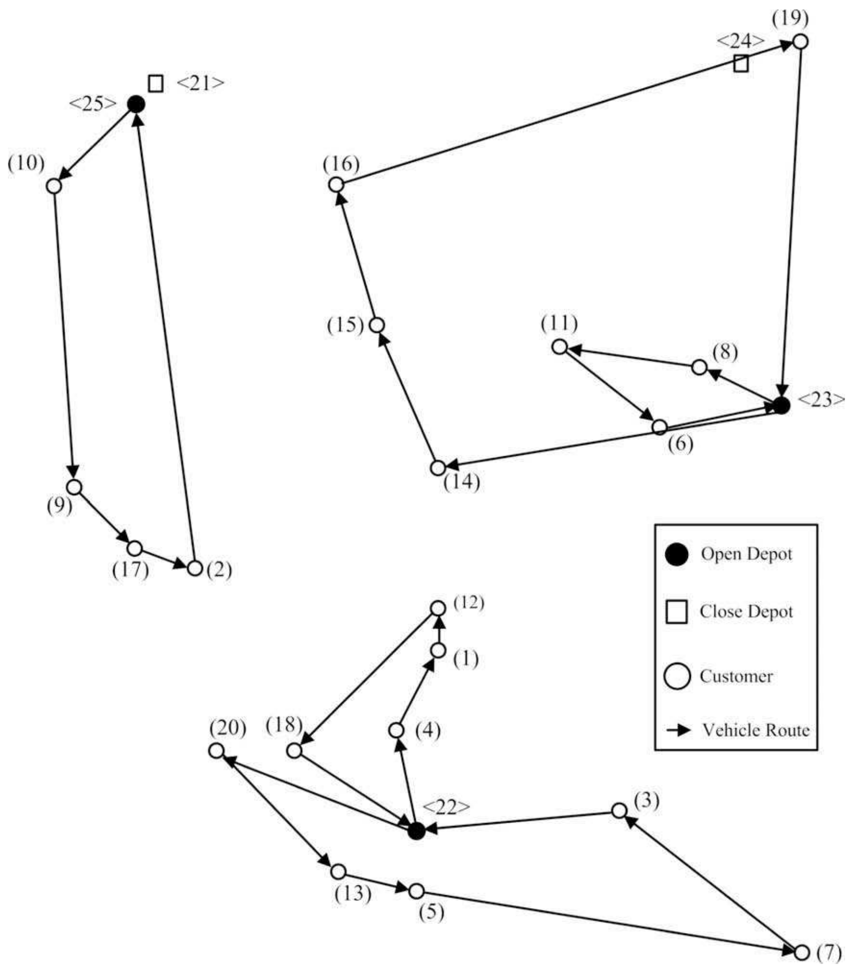

Fig. 2. Visual illustration of the example solution given in Fig. 1.

move, swap move, and 2-opt move, to solve the LRP. These moves are commonly embedded in SA heuristics, and other meta-heuristics. We define the set $N(X)$ to be the set of solutions neighboring a solution X. At each iteration, the next solution Y is selected from $N(X)$ either by insertion move, swap move, or 2-opt move. Dummy zeros are treated as customers when performing these moves.

The insertion move is carried out by randomly selecting the ith number of X and inserting it into the position immediately before another randomly selected jth number of X. The insertion move has several different effects as following.

Case 1: i and j are both customers. Customer i is reassigned to the depot that serves customer j.
Case 2: i and j are both depots. Depot i will be closed, and its customers will be reassigned to the depot before depot i.
Case 3: i is a customer and j is a depot. Customer i will be serviced by the depot immediately before depot j.
Case 4: i is a depot and j is a customer. Depot i will be servicing a new set of customers starting from customer j. All or part of depot i's customers will be serviced by the depot before depot i. Case 4 has greater impact on the structure of the solution which may help prevent the algorithm from being trapped at a local optimum.

The swap move is performed by randomly selecting the ith and the jth numbers of X, and then exchanging the positions of these two numbers. The swap move also has several different effects as following.

Case 1: i and j are both customers. Customer i is reassigned to the depot servicing customer j, and vice versa.
Case 2: i and j are both depots. Depot i will be servicing depot j's customers, and vice versa.
Case 3: i is a customer and j is a depot. Depot j will be servicing a new set of customers (customer after i) and customer i will be reassigned to the depot before depot j. All or part of depot j's original customers may be reassigned to the depot before depot j.
Case 4: i is a depot and j is a customer. This case is similar to Case 3. Cases 3 and 4 may change the original solution dramatically which may help the algorithm explore new part of the solution space.

Since the positions of depots may have greater impact on the solution structure, we set the probability of selecting a depot to perform insertion or swap move to be 20%. It can be verified that, except for seven small instances, the probability of selecting a depot is greater than that of selecting a customer or a dummy zero.

The 2-opt move, commonly used in solving VRP related problems, is modified and applied to improve existing routes. This is implemented by randomly selecting two customers that are assigned to the same depot, and then reverse the substring in the solution representation between them. Note that because of the special structure of our solution representation, the results differ from those of traditional 2-opt move slightly. If the two selected customers belong to the same route, then reversing the substring between them is essentially the same as the common 2-opt inner route move. However, the effect of this move becomes more complicated when they are on different routes. If the two routes that service these two customers are adjacent routes, and both routes are terminated by a dummy zero, rather than the load capacity constraints, then the move is similar to the traditional 2-opt inter route move. Otherwise, all routes after the first route in the substring may also be affected because of the vehicle capacity constraints, and rearranging these routes may be necessary. Nevertheless, this modified 2-opt approach results in a neighborhood that contains the traditional 2-opt move neighborhood which may facilitate the diversion of the search process.

The probabilities of choosing the swap move, insertion move, and 2-opt move are set to be 1/3, 1/3 and 1/3, respectively. Note that the insertion move and the swap move may result in infeasible solutions, with respect to the vehicle capacity constraints and the depot capacity constraints. In addition, the condition that the first entry in a solution representation must be a depot, i.e., in $\{n+1, n+2, \ldots, n+m\}$, may also be violated. The violation of vehicle capacity constraints can be repaired by dispatching additional vehicles and rearranging routes for subsequent customers. Moves that violate the depot capacity constraints are accepted with a penalty as described in Section 3.2. A move that results in a new solution that does not start with a depot will be discarded, and a new move will be generated.

3.5. Parameter settings

The SALRP uses seven parameters I_{iter}, T_0, T_F, K, P, $N_{non\text{-}improving}$ and α. I_{iter} denotes the number of iterations the search proceeds at a particular temperature. T_0 represents the initial temperature, while T_F is the final temperature below which the SALRP procedure is terminated. K is the Boltzmann constant used in the probability function to determine whether to accept a worse solution or not. P is the unit penalty cost associated with the violation of depot capacity. This penalty term is set to be 0 in cases with uncapacitated depots. $N_{non\text{-}improving}$ is the maximum allowable number of consecutive temperature reductions during which the best objective function value is not improved. Finally, α is the coefficient controlling the cooling schedule.

3.6. The SALRP procedure

In the beginning, the current temperature T is set to be the same as T_0. Then an initial solution X is generated using the greedy procedure described in Section 3.3, in which the first number in the solution representation must be a depot. The current best solution X_{best} and the best objective function value obtained so far are set to be X and obj(X, P), respectively.

At each iteration, the next solution Y is generated from $N(X)$ and its objective function value is evaluated. Let Δ denote the difference between obj(X, P) and obj(Y, P), that is $\Delta = \text{obj}(Y, P) - \text{obj}(X, P)$. The probability of replacing X with Y, given that $\Delta > 0$, is $\exp(-\Delta/KT)$. This is accomplished by generating a random number $r \in [0, 1]$ and replacing the solution X with Y if $r < \exp(-\Delta/KT)$. Meanwhile, if $\Delta \leqslant 0$, the probability of replacing X with Y is 1. X_{best} records the best solution found so far as the algorithm progresses.

The current temperature T is decreased after running I_{iter} iterations since the previous decrease, according to the formula $T \leftarrow \alpha T$, where $0 < \alpha < 1$. After each temperature reduction, a local search procedure which sequentially performs swap and insertion is used to improve the current best solution.

The algorithm is terminated when the current temperature T is lower than T_F or the current best solution X_{best} is not improved in $N_{non\text{-}improving}$ consecutive temperature reductions. Following the termination of the SALRP procedure, the facility locations and vehicle routes can be derived from X_{best}. The proposed SALRP approach is summarized in Fig. 3.

4. Computational study

The proposed SA based meta-heuristic was implemented in C language and run on a PC with an Intel Core 2 Quad CPU (2.6 GHz) and 2 GB memory. In order to verify the proposed approach, three well-known LRP problem sets are selected as test problems. For each instance, only one run of the proposed approach is executed in order to have a fair comparison with other approaches.

4.1. Instances

The first set of 19 instances was gathered by Barreto (2004). The instances in this data set are either from the literature or are obtained by adding depots to existing classical VRP instances. All routes in this data set are capacitated, and except for a few instances, the depots are also capacitated. There are no variable costs associated with depots and the traveling costs are not rounded.

The second set was created by Prins et al. (2004) which contains 30 LRP instances with capacitated routes and depots. Customers in this dataset may belong to different clusters. The number of depots is set to be either 5 or 10, the number of clients is either 20, 50, 100, or 200, the vehicle capacity is set to be either 70 or 150, and the number of clusters is in {0, 2, 3} where 0 clusters corresponds to a uniform distribution in the Euclidean plane. Each demand follows a uniform distribution in [11, 20]. The traveling costs correspond to the Euclidean distances, multiplied by 100 and rounded up to the next integer. The other data (demands, depot capacities and fixed costs) are also integers.

The third set was designed by Tuzun and Burke (1999) which comprises 36 instances with capacitated routes and uncapacitated depots. This data set was used by the authors to evaluate their tabu search heuristic. They set the number of clients to be 100, 150 or 200. The number of depots is either 10 or 20. The vehicle capacity is set to be 150, and the demand is assumed to be uniformly distributed in interval [1, 20]. In this data set, distances are not rounded.

Note that whether the distances are rounded or not were specified by the creator of each dataset, respectively. Thus, all algorithms included in the comparison were applied using the same distances.

4.2. Parameter settings

Parameter selection may influence the quality of the computational results. In the initial experiments, the following combinations of parameters were tested.

SALRP(T_0, T_F, α, K, P, $N_{non\text{-}improving}$, I_{iter})

Step 1: Let $N_{dummy} = \left\lceil \sum_i \frac{d_i}{Q_k} \right\rceil$.

    Generate the initial solution X by the greedy heuristic.

Step 2: Let T=T_0; I=0; N=0; F_{best}=obj(X, P); X_{best}=X;

Step 3: I=I+1;

Step 4: (Generate a solution Y based on X)

    Step 4.1: Generate r = random (0, 1);

    Step 4.2: Case r≤1/3: Generate a new solution Y from X by random swap operation;

        Case 1/3<r≤2/3: Generate a new solution Y from X by random insertion operation;

        Case 2/3<r≤1: Generate a new solution Y from X by random 2-opt operation;

Step 5: If Δ=obj(Y, P) - obj(X, P) ≤0 {Let X=Y;}

    Else {

        Generate r = random (0,1);

        If r<exp(-Δ/KT) {Let X=Y;}

    }

Step 6: If (obj(X, P)<F_{best} and X is feasible){X_{best}=X; F_{best}=obj(X, P);N=0;}

Step 7: If I=I_{iter} {

        T=αT; I=0;N=N+1;

        Perform Local search based on swap operation on X_{best};

        Perform Local search based on insertion operation on X_{best};

    }

    Else {Go to Step 3;}

Step 8: If T<T_F or N=$N_{non\text{-}improving}$ {Terminate the SA heuristic;}

    Else {Go to Step 3;}

Fig. 3. Pseudo-code of the proposed SALRP.

Table 3
Computational results for the first problem set.

Prob. ID	*n*	*m*	Q	BKS	CH	SA-ACS	GRASP	MA|PM	LRGTS	GAHLS	SALRP	SALRP*
B1	21	5	6000	424.9	435.9	430.4	**424.9**	**424.9**	**424.9**	**424.9**	**424.9**	**424.9**
B2	22	5	4500	585.1	591.5	586.7	**585.1**	611.8	587.4	**585.1**	**585.1**	**585.1**
B3	29	5	4500	512.1	**512.1**	**512.1**	515.1	**512.1**	**512.1**	**512.1**	**512.1**	**512.1**
B4	32	5	8000	562.2	571.7	569.3	571.9	571.9	587.4	**562.2**	**562.2**	**562.2**
B5	32	5	11,000	504.3	511.4	506.1	**504.3**	534.7	504.8	**504.3**	**504.3**	**504.3**
B6	36	5	250	460.4	470.7	470.4	**460.4**	485.4	476.5	**460.4**	**460.4**	**460.4**
B7	50	5	160	565.6	582.7	–	599.1	**565.6**	586.4	584.8	**565.6**	**565.6**
B8	75	10	140	844.4	886.3	–	861.6	866.1	863.5	851.8	848	**844.4**
B9	100	10	200	833.4	889.4	–	861.6	850.1	842.9	842.4	838.3	**833.4**
B10	12	2	140	204	**204**	**204**	–	–	–	–	**204**	**204**
B11	55	15	120	1112.1	1136.2	1118.4	–	–	–	–	1112.8	**1112.1**
B12	85	7	160	1622.5	1656.9	1651.3	–	–	–	–	**1622.5**	**1622.5**
B13	318	4	25,000	557275.2	580680.2	–	–	–	–	–	563493.1	**557275.2**
B14	318	4	8000	673297.7	747619	–	–	–	–	–	684163.5	**673297.7**
B15	27	5	2500	3062	**3062**	**3062**	**3062**	**3062**	3065.2	**3062**	**3062**	**3062**
B16	134	8	850	5709	6238	6208.8	5965.1	5950.1	5809	–	**5709**	**5709**
B17	88	8	9000,000	355.8	384.9	–	356.9	**355.8**	368.7	355.9	**355.8**	**355.8**
B18	150	10	8000,000	43919.9	46642.7	–	44625.2	44011.7	44386.3	–	45109.4	**43919.9**
B19	117	14	150	12290.3	12474.2	–	–	–	–	–	12434.5	**12290.3**

BKS: solutions obtained either by the algorithms in their published version or during their parameter analysis phase.
SALRP*: best solution obtained during the parameter analysis phase of the proposed approach.
–: the problem is not solved in the corresponding study.
Bold numbers indicate that best known solution values are attained by the corresponding approach.

Table 4
Further computational results for the first problem set.

Prob. ID	CH		SA-ACS		GRASP		MA|PM		LRGTS		GAHLS		SALRP		SALRP*
	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)
B1	2.59	N/A	1.29	N/A	0.00	0.2	0.00	0	0.00	0.2	0.00	0	0.00	18.3	0.00
B2	1.09	N/A	0.27	N/A	0.00	0.2	4.56	4.56	0.39	0.2	0.00	0	0.00	16.6	0.00
B3	0.00	N/A	0.00	N/A	0.59	0.4	0.00	0	0.00	0.4	0.00	1	0.00	23.9	0.00
B4	1.69	N/A	1.26	N/A	1.73	0.6	1.73	1.73	4.48	0.6	0.00	1	0.00	27	0.00
B5	1.41	N/A	0.36	N/A	0.00	0.5	6.03	6.03	0.10	0.5	0.00	3	0.00	25.1	0.00
B6	2.24	N/A	2.17	N/A	0.00	0.8	5.43	5.43	3.50	0.7	0.00	19	0.00	31.7	0.00
B7	3.02	N/A	–	–	5.92	2.3	0.00	0	3.68	2.4	3.39	80	0.00	52.8	0.00
B8	4.96	N/A	–	N/A	2.04	9.8	2.57	0.85	2.26	10.1	0.88	207	0.43	126.8	0.00
B9	6.72	N/A	–	N/A	3.38	25.5	2.00	1.13	1.14	28.2	1.08	408	0.59	330.8	0.00
B10	0.00	N/A	0.00	N/A	–	–	–	–	–	–	–	–	0.00	6.8	0.00
B11	2.17	N/A	0.57	N/A	–	–	–	–	–	–	–	–	0.06	112.4	0.00
B12	2.12	N/A	1.78	N/A	–	–	–	–	–	–	–	–	0.00	213.1	0.00
B13	4.20	N/A	–	N/A	–	–	–	–	–	–	–	–	1.12	2806.5	0.00
B14	11.04	N/A	–	N/A	–	–	–	–	–	–	–	–	1.61	3352.5	0.00
B15	0.00	N/A	0.00	N/A	0.00	0.4	0.00	0	0.10	0.3	0.00	10	0.00	23.3	0.00
B16	9.27	N/A	8.75	–	4.49	49.6	4.22	2.58	1.75	48.3	–	–	0.00	522.4	0.00
B17	8.18	N/A	–	–	0.31	17.3	0.00	0	3.63	17.5	0.03	582	0.00	226.9	0.00
B18	6.20	N/A	–	–	1.61	156	0.21	0	1.06	119.2	–	–	2.71	577	0.00
B19	1.50	N/A	–	–	–	–	–	–	–	–	–	–	1.17	323.4	0.00
Avg.	3.60	–	1.50	–	1.54	–	2.06	–	1.70	–	0.49	–	0.40	464.1	0.00

GAP: relative percentage gap calculated as (solution values obtained by individual algorithm − BKS)/BKS.
–: the problem is not solved in the corresponding study.
N/A: the data is not provided in the corresponding study.

$\alpha = 0.96, 0.97, 0.98, 0.99$;
$I_{\text{iter}} = 1000L, 2000L, 3000L, 4000L, 5000L, 6000L$, where L denotes the length of solution representation;
$P = 100, 200, \ldots, 2000$;
$K = 1/1, 1/2, \ldots, 1/15$.

Setting $\alpha = 0.98$, $I_{iter} = 5000L$, $P = 400$, and $K = 1/9$ seemed to give best results. Therefore they were used for further computational study. Other parameters used in the final analysis are: $T_0 = 30$, $= 0.1$, and $N_{non\text{-}improving} = 100$. Since $T_0\alpha^{283} = 30 \times 0.98^{283} < 0.1 = T_F$, the current temperature will fall below the final temperature

Table 5
Computational results for the second problem set.

Prob. ID	n	m	Q	BKS	MSLS	GRASP	MA|PM	LRGTS	GAHLS	SALRP	SALRP*
P1	20	5	70	54,793	55,806	55,021	**54,793**	55,131	**54,793**	**54,793**	**54,793**
P2	20	5	150	39,104	39,104	39,104	**39,104**	39,104	**39,104**	**39,104**	**39,104**
P3	20	5	70	48,908	49,668	48,908	**48,908**	48,908	**48,908**	**48,908**	**48,908**
P4	20	5	150	37,542	37,542	37,542	**37,542**	37,542	**37,542**	**37,542**	**37,542**
P5	50	5	70	90,111	98,079	90,632	90,160	90,160	**90,111**	**90,111**	**90,111**
P6	50	5	150	63,242	72,159	64,761	**63,242**	63,256	63,469	**63,242**	**63,242**
P7	50	5	70	88,298	90,188	88,786	**88,298**	88,715	88,709	**88,298**	**88,298**
P8	50	5	150	67,308	68,675	68,042	67,893	67,698	67,353	67,340	**67,308**
P9	50	5	70	84,055	–	**84,055**	**84,055**	84,181	84,409	**84,055**	**84,055**
P10	50	5	150	51,822	–	52,059	**51,822**	51,992	51,902	**51,822**	**51,822**
P11	50	5	70	86,203	96,756	87,380	**86,203**	**86,203**	86,203	86,456	**86,203**
P12	50	5	150	61,830	61,844	61,890	**61,830**	**61,830**	62,763	62,700	**61,830**
P13	100	5	70	275,419	284,613	279,437	281,944	277,935	281,564	277,035	**275,419**
P14	100	5	150	213,615	218,052	216,159	216,656	214,885	219,056	216,002	**213,615**
P15	100	5	70	193,671	198,041	199,520	195,568	196,545	197,156	194,124	**193,671**
P16	100	5	150	157,150	159,812	159,550	157,325	157,792	159,615	**157,150**	**157,150**
P17	100	5	70	200,079	204,906	203,999	201,749	201,952	203,723	200,242	**200,079**
P18	100	5	150	152,441	155,865	154,596	153,322	154,709	154,404	152,467	**152,441**
P19	100	10	70	287,983	323,438	323,171	316,575	291,887	325,357	291,043	**287,983**
P20	100	10	150	231,763	277,678	271,477	270,251	235,532	274,379	234,210	**231,763**
P21	100	10	70	243,590	292,181	254,087	245,123	246,708	248,331	245,813	**243,590**
P22	100	10	150	203,988	246,109	206,555	205,052	204,435	208,508	205,312	**203,988**
P23	100	10	70	250,882	257,932	270,826	253,669	258,656	264,547	**250,882**	**250,882**
P24	100	10	150	204,317	209,339	216,173	204,815	205,883	211,925	205,009	**204,317**
P25	200	10	70	477,248	576,134	490,820	483,497	481,676	–	481,002	**477,248**
P26	200	10	150	378,351	402,097	416,753	380,044	380,613	–	383,586	**378,351**
P27	200	10	70	449,849	551,914	512,679	451,840	453,353	–	450,848	**449,849**
P28	200	10	150	374,330	388,975	379,980	375,019	377,351	–	376,674	**374,330**
P29	200	10	70	472,472	555,006	496,694	478,132	476,684	–	473,875	**472,472**
P30	200	10	150	362,817	449,288	389,016	364,834	365,250	–	363,701	**362,817**

BKS: solutions obtained either by the algorithms in their published version or during their parameter analysis phase.
SALRP*: best solution obtained during the parameter analysis phase of the proposed approach.
–: the problem is not solved in the corresponding study.
Bold numbers indicate that best known solution values are attained by the corresponding approach.

after 283 temperature reductions. Thus, all the experiments were terminated after 283 iterations, or when X_{best} is not improved in 100 successive reductions in temperature.

4.3. Computational results

In this section, we demonstrate the effectiveness of our SALRP heuristic by testing it on the three benchmark datasets, namely the Barreto's dataset, the Prins et al.'s dataset, and the Tuzun and Burke's dataset.

4.3.1. First dataset – Barreto's dataset

Table 3 presents the information of the first problem set and a comparison of the solutions obtained by the proposed SALRP heuristic and other algorithms from the literature. Columns 2–5 show the number of customers (n), the number of candidate sites (m), vehicle capacity (Q), and the best known solutions (BKS) that are reported in the literature or obtained in this study. The solutions obtained by the clustering based heuristic (CH) (Barreto, Ferreira, Paixão, & Santos, 2007), SA-ACS (Bouhafs et al., 2006), GRASP (Prins et al., 2006b), MA|PM (Prins et al., 2006a), LRGTS (Prins et al., 2007), and GAHLS (Duhamel et al., 2008) are shown in columns 6–11. Column 12 shows the solutions obtained from the proposed SALRP. The best solutions obtained during the parameter analysis phase are included in the last column (SALRP*).

Compared with the previous best known solutions, using the parameters in the final analysis, the SALRP produces the best solutions to 18 problems out of the 19 problems in this dataset; 8 of them are new best solutions. During the parameter analysis phase, the proposed SALRP yields the best solutions to all 19 problems, while 9 of them are new best solutions.

Further comparison of the performance of various algorithms for this dataset is presented in Table 4. It can be seen that the proposed algorithm outperforms all other algorithms in terms of solution quality. The average relative percentage gap with the best known solution is 0.40% using the parameters determined by the parameter analysis. For each algorithm, the relative percentage gaps are obtained by dividing the difference between the solution values obtained by the algorithm by the best known solution values.

The computational speed may depend on various factors, such as the CPU of the machines, the operation system, the compiler, the computer program, and the precision used during the execution of the run. Therefore, it is not an easy task to establish a fair comparison of the efficiency of various algorithms. In general, our SALRP heuristic takes about 6.8 s for small-scaled problem (with 12 customers and 2 potential depots), and 3352.5 s for large-scaled problem (with 318 customers and four potential depots), respectively. So the computational times seems not a critical issue for the proposed heuristic.

4.3.2. Second dataset – Prins et al.'s dataset

The experiment results for the second dataset are reported in Table 5. The solutions obtained by MSLS (Prins et al., 2004), GRASP (Prins et al., 2006b), MA|PM (Prins et al., 2006a), LRGTS (Prins et al., 2007), and GAHLS (Duhamel et al., 2008) are presented in columns 6–10.

Compared with the previous best known solutions, among the 30 instances in this dataset, our SALRP heuristic obtains 16 best solutions, including seven new best solutions. Moreover, during the parameter analysis, the proposed SALRP heuristic yields best solutions to all the 30 test problems; 18 of them are new best

Table 6
Further computational results for the second problem set.

Prob. ID	MSLS		GRASP		MA|PM		LRGTS		GAHLS		SALRP		SALRP*
	Gap	CPU	Gap	CPU	Gap	CPU	Gap	CPU	Gap	CPU	Gap	CPU	Gap
P1	1.85	N/A	0.42	0.2	0.00	0.3	0.62	0.4	0.00	0	0.00	19.8	0.00
P2	0.00	N/A	0.00	0.2	0.00	0.3	0.00	0.2	0.00	0	0.00	15	0.00
P3	1.55	N/A	0.00	0.1	0.00	0.4	0.00	0.5	0.00	0	0.00	19.3	0.00
P4	0.00	N/A	0.00	0.2	0.00	0.3	0.00	0.1	0.00	0	0.00	15	0.00
P5	8.84	N/A	0.58	1.8	0.05	2.6	0.05	0.3	0.00	6	0.00	74.7	0.00
P6	14.10	N/A	2.40	1.8	0.00	3.2	0.02	1	0.36	58	0.00	57.7	0.00
P7	2.14	N/A	0.55	2.4	0.00	3.4	0.47	1.8	0.47	35	0.00	95	0.00
P8	2.03	N/A	1.09	2.5	0.87	2.9	0.58	1.8	0.07	65	0.05	58.6	0.00
P9	–	–	0.00	1.7	0.00	3.2	0.15	2	0.42	28	0.00	74.7	0.00
P10	–	–	0.46	2.6	0.00	4.2	0.33	0.9	0.15	27	0.00	66.1	0.00
P11	12.24	N/A	1.37	2.3	0.00	3.1	0.00	0.3	0.00	39	0.29	74	0.00
P12	0.02	N/A	0.10	2	0.00	4.9	0.00	0.5	1.51	17	1.41	58.2	0.00
P13	3.34	N/A	1.46	27.6	2.37	26.3	0.91	8.7	2.23	220	0.59	348.6	0.00
P14	2.08	N/A	1.19	23.2	1.42	34.5	0.59	8.3	2.55	226	1.12	268.9	0.00
P15	2.26	N/A	3.02	17.4	0.98	35.8	1.48	2.3	1.80	126	0.23	348.6	0.00
P16	1.69	N/A	1.53	22.4	0.11	36.4	0.41	3.3	1.57	342	0.00	211.5	0.00
P17	2.41	N/A	1.96	21.6	0.83	28.7	0.94	2.4	1.82	188	0.08	250.3	0.00
P18	2.25	N/A	1.41	20.3	0.58	33.3	1.49	2.9	1.29	291	0.02	196.7	0.00
P19	12.31	N/A	12.22	37.4	9.93	24.7	1.36	14.1	12.98	401	1.06	270	0.00
P20	19.81	N/A	17.14	29.5	16.61	36	1.63	14	18.39	655	1.06	202.6	0.00
P21	19.95	N/A	4.31	39.1	0.63	24.6	1.28	14.4	1.95	306	0.91	260.6	0.00
P22	20.65	N/A	1.26	29.8	0.52	31.6	0.22	10.1	2.22	801	0.65	199.3	0.00
P23	2.81	N/A	7.95	35.4	1.11	29	3.10	13.3	5.45	176	0.00	338.1	0.00
P24	2.46	N/A	5.80	39.8	0.24	36.5	0.77	10.8	3.72	359	0.34	240.3	0.00
P25	20.72	N/A	2.84	517.5	1.31	345.1	0.93	62	–	–	0.79	1428.1	0.00
P26	6.28	N/A	10.15	379.1	0.45	463	0.60	60.3	–	–	1.38	1335.8	0.00
P27	22.69	N/A	13.97	554.3	0.44	280.6	0.78	60.3	–	–	0.22	1795.8	0.00
P28	3.91	N/A	1.51	367.4	0.18	321	0.81	76.9	–	–	0.63	1245.1	0.00
P29	17.47	N/A	5.13	424.8	1.20	212.9	0.89	77.2	–	–	0.30	1776	0.00
P30	23.83	N/A	7.22	290.2	0.56	272	0.67	73.3	–	–	0.24	1326.4	0.00
Avg.	8.20	N/A	3.57	96.5	1.35	76.7	0.70	17.5	2.46	–	0.38	422.4	0.00

GAP: relative percentage gap calculated as (solution values obtained by individual algorithm – BKS)/BKS.
–: the problem is not solved in the corresponding study.
N/A: the data is not provided in the corresponding study.

solutions. Our algorithm yields more best solutions than any one of the other algorithms in the comparison. It is worth mentioning that the proposed SALRP heuristic not only obtains the best solutions for many small-scaled instances with 20 or 50 customers, but also outperforms all other algorithms by finding 17 new best solutions out of 18 instances for large-scaled instances with 100 or 200 clients. Therefore, the SALRP heuristic can effectively solve large LRP instances involving as many as 10 candidate facilities and 200 customers.

Further comparison of the performance of various algorithms for this dataset is shown in Table 6. It can be seen that the proposed SALRP heuristic is competitive with other algorithms in terms of solution quality by providing the lowest average percentage gap. The gaps between the solutions obtained by the SALRP in a single run and the best known solutions ranges from 0.00% to 1.41%; all are well below 1.5%. Moreover, the difference between the smallest gap and the largest gap is the smallest among all algorithms considered in the comparison. This indicates the robustness of the proposed SALRP heuristic.

4.3.3. Third dataset – Tuzun and Burke's dataset

The solutions obtained by the proposed SALRP and problem characteristics for the third dataset are shown in Table 7. The solutions obtained by the two-phase tabu search (Tuzun & Burke, 1999), GRASP (Prins et al., 2006b), MA|PM (Prins et al., 2006a), LRGTS (Prins et al., 2007), and GAHLS (Duhamel et al., 2008) are shown in columns 6–10.

Compared with the previous best known solutions, using the parameters in the final analysis, the SALRP obtains seven best solutions, including five new best solutions. Furthermore, during the parameter analysis, the algorithm finds 29 best solutions out of the 36 instances in this dataset; 25 of them are new best solutions.

Further comparison of the performance of various algorithms for the third dataset is presented in Table 8. One can see that our algorithm provides the average percentage gap of 1.10% to the best known solutions, which is the smallest among all algorithms compared. This again, indicates the effectiveness of our SALRP heuristic.

In summary, we solved 85 LRP benchmark instances taken from three well-known LRP benchmark problem sets to test the performance of the proposed SALRP. Our algorithm obtained 41 best solutions including 20 new best solutions in a single run using the parameters determined by a parameter analysis. Moreover, during the parameter analysis, 78 best solutions are attained, including 52 new best solutions. The comparative results of the performance of our algorithm with other algorithms in the literature indicate that the proposed SALRP heuristic is capable of effectively solving diverse LRP instances within a reasonable amount of time. Although the required CPU time is longer than other approaches, it remains reasonable for such a strategic problem that does not need to be solved every day. Besides, implementing a new logistic network that will be operated for years is worth spending a few more minutes of computer time (Prins et al., 2006b), as long as the solution is competitive.

Table 7
Computational results for the third problem set.

Prob. ID	*n*	*m*	Q	BKS	2-Phase TS	GRASP	MA|PM	LRGTS	GAHLS	SALRP	SALRP*
T1	100	10	150	1467.68	1556.64	1525.25	1493.92	1490.82	1487.35	1477.24	**1467.68**
T2	100	20	150	1449.2	1531.88	1526.9	1471.36	1471.76	1483.48	1470.96	**1449.2**
T3	100	10	150	1394.8	1443.43	1423.54	1418.83	1412.04	1444.7	1408.65	**1394.8**
T4	100	20	150	1432.29	1511.39	1482.29	1492.46	1443.06	1466.92	**1432.29**	**1432.29**
T5	100	10	150	1167.16	1231.11	1200.24	1173.22	1187.63	1185.45	1177.14	**1167.16**
T6	100	20	150	1102.24	1132.02	1123.64	1115.37	1115.95	1115.49	1110.36	**1102.24**
T7	100	10	150	791.66	825.12	814	793.97	813.28	807.85	**791.66**	**791.66**
T8	100	20	150	728.3	740.54	747.84	730.51	742.96	737.19	731.95	**728.3**
T9	100	10	150	1238.49	1316.98	1273.1	1262.32	1267.93	1251.01	**1238.49**	**1238.49**
T10	100	20	150	1245.31	1274.5	1272.94	1251.32	1256.12	1260.1	1247.28	**1245.31**
T11	100	10	150	902.26	920.75	912.19	903.82	913.06	909.98	**902.26**	**902.26**
T12	100	20	150	1018.29	1042.21	1022.51	1022.93	1025.51	1036.86	1024.02	**1018.29**
T13	150	10	150	1866.75	2000.97	2006.7	1959.39	1946.01	–	1953.85	1922.59
T14	150	20	150	1833.95	1892.84	1888.9	1881.67	1875.79	–	1899.05	**1833.95**
T15	150	10	150	1978.27	2022.11	2033.93	1984.25	2010.53	–	2057.53	**1978.27**
T16	150	20	150	1801.39	1854.97	1856.07	1855.25	1819.89	–	**1801.39**	**1801.39**
T17	150	10	150	1447.66	1555.82	1508.33	1448.27	1448.65	–	1453.3	**1447.66**
T18	150	20	150	1441.98	1478.8	1456.82	1459.83	1492.86	–	1455.5	**1441.98**
T19	150	10	150	1205.09	1231.34	1240.4	1207.41	1211.07	–	1206.24	**1205.09**
T20	150	20	150	930.99	948.28	940.8	934.79	936.93	–	934.62	**930.99**
T21	150	10	150	1699.92	1762.45	1736.9	1720.3	1729.31	–	1720.81	1704.58
T22	150	20	150	1400.01	1488.34	1425.74	1429.34	1424.59	–	1415.85	**1400.01**
T23	150	10	150	1199.51	1264.63	1223.7	1203.44	1216.32	–	1216.84	1201.23
T24	150	20	150	1152.18	1182.28	1231.33	1158.54	1162.16	–	1159.12	**1152.18**
T25	200	10	150	2279.92	2379.47	2384.01	2293.99	2296.52	–	2324.1	**2279.92**
T26	200	20	150	2185.41	2211.74	2288.09	2277.39	2207.5	–	2258.16	**2185.41**
T27	200	10	150	2234.78	2288.17	2273.19	2274.57	2260.87	–	2260.3	2245.33
T28	200	20	150	2241.04	2355.81	2345.1	2376.25	2259.52	–	2326.53	**2241.04**
T29	200	10	150	2089.77	2158.6	2137.08	2106.26	2120.76	–	2112.65	**2089.77**
T30	200	20	150	1709.56	1787.02	1807.29	1771.53	1737.81	–	1722.99	1719.96
T31	200	10	150	1466.62	1549.79	1496.75	1467.54	1488.55	–	1469.1	**1466.62**
T32	200	20	150	1084.78	1112.96	1095.92	1088	1090.59	–	1088.64	1087.05
T33	200	10	150	1970.44	2056.11	2044.66	1973.28	1984.06	–	1994.16	**1970.44**
T34	200	20	150	1918.93	2002.42	2090.95	1979.05	1986.49	–	1932.05	**1918.93**
T35	200	10	150	1771.06	1877.3	1788.7	1782.23	1786.79	–	1779.1	1776.56
T36	200	20	150	1393.16	1414.83	1408.63	1396.24	1401.16	–	1396.42	**1393.16**

BKS: solutions obtained either by the algorithms in their published version or during their parameter analysis phase.
SALRP*: best solution obtained during the parameter analysis phase of the proposed approach.
–: the problem is not solved in the corresponding study.
Bold numbers indicate that best known solution values are attained by the corresponding approach.

Table 8
Further computational results for the third problem set.

Prob. ID	2-Phase TS		GRASP		MA|PM		LRGTS		GAHLS		SALRP		SALRP*
	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)	CPU (s)	Gap (%)
T1	6.06	5	3.92	32.4	1.79	31.5	1.58	3.3	1.34	628	0.65	369.2	0.00
T2	5.71	3	5.36	40.7	1.53	35.6	1.56	6.5	2.37	922	1.50	273.5	0.00
T3	3.49	3	2.06	27.6	1.72	36.2	1.24	4.2	3.58	507	0.99	230.9	0.00
T4	5.52	4	3.49	36.2	4.20	36.4	0.75	7.4	2.42	1194	0.00	420.4	0.00
T5	5.48	4	2.83	27.7	0.52	31.9	1.75	6.9	1.57	333	0.86	348.1	0.00
T6	2.70	2	1.94	34.3	1.19	42.7	1.24	6.8	1.20	1381	0.74	342.3	0.00
T7	4.23	3	2.82	22.5	0.29	38	2.73	5.2	2.05	557	0.00	359.6	0.00
T8	1.68	3	2.68	37.3	0.30	49.3	2.01	5.9	1.22	959	0.50	418.1	0.00
T9	6.34	3	2.79	21.5	1.92	36.8	2.38	4.3	1.01	877	0.00	300.3	0.00
T10	2.34	4	2.22	36	0.48	47.7	0.87	6.3	1.19	1142	0.16	427.8	0.00
T11	2.05	4	1.10	20.3	0.17	35.1	1.20	4	0.86	465	0.00	290.5	0.00
T12	2.35	3	0.41	38.4	0.46	62.6	0.71	4.9	1.82	1009	0.56	315.6	0.00
T13	7.19	12	7.50	113	4.96	128.5	4.25	12.5	–	–	4.67	743.3	2.99
T14	3.21	12	3.00	161.4	2.60	144.2	2.28	18.5	–	–	3.55	835.1	0.00
T15	2.22	14	2.81	100	0.30	111	1.63	11.1	–	–	4.01	456.3	0.00
T16	2.97	13	3.04	132.4	2.99	144.1	1.03	15.8	–	–	0.00	833	0.00
T17	7.47	9	4.19	117.7	0.04	166.6	0.07	22	–	–	0.39	749.7	0.00
T18	2.55	12	1.03	166.1	1.24	154.8	3.53	28	–	–	0.94	827.7	0.00
T19	2.18	9	2.93	106.7	0.19	161.4	0.50	14.6	–	–	0.10	752.4	0.00
T20	1.86	9	1.05	142.4	0.41	196.1	0.64	13.7	–	–	0.39	841.8	0.00
T21	3.68	9	2.18	92.8	1.20	143.8	1.73	17.9	–	–	1.23	741.7	0.27
T22	6.31	9	1.84	128.4	2.09	155.7	1.76	18.5	–	–	1.13	832.5	0.00
T23	5.43	10	2.02	88.5	0.33	153.8	1.40	14.5	–	–	1.44	755.6	0.14
T24	2.61	9	6.87	134.9	0.55	223	0.87	14.3	–	–	0.60	837	0.00
T25	4.37	22	4.57	308	0.62	418.3	0.73	32.6	–	–	1.94	1327.9	0.00
T26	1.20	22	4.70	410	4.21	458.4	1.01	39.6	–	–	3.33	1454.6	0.00
T27	2.39	23	1.72	311.4	1.78	376.8	1.17	32.8	–	–	1.14	1319.4	0.47
T28	5.12	26	4.64	418.9	6.03	436.3	0.82	40.2	–	–	3.81	1427.6	0.00
T29	3.29	20	2.26	338	0.79	350.5	1.48	47.2	–	–	1.09	1320.3	0.00
T30	4.53	18	5.72	370	3.62	377	1.65	59.3	–	–	0.79	1400.3	0.61
T31	5.67	18	2.05	242.7	0.06	322.2	1.50	36.7	–	–	0.17	1299.2	0.00
T32	2.60	18	1.03	308.5	0.30	505	0.54	38.7	–	–	0.36	1429	0.21
T33	4.35	23	3.77	282.8	0.14	412.8	0.69	41.6	–	–	1.20	1317.5	0.00
T34	4.35	20	8.96	399.2	3.13	406.1	3.52	51.8	–	–	0.68	1411.7	0.00
T35	6.00	20	1.00	199	0.63	352.8	0.89	34	–	–	0.45	1313.8	0.31
T36	1.56	17	1.11	296.3	0.22	529.8	0.57	43.2	–	–	0.23	1427.4	0.00
Avg.	3.92	11.5	3.10	195.6	1.47	203.1	1.45	21.2	1.72	–	1.10	826.4	0.14

GAP: relative percentage gap calculated as (solution values obtained by individual algorithm − BKS)/BKS.
–: the problem is not solved in the corresponding study.
N/A: the data is not provided in the corresponding study.

5. Conclusions and outlook for future research

This paper introduces a new SA based solution approach for the LRP. Most of the early studies on LRP have been limited to a few conventional heuristics, and exact solution techniques which are limited to small size problems. Recently, a number of studies applied meta-heuristics to tackle the problem and obtained good results. Therefore, we present an SA based heuristic with a special solution encoding scheme that integrates location and routing decisions in order to enlarge the search space so that better solutions can be found.

The proposed SALRP algorithm was tested on three well-known benchmark problem sets that contain a wide variety of LRP instances. Computational results and comparisons of the SALRP algorithm with various promising LRP heuristics are presented. The results of the comparative study are very encouraging: the proposed SALRP algorithm not only performs better than the other heuristics on many instances, but its computational time requirement is also quite reasonable for realistic size problems.

Our computation times are longer than those of other approaches included in the comparison which may be due to the more complicated decoding process and that our implementation is only a basic one that does not use any intricate, problem specific speed-ups. Nevertheless, the main intention of this comparison is to show that, with respect to solution quality, our approach can produce results comparable to those of the best performing methods for the LRP from literature. Also, for a strategic problem that does not need to be solved daily, spending a few more minutes of computer time to obtain good solutions may prove worthwhile in the long run.

The encouraging results suggest that the proposed SALRP algorithm may be applied to other combinatorial problems that contain multiple levels of decision making. In this application (LRP), there are two decision components (facility location and vehicle routing) where the decisions made for one component affect the other. Thus, an integrated approach similar to the proposed SALRP may be used for other combinatorial problem with more than one component of interdependent decisions, such as multi-level location problems, hub-and-spoke network design problems, and multimodal transportation problems.

Acknowledgement

The work of the first author is partially supported by the new faculty startup grant provided by the National Taiwan University of Science and Technology. The second and the third authors are grateful to the National Science Council of Taiwan and the Chang Gung University for financially supporting this research under contract numbers NSC 97-2410-H-182-020-MY2 and UARPD370101, respectively.

References

Aksen, D., & Altinkemer, K. (2008). A location-routing problem for the conversion to the "click-and-mortar" retailing: The static case. *European Journal of Operational Research, 186*, 554–575.

Albareda-Sambola, M., Díaz, J. A., & Fernández, E. (2005). A compact model and tight bounds for a combined location-routing problem. *Computers & Operations Research, 32*(3), 407–428.

Barreto, S. S. (2004). Análise e Modelização de Problemas de localização-distribuição [Analysis and modelling of location-routing problems]. Ph.D. Thesis, University of Aveiro, Aveiro, Portugal.

Barreto, S. S., Ferreira, C., Paixão, J., & Santos, B. S. (2007). Using clustering analysis in capacitated location-routing problem. *European Journal of Operational Research, 179*(3), 968–977.

Bednar, L., & Strohmeier, E. (1979). Lagerstandortoptimierung und Fuhrparkeinsatzplanung in der Konzumgüter-Industrie. *Zeitschrift für Operations Research, 23*, B89–B104.

Berman, O., Jaillet, P., & Simchi-Levi, D. (1995). Location-routing problems with uncertainty. In Z. Drezner (Ed.), *Facility location: A survey of applications and methods* (pp. 427–452). New York: Springer.

Bookbinder, J. H., & Reece, K. E. (1988). Vehicle routing considerations in distribution system design. *European Journal of Operational Research, 37*, 204–213.

Bouhafs, L., Hajjam, A., & Koukam, A. (2006). A combination of simulated annealing and ant colony system for the capacitated location-routing problem. *Lecture Notes in Computer Science, 4251*, 409–416.

Bruns, A., Klose, A., & Stähly, P. (2000). Restructuring of Swiss parcel delivery services. *OR Spektrum, 22*, 285–302.

Chan, Y., Carter, W. B., & Burnes, M. D. (2001). A multiple-depot, multiple-vehicle, location-routing problem with stochastically processed demands. *Computers & Operations Research, 28*, 803–826.

Chien, T. W. (1993). Heuristic procedures for practical-sized uncapacitated location-capacitated routing problems. *Decision Sciences, 24*(5), 995–1021.

Cornuéjols, G., Fisher, M. L., & Wolsey, L. A. (1977). Location of bank accounts to optimize float: an analytic study of exact and approximate algorithms. *Management Science, 23*, 789–810.

Duhamel, C., Lacomme, P., Prins, C., & Prodhon, C. (2008). A memetic approach for the capacitated location routing problem. In Proceedings of the EU/meeting 2008 workshop on metaheuristics for logistics and vehicle routing. University of Technology of Troyes, France.

Ghiani, G., & Laporte, G. (2001). Location-arc routing problems. *OPSEARCH, 38*, 151–159.

Jacobsen, S. K., & Madsen, O. B. G. (1980). A comparative study of heuristics for a two-level routing-location problem. *European Journal of Operational Research, 5*, 378–387.

Karp, R. (1972). Reducibility among combinatorial problems. In R. Miller & J. Thatcher (Eds.), *Complexity of computer computations* (pp. 85–104). New York: Plenum Press.

Kirkpatrick, S., Gelatt, C. D., Jr., & Vecchi, M. P. (1983). Optimization by simulated annealing. *Science, 220*, 671–680.

Kulcar, T. (1996). Optimizing solid waste collection in Brussels. *European Journal of Operational Research, 90*, 26–44.

Labadi, N. (2003). Problèmes tactiques et stratégiques en tournées sur arcs [tactical and strategic problems on arc routing]. Ph.D. Dissertation, University of Technology of Troyes, France.

Laporte, G. (1988). Location routing problems. In: B. L. Golden & A. A. Assad, (Eds.), Vehicle routing: Methods and studies. Amsterdam, North-Holland.

Laporte, G., Louveaux, F., & Mercure, H. (1989). Models and exact solutions for a class of stochastic location-routing problems. *European Journal of Operational Research, 39*, 71–78.

Laporte, G., & Nobert, Y. (1981). An exact algorithm for minimizing routing and operating costs in depot location. *European Journal of Operational Research, 6*(2), 224–226.

Laporte, G., Nobert, Y., & Arpin, D. (1986). An exact algorithm for solving a capacitated location-routing problem. *Annals of Operations Research, 6*(9), 293–310.

Laporte, G., Nobert, Y., & Pelletier, P. (1983). Hamiltonian location problems. *European Journal of Operational Research, 12*, 80–87.

Laporte, G., Nobert, Y., & Taillefer, S. (1988). Solving a family of multi-depot vehicle routing and location-routing problems. *Transportation Science, 22*(3), 161–172.

Lenstra, J. K., & Rinnooy Kan, A. H. G. (1981). Complexity of vehicle routing and scheduling problems. *Networks, 11*, 221–227.

Lin, C. K. Y., Chow, C. K., & Chen, A. (2002). A location-routing-loading problem for bill delivery services. *Computers & Industrial Engineering, 43*(1–2), 5–25.

Lin, S., & Kernighan, B. W. (1973). An effective heuristic algorithm for the traveling salesman problem. *Operations Research, 21*, 498–516.

Madsen, O. B. G. (1983). Methods for solving combined two level location routing problems of realistic dimensions. *European Journal of Operational Research, 12*, 295–301.

Marinakis, Y., & Marinaki, M. (2008). A particle swarm optimization algorithm with path relinking for the location routing problem. *Journal of Mathematical modeling and algorithms, 7*, 59–78.

Metropolis, N., Rosenbluth, A. W., Rosenbluth, M. N., Teller, A. H., & Teller, E. (1953). Equations of state calculations by fast computing machines. *Journal of Chemical Physics, 21*, 1087–1092.

Min, H., Jayaraman, V., & Srivastava, R. (1998). Combined location-routing problems: A synthesis and future research directions. *European Journal of Operational Research, 108*(1), 1–15.

Murty, K. G., & Djang, P. A. (1999). The US army national guard's mobile training simulators location and routing problem. *Operations Research, 47*, 175–182.

Nagy, G., & Salhi, S. (2007). Location-routing: Issues, models and methods. *European Journal of Operational Research, 177*, 649–672.

Prins, C., Prodhon, C., Ruiz, A., Soriano, P., & Wolfler Calvo, R. (2007). Solving the capacitated location-routing problem by a cooperative Lagrangean relaxation-granular tabu search heuristic. *Transportation Science, 41*(4), 470–483.

Prins, C., Prodhon, C., & Wolfler Calvo, R. (2004). Nouveaux algorithmes pour le problème de localisation et routage sous contraintes de capacité. In Proceedings of the MOSIM' 04 (Vol. 2, pp. 1115–1122). Lavoisier, Ecole des Mines de Nantes, France.

Prins, C., Prodhon, C., & Wolfler Calvo, R. (2006a). A memetic algorithm with population management (MA|PM) for the capacitated location-routing problem. *Lecture Notes in Computer Science, 3906*, 183–194.

Prins, C., Prodhon, C., & Wolfler Calvo, R. (2006b). Solving the capacitated location-routing problem by a GRASP complemented by a learning process and a path relinking. *4OR, 4*(3), 221–238.

Salhi, S., & Rand, G. K. (1989). The effect of ignoring routes when locating depots. *European Journal of Operational Research, 45*, 150–156.

Srivastava, R. (1993). Alternate solution procedures for the locating-routing problem. *OMEGA, 21*(4), 497–506.

Su, C. T. (1999). Dynamic vehicle control and scheduling of a multi-depot physical distribution system. *Integrated Manufacturing Systems, 10*, 56–65.

Tuzun, D., & Burke, L. I. (1999). A two-phase tabu search approach to the location routing problem. *European Journal of Operational Research, 116*(1), 87–99.

Wasner, M., & Zäpfel, G. (2004). An integrated multi-depot hub location vehicle routing model for network planning of parcel service. *International Journal of Production Economics, 90*, 403–419.

Watson-Gandy, C. D. T., & Dohrn, P. J. (1973). Depot location with van salesmen – a practical approach. *OMEGA, 1*, 321–329.

Wu, T. H., Low, C., & Bai, J. W. (2002). Heuristic solutions to multi-depot location-routing problems. *Computers & Operations Research, 29*(10), 1393–1415.