PERGAMON

Computers & Operations Research 29 (2002) 1393–1415

computers & operations research

www.elsevier.com/locate/dsw

Heuristic solutions to multi-depot location-routing problems

Tai-Hsi Wu*, Chinyao Low, Jiunn-Wei Bai

Department of Industrial Engineering, Da-Yeh University, Changhua, Taiwan

Received 1 November 1999; received in revised form 2 February 2001

Abstract

This paper presents a method for solving the multi-depot location-routing problem (MDLRP). Since several unrealistic assumptions, such as homogeneous fleet type and unlimited number of available vehicles, are typically made concerning this problem, a mathematical formulation is given in which these assumptions are relaxed. Since the inherent complexity of the LRP problem makes it impossible to solve the problem on a larger scale, the original problem is divided into two sub-problems, i.e., the location-allocation problem, and the general vehicle routing problem, respectively. Each sub-problem is then solved in a sequential and iterative manner by the simulated annealing algorithm embedded in the general framework for the problem-solving procedure. Test problems from the literature and newly created problems are used to test the proposed method. The results indicate that this method performs well in terms of the solution quality and run time consumed. In addition, the setting of parameters throughout the solution procedure for obtaining quick and favorable solutions is also suggested.

Scope and purpose

In many logistic environments managers must make decisions such as location for distribution centers (DC), allocation of customers to each service area, and transportation plans connecting customers. The location-routing problems (LRPs) are, hence, defined to find the optimal number and locations of the DCs, simultaneously with the vehicle schedules and distribution routes so as to minimize the total system costs. This paper proposes a decomposition-based method for solving the LRP with multiple depots, multiple fleet types, and limited number of vehicles for each different vehicle type. The solution procedure developed is very straightforward conceptually, and the results obtained are comparable with other heuristic methods. In addition, the setting of parameters throughout the solution procedure for obtaining quick and favorable solutions is also suggested.

Keywords: Location-routing; Simulated annealing; Location-allocation; Vehicle routing

* Corresponding author. Tel.: + 886-4-8528469 x 2230; fax: + 886-4-8520781.
E-mail address: taiwu@mail.dyu.edu.tw (T.-H. Wu).

PII: S0305-0548(01)00038-7

1. Introduction

In many logistic environments managers must make decisions such as (1) location for factories/warehouses/distribution centers (DC), referred to as depots, (2) allocation of customers to each service area, and (3) transportation plans connecting customers, raw materials, plants, warehouses, and channel members. These decisions are important in the sense that they greatly affect the level of service for customers and the total logistic system cost. For determining the depot locations, many mathematical models have been developed to solve the problems. However, in these models the transportation costs between depot and customers were incorrectly assumed and performed on a straight-and-back basis (the moment sum equation [1]). The fact that several customers can be served on a single route, provided that the total demand does not exceed the tour capacity, reveals the necessity for unifying the location and routing problems. The interdependence of these two problems was not recognized until the 1970s [2]. The location-routing problems (LRPs) can, hence, be defined as vehicle routing problems (VRPs) in which the optimal number and locations of the depots are to be determined simultaneously with the vehicle schedules and distribution routes so as to minimize the total system costs. Recent surveys of LRPs can be found in Laporte [3]. Research on LRPs is quite limited compared with the extensive literature on pure location problems, VRPs and their variants. Since VRPs have long been recognized as NP-hard problems due to the embedded traveling salesman problem, not to mention the difficulty of the more complicated LRPs, it was considered impractical to incorporate the VRPs into the location problems until the late 1970s. In addition to the problem complexity, some sensitivity analyses such as the changes in vehicle capacity, depot capacity, and number of potential depot locations, bring valuable information to decision makers and, thus, require repetitive problem solving. It is, hence, not surprising that approximation methods have been much more widely used than exact methods for solving the LRPs. Several studies, for instance, Golden et al. [4], Or and Pierskalla [5], Jacobsen and Madsen [6], Srikar and Srivastava [7], Perl [8], Perl and Daskin [2], have proposed formulations and algorithms for the general LRPs and LPRs under some side constraints such as capacity limit and maximum cost/tour-length restriction. In terms of the procedure for solving the LRP, it can be viewed as a compilation of the following three sub-problems: (1) facility location (2) demand allocation and (3) vehicle routing. Separate optimization of these sub-problems always leads to a non-optimal decision. An incorporation of all the sub-problems together, however, is computationally impractical. For solving the LRP more efficiently, two types of sequential methods are commonly used: (1) location-allocation-routing (LAR), for example, [5,6] and (2) allocation-routing-location (ARL), [6,7]. Other than heuristic algorithms, Laporte et al. made a series of significant contributions in the presentation of exact methods [3,9–12]. In Laporte [11] a method was presented which transforms the multi-depot VRP (MDVRP) and LRP into an equivalent constrained assignment problem through the use of graphical representation. The problem was then solved by the branch-and-bound method. The LRP with 80 nodes was solvable within a reasonable amount of time. In Laporte et al. [12] a stochastic LRP model was introduced in which the demands are known only when vehicles reach the customers. In the first stage LRP decisions are still made without the actual demand information. It is not surprising that the restriction of route capacities might be violated. If that happens, corrective recourse action is taken accompanied by a penalty in the second stage. The problem is formulated and solved to optimality, but only solutions for problems with up to 30 nodes are provided. Due to the inherent characteristics of the optimization approach, the number of depots to be established and the number of customer nodes in the exact models must be very restricted. Thus these exact formulations can be applied to only very small-sized problems. Perl and Daskin [2] first formulated the warehouse location-routing problem (WLRP). After making some modifications to the WLRP, they presented a solution method which solved the MWLRP (modified WLRP) in a sequential manner. Hansen et al. [13] further proposed a more efficient method for the MWLRP. Srivastava and Benton [14] investigated several environmental factors that might influence the design of a distribution system. Chien [15] proposed an approximate approach for the LRP, in which two route length estimators are used in calculating the routing cost. Salhi and Fraser [16] proposed an iterative method that alternates between the location phase and routing phase until a suitable stopping criterion is met. In that study a more practical version of the LRP is addressed, where the vehicles may have different capacities. Nagy and Salhi [17,18] adopted the concept of nested methods to treat the routing element as a sub-problem within the larger problem of location when solving the LRP. Min et al. [19] synthesized the past research and suggested some future research directions for the LRP.

Though the close relationship between location problems and the VRP and the correct way for calculating depot–customer distance have been recognized and presented by both academics and practitioners [2], research in the past seems to have ignored the fact that most companies own delivery fleets of different capacities. To our best knowledge, the study by Salhi and Fraser [16] seems to be the only one in which the vehicles involved in the LRP do not necessarily have the same capacities. But as typically assumed in the literature, the number of vehicles for each type is unlimited in their study, an assumption that does not agree with the real situation. In this paper, we address an extended and more practical version of the LRP, i.e., an LRP considering multiple depots, multiple fleet types, and limited number of vehicles for each different vehicle type. This setting allows one to reflect reality more accurately and, in addition, analyze the effect of fleet type and size. The solution of problems under the assumption of no restrictions on fleet type and size can be obtained by setting the fleet type to one, and the fleet size to a very large number.

This paper proposes a heuristic method, which decomposes the LRP into a location-allocation problem (LAP) and a vehicle routing problem (VRP). Both LAPs and VRPs are difficult to solve. Hence, heuristic methods for the LAP and VRP, respectively, were developed. Since simulated annealing (SA) has been applied to a number of combinatorial problems with fairly good results [20], it was selected as the basis for developing search methods for both the LAP and VRP. SA can be viewed as a process which attempts to move from the current solution to its neighborhood solutions resulting in better objective values. However, for solutions with worse objective values, they are accepted with a specified probability mainly to escape from the local optima in its search for the global optima.

This paper is organized as follows. Section 2 defines the multi-depot location-routing problem and gives a corresponding mathematical formulation. Section 3 presents a heuristic model in which the LRP model is decomposed into three sub-problems. The heuristic solution for each sub-problem is also provided in Section 3. Computational results of test problems from Perl [8] and simulated problems are presented and discussed in Section 4. Some managerial issues regarding integrating both location and routing problems are discussed in Section 5. Section 6 summarizes the conclusions of this study.

2. Multi-depot location-routing problem

The multi-depot location-routing problem is first defined. In a logistic system assume that the number, location, and demand of customers, the number, and location of all potential depots, as well as the fleet type and size are given. The distribution and routing plan must be designed so that:

(1) the demand of each customer can be satisfied,
(2) each customer is served by exactly one vehicle,
(3) the total demand on each route is less than or equal to the capacity of the vehicle assigned to that route, and
(4) each route begins and ends at the same depot.

The problem is to simultaneously determine the number, locations of depots, assignment of customers to depots, vehicle types to routes, and the corresponding delivery routes, so that the total costs consisting of depot-establishing cost, transportation cost, and dispatching cost for vehicles are minimized. The proposed mathematical programming formulation below differs slightly from Perl's MWLRP model [2] on three points:

1. Another set of sub-tour elimination constraints resulting in a much lower number of constraints is introduced;
2. The variable warehousing cost is not considered in the objective function while adding the dispatching cost for vehicles assigned;
3. In addition to the homogeneous fleet type, the flexibility of allowing the fleet to have different capacities in the problems is added.

For analyzing the close relationship between location and routing problems in the LRP, we consider only the depot establishing cost, transportation, and vehicles' dispatching costs in the objective function of the proposed model. Other costs such as the variable warehousing cost appearing in Perl's formulation, proportional to the quantity of throughput at each depot, are ignored in this study.

For reducing computational complexity, Perl and Daskin [2] assumed that the delivery fleet consists of standard vehicles and the fleet size is unlimited. It is, thus, unnecessary to consider the fixed cost of the vehicles in their model. However, the addition of the vehicles' fixed cost is important because it provides the capability for exploring the effect of a limited fleet size. The sets, parameters and variables used in the mathematical model are defined below:

Sets:
I set of all potential DC sites
J set of all customers
K set of all vehicles
Parameters:
N number of customers
C_{ij} distance between points i and j. $i, j \in I \cup J$
G_i fixed costs of establishing depot i

F_k fixed costs of using vehicle k
V_i maximum throughput at depot i
d_j demand of customer j
Q_k capacity of vehicle (or route) k
Decision variables:
x_{ijk} = 1, if point i immediately precedes point j on route k ($i,j \in I \cup J$); 0 otherwise.
y_i = 1, if depot i is established; 0 otherwise.
z_{ij} = 1, if customer j is allocated to depot i; 0 otherwise.
U_{lk} auxiliary variable for sub-tour elimination constraints in route k.
Mathematical model:

$$\mathrm{Min} \sum_{i \in I} G_i y_i + \sum_{i \in I \cup J} \sum_{j \in I \cup J} \sum_{k \in K} C_{ij} x_{ijk} + \sum_{k \in K} F_k \sum_{i \in I} \sum_{j \in J} x_{ijk}, \tag{1}$$

s.t.

$$\sum_{k \in K} \sum_{i \in I \cup J} x_{ijk} = 1, \quad j \in J, \tag{2}$$

$$\sum_{j \in J} d_j \sum_{i \in I \cup J} x_{ijk} \leqslant Q_k, \quad k \in K, \tag{3}$$

$$U_{lk} - U_{jk} + N x_{ljk} \leqslant N - 1, \quad l,j \in J,\ k \in K, \tag{4}$$

$$\sum_{j \in I \cup J} x_{ijk} - \sum_{j \in I \cup J} x_{jik} = 0, \quad k \in K,\ i \in I \cup J, \tag{5}$$

$$\sum_{i \in I} \sum_{j \in J} x_{ijk} \leqslant 1, \quad k \in K, \tag{6}$$

$$\sum_{j \in J} d_j z_{ij} - V_i y_i \leqslant 0, \quad i \in I, \tag{7}$$

$$-z_{ij} + \sum_{u \in I \cup J} (x_{iuk} + x_{ujk}) \leqslant 1, \quad i \in I,\ j \in J,\ k \in K, \tag{8}$$

$$x_{ijk} = 0,1, \quad i \in I,\ j \in J,\ k \in K, \tag{9}$$

$$y_i = 0,1, \quad i \in I, \tag{10}$$

$$z_{ij} = 0,1, \quad i \in I, \quad j \in J, \tag{11}$$

$$U_{lk} \geqslant 0, \quad l \in J, k \in K, \tag{12}$$

The objective function minimizes the sum of the fixed depot-establishing cost, delivery cost, and dispatching cost for the vehicles assigned, respectively. Eqs. (2) require that each customer be assigned to a single route. Eqs. (3) are the capacity constraint set for vehicles. Eqs. (4) are the new sub-tour elimination constraint set. Flow conservation constraints are expressed in (5). Constraints (6) assure that each route can be served at most once. Capacity constraints for the DCs are given in (7). Constraints (8) specify that a customer can be assigned to a DC only if there is a route from that DC going through that customer. Constraint sets (9), (10), and (11) are the binary requirements on the decision variables. The U_{lk}'s, auxiliary variables taking positive values are declared in (12).

Note that introducing a new set of sub-tour elimination constraints does result in a smaller number of constraints in the model. However, a larger number of auxiliary variables, U_{lk}'s is also added to the model. Christofides et al. [21] presented an approach in which only a set of auxiliary variables is used to eliminate sub-tours.

3. Heuristic solution to MDLRP/MWLRP

To solve the MDLRP on a large scale, a heuristic approach is proposed. The unrealistic assumption regarding fleet type is relaxed in this model. To demonstrate the solving efficiency by comparing the results of the proposed method to those of other works, two versions of the heuristic approaches are developed and discussed, with one dealing with the case of the homogeneous fleet type MDLRP, the other with the multiple-fleet type case.

3.1. Perl's decomposition method

For the MDLRP/MWLRP, Perl and Daskin [2] divided the original problem into three sub-problems: (1) the multi-depot vehicle-dispatch problem (MDVDP), (2) the warehouse location-allocation problem (WLAP), and (3) the multi-depot routing allocation problem (MDRAP), respectively. These sub-problems can be solved by either optimal techniques or heuristic algorithms in a sequential manner while taking into account the inter-relationship between sub-problems. Hansen et al. [13] later used the same fragmentational logic while presenting more efficient algorithms for solving the sub-problems. Since both the MDVDP and the MDRAP contain routing problems and the WLAP is a location-type problem, these sub-problems are difficult to solve to optimality. Solving the MDLRP/MWLRP in a sequential manner and developing good heuristic approaches are thus more appropriate than the optimum method in terms of solving efficiency.

The problem definition of each sub-problem in Ref. [2] and the inter-relationship are described next. In the MDVDP it is assumed that all potential DC sites are used while the warehousing cost and warehouse capacity restriction are ignored. The objective is to find an initial set of routes to minimize the total delivery cost. The set of routes obtained in the MDVDP is given as an input to the WLAP. Also given are the warehousing costs at each site and the stem distances between each route and each warehouse site. The problem here is to determine the number, size, and locations of the warehouses, and the allocation of routes to each to minimize the objective function consisting of warehousing and delivery costs. The locations of warehouses selected in the WLAP are then used as input for the third sub-problem, the MDRAP. The primary purpose of the MDRAP is to reallocate each customer to the given warehouses and simultaneously solve the multi-depot routing problem. The output of the MDRAP, a set of delivery routes, is then used as an input to the WLAP. This procedure is repeated iteratively until the convergence criterion is met.

3.2. Proposed decomposition method

Two assumptions made by Perl and Daskin [2] initiate the present heuristic approach, namely, (1) all potential DC sites are used in the MDVDP for constructing the initial set of feasible routes, and (2) the variable running-distance component of each route is fixed, whereby only the stem distance of each route is considered in the objective function when the WLAP is solved. The stem distance is defined as the sum of the distances from the DC to the first customer on the route and from the last customer back to the DC. The variable running distance denotes the distance from the first to the last customer. Assumption 1 opens every potential DC site as a starting feasible solution. Although this solution will surely be improved in later iterations, it actually consumes some unnecessary computation time. In Assumption 2, the first and last customer in the visiting sequence for each route are fixed and used to calculate the stem distances between routes and warehouses. However, there might exist possibilities that other pairs of starting and ending nodes would result in shorter stem distances.

In this paper a heuristic method which decomposes the MWLRP into the following two sub-problems is proposed:

Phase 1. Location-allocation problem (LAP) [22,23].

Phase 2. Vehicle Routing problem (VRP) [24].

These sub-problems can be solved by either optimization techniques or heuristic methods iteratively and at the same time, the dependence between each sub-problem can be considered. In each iteration, current routes are unified to include more customers and allocated to selected warehouses while taking into account the capacity restrictions of the warehouses. Since both LAP and VRP are well-known combinatorial optimization problems, heuristic methods that give quick and good solutions have been presented. See Refs. [25–29] for VRP, and [30–32,5] for LAP.

In the first iteration, the solutions of the LAP are some selected DC sites (only necessary ones, not all) and a plan for allocating customers to each chosen DC site. Notice that the calculation of the distance for the LAP in this iteration (the first) is based on the "moment-sum" equation used by the traditional location problems. These solutions are then used as input to the VRP to generate a starting feasible set of routes. At this moment it is very possible that the number of DCs established can still be reduced. To achieve this objective, each current route consisting of several customers is viewed as a single node with the demand represented by the sum of demands of all customers in that route. These "big nodes" are then consolidated for reducing the number of DCs established and, thus, the total cost. The consolidation process starts from the second iteration and is performed by the LAP module, followed by the VRP module. This procedure is repeated until the convergence criterion is met.

Perl and Daskin [2] fixed the first and last customer in the sequence of each route and calculated the stem distances between routes and warehouses. This possibly leads to a loss of opportunities for exploiting better combinations of beginning and ending customers. Therefore, a new distance matrix (saving matrix) for solving the LAPs starting from the second iteration is defined. The idea is that "savings" can be realized if the demand of the "big node" is allocated from the current DC to other DCs with less distance resulting. These savings are calculated by evaluating any two adjacent customers on the route, which are candidates for being a pair representing the first and last customers. This evaluation surely covers the current pair of first and last customers, which is assumed fixed in Ref. [2]. Fig. 1 and Eq. (13) illustrate the above description.

Consider

$$S(i,k) = \min_{j_i \in J^k} (C(i,j_s) + C(i,j_{s+1}) + C(j_1,j_f) - C(i^*,j_1) - C(i^*,j_f) - C(j_s,j_{s+1})), \tag{13}$$

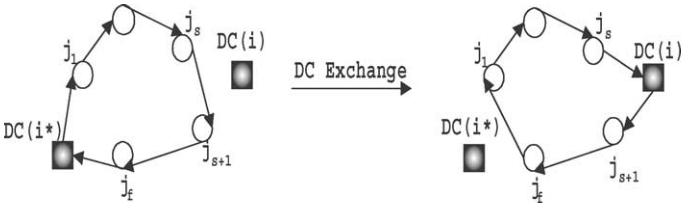

Fig. 1. Route reallocation.

where i^* is the current DC site to which route J^k is assigned; i is the index of potential DC sites, and $i^* \neq l$; J^k is the set of all customers in route k; (j_s, j_{s+1}) is a pair of adjacent customers in the route; $S(i,k)$ represents the savings if the customers on route k are reallocated from the current DC site to a new DC located in point i.

These sub-problems can be solved by either optimal or heuristic approaches. The mathematical formulation of the LAP is given below. As for the mathematical models for homogeneous and non-homogeneous fleet type VRPs, readers are referred to Refs. [4,33], respectively. Corresponding heuristic methods are discussed in the next section.

The formulation for the LAP of the first iteration in the solution procedure is as follows:

$$\mathrm{Min} \sum_{i \in I} \sum_{j \in J} C_{ij} z_{ij} + \sum_{i \in I} G_i y_i, \tag{13}$$

s.t.

$$\sum_{i \in I} z_{ij} = 1, \quad j \in J, \tag{14}$$

$$y_i \geqslant z_{ij}, \quad i \in I, j \in J, \tag{15}$$

$$\sum_{j \in J} d_i z_{ij} \leqslant V_i, \quad i \in I, \tag{16}$$

$$z_{ij} = 0,\ 1, \quad i \in I,\ j \in J, \tag{17}$$

$$y_i = 0,\ 1, \quad i \in I. \tag{18}$$

The formulation for the LAP starting from the second iteration in the solution procedure is as follows:

$$\mathrm{Min} \sum_{i \in I} \sum_{k \in K} S_{ik} z_{ik} + \sum_{i \in I} G_i y_i, \tag{19}$$

s.t.

$$\sum_{i \in I} z_{ik} = 1, \quad k \in K, \tag{20}$$

$$y_i \geqslant z_{ik}, \quad i \in I, k \in K, \tag{21}$$

$$\sum_{k \in K} d_i z_{ik} \leqslant V_i, \quad i \in I, \tag{22}$$

$$z_{ik} = 0,\ 1, \quad i \in I, k \in K,$$

$$y_i = 0,\ 1, \quad i \in I. \tag{23}$$

3.3. Heuristic solutions for sub-problems

Both LAPs and VRPs are difficult to solve. Hence, heuristic methods for the LAP and VRP, respectively, were developed. Since simulated annealing (SA) has been applied to a number of combinatorial problems with fairly good results [20], it was selected as the basis for developing search methods for both the LAP and VRP.

Metropolis [34] proposed an algorithm to simulate the annealing process. At each temperature small perturbations are given to atoms and thus result in energy change, ΔE. These changes are computed. If the new configuration results in energy reduction, the configuration is accepted. However, if the new configuration yields an energy increase, the configuration can still be accepted according to the probability function, $P(\Delta E) = \exp(-\Delta E/k_B T)$, where k_B is Boltzmann's constant and T is the current temperature. This check is performed by first selecting a random number from (0, 1). If the value is less than or equal to the probability value, the new configuration is accepted; otherwise, it is rejected.

In SA cycling occurs if the number of feasible neighborhood moves is limited. In this study to prevent cycling, the "tabu list" concept from the tabu search [35] is adopted and embedded in the framework of the SA algorithm to record the recent search trajectory. The tabu search is not trapped at the local optimum when no improvement is possible. To avoid cycling during the search for a solution, transformations leading to solutions recently found are prohibited. A data structure known as the tabu list stores this sort of information. The tabu list contains links discarded from the current solution during the last few iterations. Then, any neighborhood moves are tabu if each new link introduced into the solution is found in the tabu list.

3.4. Heuristic solution for LAP

In the LAP, the objective is to determine the number and locations of DCs and the allocation of customers for each DC selected. The solution obtained in this phase is used as an input to the VRP in the next phase. The entire solution procedure in this phase consists of two parts: the initial solution construction and improvements, as shown in Fig. 2.

3.4.1. Initial solution construction

At this stage, finding a heuristic algorithm giving rapid approximations to the optimum seems to be more appropriate than those well-developed methods which typically consume too much time. Hence, spacefilling curves (SFC) are chosen as the solution method due to their possessing the required properties described above.

"The spacefilling curves can be thought of as the limit of a sequence of recursive constructions whereby the square is divided into smaller squares, into which are copied scaled versions of the preceding construction [36]". SFC has the capability to transform the problem defined in unit

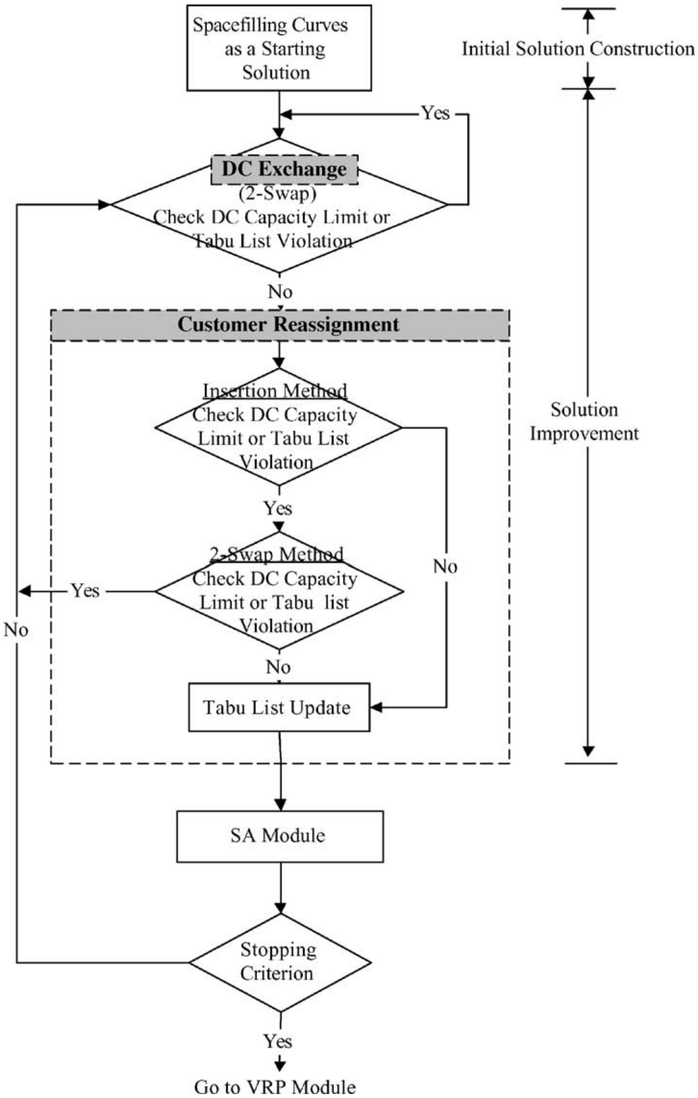

Fig. 2. Flowchart of solution procedure for LAP.

square into an easier one defined in unit interval. The solution to the easier problem can, thus, be a rapid approximation to the solution of the original problem while preserving the "nearness" among points. Since the LAP considers the spatial relationship between demand points and potential DC sites, the SFC is, hence, a very good tool to solve the LAP. The SFC algorithm for solving the LAP is described as follows:

Step 1: For each demand point and potential DC site in unit square, calculate, via SFC, a corresponding position on the unit interval.

Step 2: Sort the demand points and potential DC sites according to their corresponding positions on the unit interval.

Step 3: Form the clusters on the basis of the DC's capacity and give ranks to these clusters from the left to the right side of the unit interval. Define the total number of clusters as *TC*.

Step 4: Suppose the number of potential DC sites equals n, the DC sites whose ranks are $\lceil n/2TC \rceil, \lceil 3n/2TC \rceil, \ldots, \lceil 2(TC-1)n/2TC \rceil$ are chosen as the open DCs corresponding to the clusters generated in *Step* 3.

3.4.2. Improvement of iterative solution

At this stage in the procedure, the starting solution consisting of the DCs selected and the allocation of customers to each DC (i.e., clusters) provided by the SFC is improved through a sequence of neighborhood moves. Two types of neighborhood moves are performed, namely, the DC Exchange, followed by the Customer Reassignment, as shown in Fig. 2.

The DC Exchange improvement considers possible exchanges between selected DCs and those not yet selected. The 2-swap method is used to provide the required exchange. As for the Customer Reassignment improvement, the insertion and 2-swap methods are used. In the insertion method a customer is selected randomly from a cluster and then inserted into another cluster. While in the 2-swap method, two customers are chosen randomly from two different clusters and their positions exchanged.

Note that for any legal moves generated by the insertion method, the 2-swap procedure can be skipped. Only those failing in the insertion procedure (e.g., violating the DC capacity restriction or a recorded tabu move) must proceed to the 2-swap procedure. The reason for this setting (placing the insertion method before the 2-swap method) is that the insertion method seems to have better opportunities for satisfying the restriction of the DC capacity earlier than the 2-swap method. Before accepting any move, restrictions from the DC capacity limit and the tabu list must be checked. As soon as the check is passed, feasible solutions generated by the above procedure are then recorded in the tabu list and applied to the simulated annealing procedure. The process is repeated until convergence or the stopping criterion is met. The solution obtained at this moment is then input to the VRP module described below.

3.5. Heuristic solution for VRP

Before proceeding to the VRP module, vehicles are assigned to customers in each cluster on the basis of the vehicular capacity limit. Such an assignment is for the case in which the fleet type is homogeneous. When a multiple fleet type is involved, vehicles with larger capacities are assigned before those with smaller capacities. After the process of vehicle assignment, a feasible initial routing solution is thus generated and ready for the improvement discussed in Section 3.5.1. Fig. 3 shows the solution procedure for the entire VRP module. Starting with a feasible routing solution, the solution improvement phase includes "within-route" and "between-routes." The solutions obtained from the initial and improvement phases are then applied to the simulated annealing module. The process is repeated until the stopping criterion is met.

In addition to the SA module associated with the entire improvement phase, there is another SA module embedded in the within-route improvement. The reason is that whenever between-routes improvement is performed, within-route improvement is necessary only for those two routes where changes occurred. Therefore, the within-route improvement is actually driven by the between-routes improvement. If only one iteration is performed in the within-route improvement module,

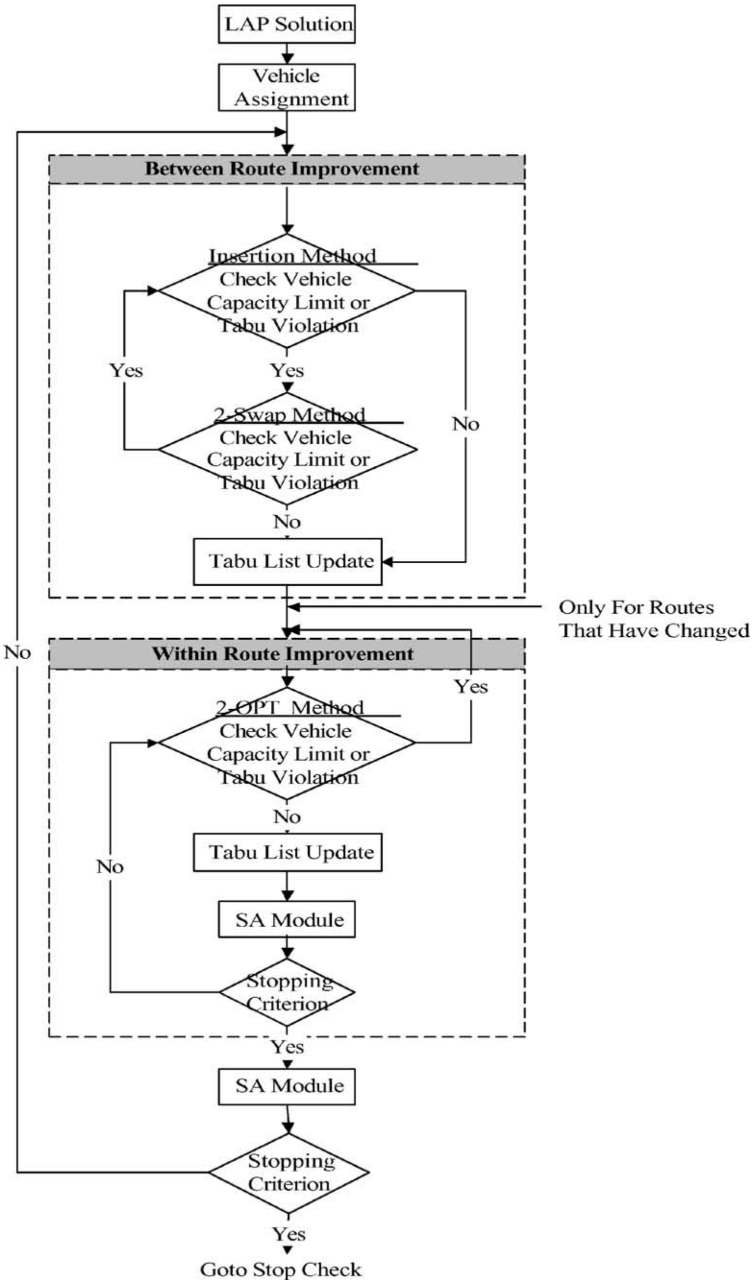

Fig. 3. Flowchart for solution procedure for VRP.

not much improvement can be realized in terms of the solution quality. Therefore, a complete SA module is applied in order to obtain a significant within-route improvement.

The between-route improvement methods exchange or relocate the positions of customers between any two routes in a feasible solution. The within-route search method seeks a move to improve the objective function (route distance or travel time) by altering the visiting sequence within a route. Methods such as 2-opt [37], 3-opt [37], or Or-opt [32] are in this class of neighborhood moves.

For the within-route improvement, 2-opt is used in this study, while insertion and 2-swap are selected for the between-route improvement for their simplicity and easy implementation. The *Insertion* method attempts to insert a customer from one route into another route. The 2-*swap* method tries to exchange the positions of customers belonging to two different routes without changing the sequence of each route. The 2-*opt* method is a little more complicated. Two edges not adjacent to each other in a feasible route are exchanged for another two edges not in that solution as long as the result remains in the tour and the length of that tour is less than that of the previous tour. This exchanging procedure repeats until no feasible exchange that can improve the current solution can be found.

As in the LAP module, checks regarding the vehicle capacity limit and the tabu list must be performed before accepting any move.

4. Computational results

This section illustrates the proposed solution method for the multi-depot location routing problem by using test problems from the literature as well as newly created problems.

4.1. Results for Perl's test problems

Three problems from Perl [8] are used to test the performance of the proposed method. Before the results are displayed, notice that:

1. Three problems with 12,55,85 customers are analyzed, respectively.
2. Hansen [13] presented a heuristic approach, which outperformed Perl's results in all three problems.
3. Hansen changed the coordinate of customer No. 12 from (17,58) to (17,51) in test problem #3 to fit Perl's graphical presentation. This study follows Hansen's correction.
4. The objective function of the proposed mathematical model in Section 2 differs from that of Perl's and Hansen's studies. However, for comparison with Perl's and Hansen's results, the current objective function has been changed to comply with theirs. In other words, the objective function is composed of three components: warehouse establishing cost, delivery cost, and variable warehousing cost. In addition, the delivery fleet consists of standard vehicles as in Perl's and Hansen's studies.
5. Only the optimal solution of test problem #1 is obtained in an acceptable amount of time. The optimality test is performed on a Pentium II 400 personal computer using LINGO for Windows as the modeling language, while the heuristic test is implemented by using Visual C++ on the same computer.

Table 1 compares the results of Perl's, Hansen's, and the proposed method for three test problems. It can be observed from these tables that the proposed method is able to find the optimal

Table 1
Results for comparison of test problems #1, #2, and #3

Problem No.	Methods	Depots	Routes	Distances	Costs	CPU seconds
1	Optimum	1	2	103.97	355.58	2501
1	Perl's solution	1	2	103.97	355.58	—
1	Hansen's solution	1	2	103.97	355.58	—
1	Proposed method	1	2	103.97	355.58	0.62
2	Perl's solution	2, 10, 12	10	4261.32	5795.62	—
2	Hansen's solution	2, 7, 12, 13	10	3843.67	5617.67	—
2	Proposed method	5, 10, 12	10	3998.28	5532.28	4.12
3	Perl's solution	2, 4, 5	11	5415.96	7789.96	—
3	Hansen's solution	2, 4, 6	11	5177.61	7551.61	—
3	Proposed method	2, 4, 6	12	5407.21	7781.21	10.21

solution for test problem #1 in only 0.62 s, whereas the optimization approach takes 2501 s. For problem #2, the proposed method outperformed both Perl's and Hansen's results in terms of total costs. Notice that Hansen's method results in less distance because one more DC is established than in the proposed method. The proposed method still provides a better solution than Perl's study in test problem #3; however, Hansen's method gives even fewer costs and distance than the proposed method.

4.2. Results in new problem instances

In addition to the set of test problems from Ref. [8], the proposed method was tested over a number of newly generated problems. The results of the traditional MDLRP with a single fleet type are shown in Section 4.2.1, whereas Section 4.2.2 lists results of the MDLRP with a multiple-fleet type and limited number of vehicles. Fixed settings for both cases are described first.

1. The coordinate for each customer is generated in $[0,100]^2$ according to a uniform distribution.
2. The demand from each customer is generated on the basis of a uniform distribution on [1,50].
3. The fixed cost for establishing each DC is set at 240.
4. The fixed dispatching cost for each vehicle is set at 20.
5. The cooling rate used in SA is set at 0.99.
6. The final temperature of SA is set at 1.

Except for the fixed settings, the problems are generated, basically, through combinations of several problem parameters, which include the number of customers, the number of potential DC sites, the vehicle filling coefficient, and algorithmic parameters such as the initial temperature of the SA, the stopping criterion of the VRP module, and the tabu list size. To introduce the parameter settings, the following notations are first defined:

Dem_node number of customers
DC_node number of potential DC sites

Node: Total number of nodes, Node = Dem_node + DC_node
L: Potential DC ratio
Tdem: Total demand
Dcap: Capacity restriction for DCs
Vcap: Vehicle capacity
M: Filling coefficient for vehicles

4.2.1. MDLRP with single fleet type

The corresponding problem and algorithmic parameters are described first:

Problem parameters: The settings for the problem parameters are described as follows:

1. The number of customers, Dem_node, has four cases: 50,75,100,150.
2. The number of potential DC sites, DC_node, is set equal to the product of Node and L, i.e., DC_node = Node*L, where L has three values: 1/20, 1/10, and 1/5.
3. The capacity limit for DC, Dcap, is defined as Tdem/(Node*(1/20)).
4. To assure that there are at least two vehicles available in each DC, through the use of the "filling coefficient" M, Vcap = (Dcap/2)*M. Three values of M were tested: 0.9, 0.7, and 0.5. As a matter of fact, problem tightness can be controlled by using different vehicle capacities.

For all the formulas presented above, if the resulting value is a non-integer, replace it with the nearest integer.

Algorithmic parameters: Three algorithmic parameters are chosen for constructing the experimental runs, namely, the temperature of the SA, the number of iterations for the within-route improvement as the stopping criterion in the VRP module, and the tabu list size throughout the solution procedure. The initial temperature T_0 was taken as 100 and 200. There were three values for the stopping criterion of the within-route improvement in the VRP module, i.e., 50,100, and 150. The tabu list size takes the value of either 7 or 14.

Due to the stochastic features the proposed method might have, three independent runs were performed on each parameter combination instance. For ease of presentation, the 50-node problem is used as an example to illustrate the test results. Results associated with problems of 75,100, and 150 nodes are listed in appendix A.

Considering all possible combinations of problem parameters and algorithmic parameters, 324 ($= 3^4 \times 2^2$) instances were generated and solved for the problem with 50 nodes. In addition to the 50 nodes, there are problems with 75, 100, and 150 nodes, for a total of 1296 runs.

After analyzing the results, including the objective function value and run time carefully, it was discovered that there is no obvious relationship between the algorithmic parameters and the objective function values. However, as the tabu list size increases, the run time increases stably. Similar trends can also be found in both the stopping criterion of the VRP module and the initial temperature of the SA. By examining the computational results of all settings for the 50-node LRP of homogeneous fleet type, it was discovered that the algorithmic parameters do not have a significant effect on the solution values. Computational results with concise format are shown in Table 2. In Table 2, for each problem setting, only the results from the best single run and the best average accompanied by the CPU time and the corresponding algorithmic parameters combinations are reported. It can be seen from Table 2 that as the vehicle capacities become more

Table 2
Computational results for homogeneous fleet type MDLRP of 50 nodes

DC (L)	Vcap (M)	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	0.9	776.84	1.60	(200,100,14)	776.84	1.59	(200,100,14)
	0.7	809.14	2.02	(100,100,14)	828.20	2.17	(100,150,14)
	0.5	935.82	1.84	(100,150,14)	935.82	1.84	(100,150,14)
1/10	0.9	713.95	2.25	(100,150,7)	713.95	2.25	(100,150,7)
	0.7	753.65	1.18	(200,50,7)	753.65	1.19	(200,50,7)
	0.5	847.43	3.02	(200,150,14)	869.58	2.32	(200,100,14)
1/5	0.9	700.01	1.71	(200,100,7)	700.01	1.63	(200,100,7)
	0.7	656.57	1.13	(200,50,7)	656.57	1.11	(200,50,7)
	0.5	725.06	3.00	(200,100,7)	770.39	2.89	(200,100,7)

restrictive, the total costs increase, because more vehicles must be dispatched to satisfy the customers' demands.

4.2.2. MDLRP with multiple fleet type and limited number of vehicles

For the MDLRP of the multiple-fleet type and a limited number of vehicles, one additional parameter must be introduced: MIX, the number of fleet types. Three values of MIX were tested: 2, 3, and 4. The filling coefficients of the vehicles, the fixed costs, and the number of vehicles available for each fleet type are given in Table 3. By examining the results, it is found that the algorithmic parameters do not have an effect on the objective values either, as in the case of the single-fleet type MDLRP. A concise table with a format similar to Table 2 was constructed in Table 4. Results associated with problems of 75, 100, and 150 nodes can be found in Appendix B.

It can be observed that the more different types of fleets there are, the fewer total costs are yielded. This is easily explained, because more types bring more choices and, hence, fewer costs.

5. Discussion

Some may question that it is appropriate to rely on routing decisions when locating depots, since location decisions are strategic while routing decisions are operational in practice. The study by Salhi and Nagy [38] seems to be the only one that has ever conducted a consistency and robustness analysis through simulation to show that the nested methods presented earlier [17,18] consistently produce promising solutions and are stable enough to be reliable for practical use. In their nested methods, an initial set of depots is generated first and the resulting multi-depot routing problem is then solved. After that, the neighborhood solutions of the current depot locations are explored by the three moves *add*, *drop*, and *shift*. In each iteration, the total costs of all possible moves are evaluated and the one with the largest cost improvement is selected.

Table 3
Problem related-information for the LRP of multiple-fleet type

MIX	Filling coefficients of vehicles	Fixed cost of vehicles	Number of vehicles of each fleet type
2	(0.9, 0.8)	(200, 300)	(5, 4)
3	(0.9, 0.8, 0.7)	(200, 300, 400)	(5, 4, 3)
4	(0.9, 0.8, 0.7, 0.6)	(200, 300, 400, 600)	(5, 4, 3, 2)

Table 4
Computational results for 50-node MDLRP of multiple-fleet type

DC (L)	MIX	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	2	3310.61	1.86	(200, 150, 14)	3310.61	1.85	(200, 150, 14)
	3	2636.44	0.65	(100, 50, 7)	2636.44	0.67	(100, 50, 7)
	4	2418.19	2.20	(200, 150, 7)	2454.43	2.32	(200, 150, 7)
1/10	2	2828.60	0.91	(100, 100, 14)	2828.60	0.90	(100, 100, 14)
	3	2542.59	0.56	(200, 50, 7)	2542.59	0.51	(200, 50, 7)
	4	2187.73	1.98	(100, 150, 14)	2194.25	0.60	(200, 50, 14)
1/5	2	2717.93	0.45	(100, 50, 7)	2717.93	0.42	(100, 50, 7)
	3	2202.57	1.20	(100, 150, 14)	2233.77	1.74	(200, 150, 14)
	4	2048.94	0.80	(100, 100, 14)	2048.94	0.78	(100, 100, 14)

The nested methods and the proposed algorithm differ mainly in the location phase. In the proposed algorithm, starting from the second iteration, routes are unified to include more customers for reducing the number of DCs established. This step is equivalent to the *drop* move in the nested methods. In addition, a saving matrix is generated at the end of each iteration of the proposed algorithm. This matrix guides the algorithm to reallocate the consolidated nodes from the current DC to other DCs with less distance resulting. This step is equivalent to the *shift* move in the nested methods. Thus, the nested methods by Salhi and Nagy and our decomposition method are actually similar in terms of the method for finding the neighborhood solutions of the current set of open locations. The proposed heuristic algorithm should thus share the desirable properties of being robust and consistent possessed by the nested methods, as concluded by Salhi and Nagy.

6. Concluding remarks

In this paper the researchers have first relaxed the assumptions typically made in the traditional MDLRP that a standard fleet size is involved and an unlimited number of vehicles are available.

Hence, a modified model has been presented; then, a simulated annealing-based decomposition approach for both the traditional and modified MDLRPs has been described. The computational results indicate that in the traditional MDLRP, the proposed heuristic produced an optimal solution for the test problem with 12 nodes. For the problem with 55 nodes, it outperformed both Perl's [8] and Hansen's [13] methods. For the problem with 85 nodes, the proposed method still provides a better solution than Perl's work; however, Hansen's method results in even fewer costs than the proposed method.

The primary objective of this research has been to develop an efficient approach for solving the MDLRP and its variants. Considerable efforts were devoted to the design of the decomposition logic and heuristic algorithms. In this research, the MDLRP was decomposed into two sub-problems, i.e., the location-allocation problem and the vehicle routing problem. In comparison with the three sub-problems (MDVDP, WLAP, MDRAP) in Perl's work, in which the VRP must be solved twice in each iteration, the proposed method seems to be more straightforward conceptually and more efficient, since the VRP was solved only once in each iteration.

In addition to a more straightforward and efficient decomposition logic, in the phase for generating a starting solution, only necessary DCs are open. In contrast with Perl's opening every potential DC and then closing some of them in later iterations, the present setting saves some unnecessary run time. For the problem with 50 nodes, all cases were solved in 3 s, while the instance taking the longest run time occurred in the problem with 150 nodes, and was finished in 26 s.

The performance of the proposed heuristic procedure under various parameter settings has also been provided, which can be very helpful in identifying suitable SA algorithmic parameters for obtaining quick and favorable solutions under different circumstances. From the results it can be concluded that algorithmic parameters such as the initial temperature of the SA, the stopping criterion of the VRP module, and the tabu list size of the entire solution procedure do not have an obvious relationship to the objective values. However, the run time tends to increase as the values of the above three parameters increase.

Two improvements in the solution of the MDLRP and its variants can be envisaged. First, alternative ways of generating a starting solution can be exploited. Second, mathematical and/or heuristic approaches considering the stochastic demands of each customer can be developed to reflect reality more accurately.

Acknowledgements

This work was supported by the National Science Council of Taiwan, under grants NSC 89-2213-E-212-043. This support is gratefully acknowledged. The authors are grateful to the anonymous referees for their helpful comments. We also thank Professor S. Salhi for supplying some of his works.

Appendix A. Computational results for homogeneous fleet type MDLRP

(Tables 5–7)

Table 5
Computational results for homogeneous fleet type MDLRP with 75 nodes

DC (*L*)	Vcap (*M*)	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	0.9	1106.81	3.13	(100, 150, 7)	1106.81	3.13	(100, 150, 7)
	0.7	1163.01	2.52	(100, 50, 14)	1166.01	1.92	(100, 50, 7)
	0.5	1397.11	2.89	(100, 100, 7)	1492.17	6.66	(200, 100, 14)
1/10	0.9	911.22	2.89	(200, 50, 14)	912.69	2.54	(100, 50, 7)
	0.7	958.99	1.85	(100, 50, 7)	958.99	1.85	(100, 50, 7)
	0.5	1118.95	4.72	(200, 150, 14)	1118.95	4.72	(200, 150, 14)
1/5	0.9	797.96	2.76	(200, 100, 14)	797.96	2.76	(200, 100, 14)
	0.7	837.60	3.95	(100, 100, 14)	837.60	3.95	(100, 100, 14)
	0.5	1017.98	2.96	(100, 100, 7)	1017.98	2.96	(100, 100, 7)

Table 6
Computational results for homogeneous fleet type MDLRP with 100 nodes

DC (*L*)	Vcap (*M*)	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	0.9	1384.24	5.97	(200, 50, 14)	1384.24	5.97	(200, 50, 14)
	0.7	1329.43	8.44	(100, 150, 14)	1383.43	4.63	(100, 50, 14)
	0.5	1751.36	5.47	(100, 100, 14)	1800.49	5.83	(200, 50, 7)
1/10	0.9	1127.82	5.06	(200, 100, 7)	1127.82	5.06	(200, 100, 7)
	0.7	1161.02	8.18	(100, 150, 7)	1108.02	8.18	(100, 150, 7)
	0.5	1404.19	4.06	(200, 100, 14)	1404.19	4.06	(200, 100, 14)
1/5	0.9	1000.14	11.05	(100, 150, 7)	1000.14	11.05	(100, 150, 7)
	0.7	946.78	7.98	(200, 100, 14)	1002.69	7.98	(200, 100, 14)
	0.5	970.49	4.17	(100, 100, 14)	970.49	4.17	(100, 100, 14)

Table 7
Computational results for homogeneous fleet type MDLRP with 150 nodes

DC (*L*)	Vcap (*M*)	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	0.9	1601.24	12.70	(200, 150, 14)	1601.24	12.70	(200, 150, 14)
	0.7	1662.78	30.33	(200, 150, 7)	1709.61	9.43	(100, 100, 7)
	0.5	2043.54	18.97	(100, 100, 14)	2092.83	18.97	(100, 100, 14)
1/10	0.9	1489.34	20.33	(200, 150, 14)	1489.34	20.33	(200, 150, 14)
	0.7	1502.70	26.09	(200, 50, 7)	1570.73	26.09	(200, 50, 7)
	0.5	1947.20	8.49	(200, 50, 7)	1947.20	8.49	(200, 50, 7)
1/5	0.9	1206.45	10.85	(200, 100, 7)	1216.70	17.81	(100, 150, 14)
	0.7	1166.79	11.42	(100, 150, 7)	1166.79	11.42	(100, 150, 7)
	0.5	1469.09	14.44	(200, 100, 14)	1513.09	14.44	(200, 100, 14)

Appendix B. Computational results for multiple-fleet type MDLRP

(Tables 8–10)

Table 8
Computational results for 75-node MDLRP of multiple fleet type

DC (L)	MIX	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	2	4862.08	1.46	(100, 100, 7)	4862.08	1.46	(100, 100, 7)
	3	3990.67	2.94	(200, 150, 14)	3995.16	2.96	(200, 150, 7)
	4	3700.73	2.45	(200, 150, 14)	3700.73	2.45	(200, 150, 14)
1/10	2	4275.50	0.66	(100, 50, 14)	4275.50	0.66	(100, 50, 14)
	3	3673.21	0.58	(100, 50, 14)	3673.21	0.58	(100, 50, 14)
	4	3264.54	0.69	(100, 50, 14)	3264.54	0.69	(100, 50, 14)
1/5	2	4063.55	1.33	(200, 100, 14)	4063.55	1.33	(200, 100, 14)
	3	3297.24	0.84	(200, 50, 14)	3297.24	0.84	(200, 50, 14)
	4	2918.86	1.64	(200, 100, 7)	2959.59	1.64	(200, 50, 7)

Table 9
Computational results for 100-node MDLRP of multiple fleet type

DC (L)	MIX	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	2	7638.73	3.46	(200, 150, 7)	7638.73	3.46	(200, 150, 7)
	3	6052.78	2.38	(200, 100, 14)	6052.78	2.38	(200, 100, 14)
	4	5743.54	3.91	(200, 150, 14)	5743.54	3.91	(200, 150, 14)
1/10	2	6992.32	1.70	(100, 100, 7)	6992.32	1.70	(100, 100, 7)
	3	5652.45	0.78	(100, 50, 7)	5652.45	0.78	(100, 50, 7)
	4	5015.27	2.21	(200, 100, 14)	5125.19	2.21	(200, 100, 14)
1/5	2	6125.29	2.08	(200, 100, 7)	6234.40	2.08	(200, 100, 7)
	3	5067.63	3.00	(200, 150, 14)	5067.63	3.00	(200, 150, 14)
	4	4607.25	1.61	(200, 100, 14)	4607.25	1.61	(200, 100, 14)

Table 10
Computational results for 150-node MDLRP of multiple fleet type

DC (L)	MIX	Best single run			Best average run		
		Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)	Objective value	CPU time (s)	Parameter setting (T_0, Ite_No, T_list)
1/20	2	10394.13	1.28	(100, 50, 7)	10394.13	1.28	(100, 50, 7)
	3	8117.04	3.38	(200, 100, 14)	8146.76	1.61	(200, 50, 14)
	4	7746.84	5.01	(200, 150, 14)	7768.32	5.01	(200, 150, 14)
1/10	2	9864.74	2.43	(200, 100, 7)	9864.74	2.43	(200, 100, 7)
	3	7853.70	3.68	(200, 100, 7)	7875.60	1.37	(200, 50, 14)
	4	7119.67	1.39	(200, 50, 14)	7119.67	1.39	(200, 50, 14)
1/5	2	8709.40	1.11	(100, 50, 7)	8709.40	1.11	(100, 50, 7)
	3	6734.67	1.46	(200, 50, 7)	6749.69	3.66	(200, 100, 14)
	4	6221.35	2.98	(100, 150, 7)	6221.35	2.98	(100, 150, 7)

References

[1] Webb MH. Cost function in the location of depots for multi-delivery journeys. Operations Research Quarterly 1968;19:311–28.

[2] Perl J, Daskin MS. A warehouse location problem. Transportation Research Quarterly 1985;19B:381–96.

[3] Laporte G, Nobert Y, Arpin D. An exact algorithm for solving a capacitated location-routing problem. Annals of Operations Research 1986;239–310.

[4] Golden BL, Magnanti TL, Nguyen HQ. Implementing vehicle routing algorithms. Networks 1977;7:113–48.

[5] Or I, Pierskalla WP. A transportation location-allocation model for regional blood banking. AIIE Transactions 1979;11:86–95.

[6] Jacobsen SK, Madsen OB. A comparative study of heuristics for a two-level routing-location problem. European Journal of Operational Research 1980;5:378–87.

[7] Srikar B, Srivastava R. Solution methodology for the location-routing problem. Presented at the ORSA/TIMS Conference, Orlando, USA, 1983.

[8] Perl J. A unified warehouse location-routing analysis. UMI Dissertation Information Service, USA, 1983.

[9] Laporte G, Nobert Y. An exact algorithm for minimizing routing and operating costs in depot location. European Journal of Operational Research 1981;6:224–6.

[10] Laporte G. Location-routing problems. In: Gloden BL, Assad AA, editors. Vehicle routing: methods and studies. Amsterdam: North-Holland, 1988. p. 163–96.

[11] Laporte G, Nobert Y, Taillefer S. Solving a family of multi-depot vehicle routing and location-routing problems. Transportation Science 1988;22:172.

[12] Laporte G, Louveaux F, Mercure H. Models and exact solutions for a class of stochastic location-routing problems. European Journal of Operational Research 1989;39:71–8.

[13] Hansen PH, Hegedahl B, Hjortkjær S, Obel B. A heuristic solution to the warehouse location-routing problem. European Journal of Operational Research 1994;76:111–27.

[14] Srivastava R, Benton WC. The location-routing problem: considerations in physical distribution system design. Computers and Operations Research 1990;17:427–35.

[15] Chien TW. Heuristic procedures for practical-sized uncapacitated location-capacitated routing problems. Decision Sciences 1993;24:995–1021.
[16] Salhi S, Fraser M. An integrated heuristic approach for the combined location vehicle fleet mix problem. Studies in Locational Analysis 1996;8:3–21.
[17] Nagy G, Salhi S. Nested heuristic methods for the location-routing problem. Journal of Operational Research Society 1996;47:1166–74.
[18] Nagy G, Salhi S. A nested location-routing heuristic using route length estimation. Studies in Locational Analysis 1996;10:109–27.
[19] Min H, Jayaraman V, Srivastava R. Combined location-routing problems: a synthesis and future research directions. European Journal of Operational Research 1998;108:1–15.
[20] Van Laarhoven PJ. Theoretical and computational aspects of simulated annealing. The Netherlands: Center for Mathematics and Computer Science, 1988.
[21] Christofides N, Mingozzi A, Toth P. The vehicle routing problem. In: Christofides N, Mingozzi A, Toth P, Sandi C, editors. Combinatorial optimization. New York: Wiley, 1979. p. 315–38.
[22] Cooper L. Location-allocation problems. Operations Research 1963;11:331–43.
[23] Cooper L. Heuristic methods for location-allocation problems. SIAM Review 1964;6:37–53.
[24] Clarke G, Wright JW. Scheduling of vehicles from a central depot to a number of delivery points. Operations Research 1964;12:568–81.
[25] Duhamel C, Potvin J, Rousseau J. A tabu search heuristic for the vehicle routing problem with backhauls and time windows. Transportation Science 1997;31:49–59.
[26] Fisher ML, Jaikumar R. A generalized assignment heuristic for vehicle routing. Networks 1981;11: 109–24.
[27] Gendreau M, Hertz A, Laporte G. A tabu search heuristic for the vehicle routing problem. Management Science 1994;40:1276–90.
[28] Osman IH. Metastrategy simulated annealing and tabu search algorithms for the vehicle routing problem. Annals of Operations Research 1993;41:421–51.
[29] Toth P, Vigo D. The vehicle routing problem. SIAM, Philadelphia Monograph on Discrete Mathematics and Applications, 2000.
[30] Bongartz I, Calamai PH, Conn AR. A projected method for l_p norm location-allocation problem. Mathematical Programming 1994;66:283–312.
[31] Hodgson MJ, Rosing KE, Shmulevitz F. A review of location-allocation applications literature. Isolde VI Survey Papers. Studies in Locational Analysis 1993;5:3–29.
[32] Or I. Traveling salesman-type combinatorial problems and their relations to the logistics of blood banking Dissertation, Northwestern University, USA, 1976.
[33] Salhi S, Rand GK. Incorporating vehicle routing into the vehicle fleet composition problem. European Journal of Operational Research 1993;66:313–30.
[34] Metropolis N, Rosenbluth AW, Teller AH. Equation of state calculations by fast computing machines. Journal of Chemical Physics 1953;21:1087–92.
[35] Glover F, Laguna M. Tabu search. Boston: Kluwer Academic Publishers, 1997.
[36] Bartholdi J, Platzman LK. Heuristics based on spacefilling curves for combinatorial problems in Euclidean space. Management Science 1988;34:291–305.
[37] Lin S. Computer solutions of the traveling salesman problem. Bell System Technology Journal 1965;44: 2245–69.
[38] Salhi S, Nagy G. Consistency and Robustness in Location-Routing. Studies in Locational Analysis 1999;13: 3–19.

Tai-Hsi Wu is an associate professor in Industrial Engineering Department at Da-Yeh University, Taiwan. He received his master and Ph.D. degrees in Industrial Engineering from Texas A&M University, USA, in 1994. His research and teaching interests include developing models and algorithms for production and transportation problems.

Chinyao Low is an associate professor in Industrial Engineering Department at Da-Yeh University, Taiwan. He received his Ph.D. degree in Industrial Engineering from Cleveland State University, USA, in 1993. His research interests include FMS scheduling and production management.

Jiunn-Wei Bai received his master degree in Industrial Engineering from Da-Yeh University in 1999.