A Hybrid Metaheuristic Algorithm for the Multi-depot Vehicle Routing Problem with Time Windows

Tong Zhen
College of Information Science and Engineering
Henan University of Technology
Zhengzhou 450001, China
E-mail: zhentong1217@163.com

Qiuwen Zhang
College of Information Science and Engineering
Henan University of Technology
Zhengzhou 450001, China
E-mail: zhangqwen@sohu.com

***Abstract*—The multi-depot vehicle routing problem with time windows (MDVRPTW) is an extension to the classical Vehicle Routing Problem (VRP); it is the major research topics in the supply chain management field. It aims to designing a set of minimum-cost routes for a vehicle fleet servicing many customers with known demands and predefined time windows. In this paper a hybrid metaheuristic algorithm is proposed to solve MDVSPTW successfully by ants transfer policy and algorithm to construct solution designed in this dissertation. To improve hybrid ant colony algorithm performance, a local search improvement algorithm that explores a large neighborhood of the current solution to discover a cheaper set of feasible routes. The neighborhood structure comprises all solutions that can be generated by iteratively performing node exchanges among nearby trips followed by a node reordering on every route. The experiment indicates the validity of the technique to MDVSPTW with the above-mentioned conditions.**

Keywords-hybrid; Metaheuristic Algorithm; multi-depot; vehicle routing problem with time windows

I. INTRODUCTION *(HEADING 1)*

A company that transports goods to supply customers usually needs to plan the routes that the fleet must follow, since the transportation means a high percentage of the value added to goods, 5% to 20% of the total cost [1], therefore it is necessary to find efficient ways to plan it.

The distribution of finished products from depots to customers is a practical and challenging problem in logistics management. Better routing and scheduling decisions can result in higher level of customer satisfaction because more customers can be served in a shorter time. The distribution problem is generally formulated as the vehicle routing problem (VRP). Nevertheless, there is a rigid assumption that there is only one depot. In cases, for instance, where a logistics company has more than one depot, the VRP is not suitable. To resolve this limitation, this paper focuses on the VRP with multiple depots, or multi-depot VRP (MDVRP). The MDVRP is NP-hard, which means that an efficient algorithm for solving the problem to optimality is unavailable. To deal with the problem efficiently, a hybrid metaheuristic algorithm is developed in this paper.

This paper is organized as follows. Section 2 surveys on the relevant literature. Section 3 describes the hierarchy of decisions in the MDVRP. Section 4 discusses the principles of the algorithms used to solve the MDVRP. Section 5 compares the performance of the algorithms. Finally, Section 6 concludes the paper.

II. VEHICLE ROUTING PROBLEM (VRP)

The classical problem of VRP is defined on a graph with a differentiated vertex (depot) that has travel costs or times associated with each arc. A set of m vehicles routes starting and ending at the depot, with minimal total cost, have to be designed in such a way that each of the remaining vertices is visited by exactly one vehicle. The value of m can be part of the data or the decision variables.

The types of VRPs are built by adding new restrictions such as capacity in the vehicles (CVRP, [2]), time windows for the customers and depots (VRPTW, [3]), several depots to supply the demands (MDVRP, [4]), split deliveries (SDVRP, [5],), multiple use of vehicles (VRPM, [6]), vehicles with a heterogeneous capacity (HVRP, [7]), and others.

In the revised literature, most of scientific papers target a specific VRP problem. Only a few papers like [8] present approaches to solve several types of VRPs. In order to overcome these limitations a methodology to solve several types of VRP problems that are involved in the definition of real-world transportations instances is proposed.

III. THE MULTI-DEPOT VEHICLE ROUTING PROBLEM WITH TIME WINDOWS(MDVRPTW)

Consider a distribution company with multiple depots. The number and locations of the depots are predetermined. Each depot is large enough to store all the products ordered by the customers. A fleet of vehicles with limited capacity is used to transport the products from depots to customers. Each vehicle starts and finishes at the same depot. The location and demand of each customer is also known in advance. Each customer is visited by a vehicle exactly once. This practical distribution problem can be regarded as the MDVRP, in which there are three decisions as shown in Fig. 1. The decision makers first need to cluster a set of customers to be served by the same depot, that is, the grouping problem. They then have to assign customers of the same depot to several routes so that the vehicle capacity constraint is not violated. At last, the decision on delivery sequence of each route is made. Generally, the objective of the MDVRP is to minimize the total delivery distance or

978-0-7695-3610-1/09 $25.00 © 2009 IEEE
DOI 10.1109/NSWCTC.2009.167

time spent in serving all customers. Shorter delivery time results in higher level of customer satisfaction. Besides, the objective can also be the minimization of the number of vehicles needed. Fewer vehicles imply that the total operation cost is reduced. No matter which type of objectives is defined, the ultimate goal of the MDVRP is to increase the efficiency of the delivery.

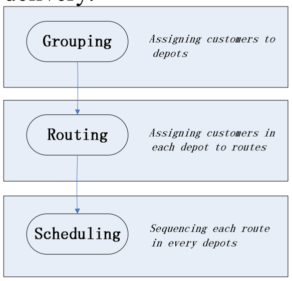
Grouping
Assigning customers to depots
Routing
Assigning customers in each depot to routes
Scheduling
Sequencing each route in every depots

Figure 1. The hierarchy of decisions in the MDVRP

A. Model Assumptions

a. Problem data are known with certainty and remain invariant with time; i.e. a deterministic, static VRPTW problem is studied.

b. Either pick-up or delivery services are provided to customers but not both.

c. Each pick-up or delivery node must be visited within the specified time window just once.

d. Though the problem can involve several depots, each route should start and end at the same depot.

e. The total load transported by a vehicle must never exceed its capacity.

f. Time-window and maximum trip duration constraints can be relaxed by including penalty cost terms in the objective function that linearly increases with the time-window violation or the trip over duration.

B. Mathematical Formulation and Column Generation

First, let us associate with our problem the directed multigraph G = (V, A) where each trip T_i is represented by a trip-vertex *i* and each depot D_k, k □ K, by a source-vertex n + k and a sink-vertex n+|K|+k (every depot is duplicated into a source-depot and a sink-depot). The vertex set V of G is then the union of sets N= {1, 2,..., n},{n+1,n+2,...,n+|K|} and {n+|K|+1,n+|K|+2,...,n+2|K|}; the arc set A consists of |K| sets $A^1,...,A^{|K|}$ such that each A^k is the union of the following three arc sets:

$$\min Z = \sum_{h=1}^{H}\left\{\sum_{k=1}^{K_h}\left[\sum_{i=1}^{n_{hk}} d_{r_{hk(i-1)}r_{hki}} + d_{r_{hkn_{hk}}r_{hk0}} \bullet sign(n_{hk})\right]\right\} \tag{1}$$

$$\text{s.t.} \quad \sum_{i=1}^{n_{hi}} q_{hr_{hki}} \le Q_{hk} \tag{2}$$

$$d_{r_{hk(i-1)}r_{hki}} + d_{r_{hkn_{hk}}r_{hk0}} \bullet sign(n_{hk}) \le D_{hk} \tag{3}$$

$$0 \le n_{hk} \le L_h \tag{4}$$

$$\sum_{k=1}^{K_h} n_{hk} = L_h \tag{5}$$

$$\sum_{h=1}^{H} L_h = M \tag{6}$$

$$R_{nk} = \{r_{nki} \mid r_{nki} \in \{1,2,...,L_h\}, i = 1,2,...,n_{hk}\} \tag{7}$$

$$R_{hk_1} \bigcap R_{hk_2} = \phi \qquad \forall hk_1 \ne hk_2 \tag{8}$$

$$sign(n_{hk}) = \begin{cases} 1 & n_{hk} \ge 1 \\ 0 & 其他 \end{cases} \tag{9}$$

$$S_{r_{hk(l-1)}} + t^{w}_{r_{hk(t-1)}} + t^{u}_{r_{hk(t-1)}} + t_{r_{hk(l-1)}r_{hkl}} = S_{r_{hkl}} \tag{10}$$
$$i = 1,2,\ldots,n_{hk}; h = 1,2,\ldots,H$$

Soft Time Window:

$$t^{w}_{hi} = \max\{a_{hi} - s_{hi}, 0\}, i = 1,2,\ldots,L_h; h = 1,2,\ldots H \tag{11}$$

$$d_{ij} = d_{ij} + c \times \max\left(a_{hj} - s_{hj}, 0\right) + d \times \max\left(s_{hj} - b_{hj}, 0\right) \tag{12}$$
$$i, j = 1,2,\ldots,L_h; h = 1,2,\ldots,H$$

$$d_{hi} = d_{hi} + c \times \max\left(a_{hi} - s_{hi}\right) + d \times \max\left(s_{hi} - b_{hi}, 0\right) \tag{13}$$
$$i = 1,2,\ldots,L_h; h = 1,2,\ldots,H$$

Hard Time Window:

$$a_{hi} \le s_{hi} \le b_{hi}, i = 1,2,\ldots,L_h; h = 1,2,\ldots,H \tag{14}$$

$$t^{w}_{hi} = 0, i = 1,2,\ldots,L_h; h = 1,2,\ldots,H \tag{15}$$

IV. THE HYBRID METAHEURISTIC ALGORITHM

The methodology of solution consists of two steps; the first solves the Routing and Loading tasks with the heuristics algorithms ACS and a local search improvement algorithm. The ACS algorithm uses local search improvement algorithm to distribute the products in the vehicle during the construction of the routes. The second step solves the scheduling task.

A. A Local Search Improvement Algorithm

The ACS algorithm designed to solve the Routing task is a distributed metaheuristic based on the approach shown in [3] and its goal is to find the minimum number of vehicles needed to satisfy all the customer demands in a Logistics Routing-Loading Scheduling Problem instance.

As the neighborhood structure accounts for solutions generated from the best available set of routes by reordering nodes on each individual tour or relocating customers to neighboring trips, there is no sense in tackling the whole VRPTW problem at once. In a local search environment, each tour just exchanges nodes with a few other routes closed to it and the attention should therefore be focused on a much smaller geographical area where such interacting trips are confined. In order to take advantage of such a problem feature, a Rotating Angular Sector (RAS) is defined with origin at the central depot (CD) and delimiting rays emanating from the CD with angular coordinates Ω_1 and, Ω_2 respectively (see Fig. 2). At each RAS-location, sub problems I and II will be repeatedly solved until the procedure converges to a local optimum (the normal mode). It may happen that no improvement at all has been achieved through the normal node after sweeping the N locations, i.e. the whole service area. In order to avoid getting stuck on a local optimum, sub problem III will be activated (the perturbation or mixed mode), if necessary, just on the next turn to move forward towards a feasible/infeasible solution with a better value of the objective function. If the perturbation move is successful, then the normal mode is applied again. The procedure is repeated until the normal mode becomes trapped on a local optimum and the perturbation mode fails to get an improved solution. When this happens, the procedure is stopped and the best set of routes is given by the current incumbent solution.

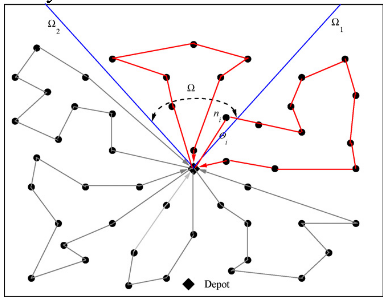

Figure 2. The rotating angular sector (RAS) decomposing the service region into smaller zones.

B. *Ant Colony System (ACS) Algorithm*

The ACS algorithm designed to solve the Routing task is a distributed metaheuristic based on the approach shown in [3] and its goal is to find the minimum number of vehicles needed to satisfy all the customer demands in a Logistics Routing-Loading Scheduling Problem instance.

(i)Routes building: at each iteration of ACO each ant builds a solution for the CVRP, moving to next client (state in the general ACO scheme) according to transition rules based on a combination of the amount of pheromone at each arc and length of it. The role of Tabu list mentioned at previous section is taken here by a set of already visited neighbors, forbidden for the ant at current iteration. We implemented two versions of the order in which routes are determined [8-13]:

(a)Sequential: each ant start its solution determining the route for the first vehicle till its capacity is complete. Then it continues with others vehicles till complete the solution. Each ant starts its solution from a different client.

(b) Parallel: each ant designs the route for all vehicles at the same time. At each iteration of the algorithm only one client is chosen, according to transition rule. Then best tour is extended.

(ii) Transition rules:

(a)Random-Proportional rule: a neighbor client is randomly chosen according to probability $P_k(i,j)$ calculated as described in [3].

(b) Pseudo-Random-Proportional rule: this rule for choosing next client to visit combines random selection with best option. Let such that $0 \le q_0 \le 1$, we generate q a random number in [0, 1], then next client j is chosen according to:

$$j = \begin{cases} \max_{u\in\Gamma(i)} \left[\tau(i,U)^{\alpha}\right]\left[\eta(i,u)\right]^{\beta} & if q \le q_0 \\ J & if q \ge q_0 \end{cases} \tag{16}$$

where j is randomly selected according to $P_k(i,j)$.

(iii) Pheromone actualization: we tried the following alternatives that appear at the ant colony literature:

(a) Global actualization: is done after each ACO iteration is completed.

All the solutions: is the one proposed in the original version of the algorithm, where pheromone levels are actualized at each iteration after all ants complete their routes.

Elite Ants: only ants that obtained the best solutions are take into account for the actualization:

$$\tau(i,j) = \varphi \cdot \tau(i,j) + \sum_{\mu}^{\sigma-1} \Delta\tau_{\mu} + \sigma \cdot \Delta\tau^{*}(i,j) \tag{17}$$

where φ is the factor de pheromone persistence and σ is the number of elite ants,

$$\Delta\tau_{\mu}(i,j) = \begin{cases} \frac{(\sigma-\mu)}{L\mu} & if\ (i,j)\ is\ part of\ a\ solution of\ the \mu-bestan \\ 0 & otherwise \end{cases} \tag{18}$$

where Lμ is the solution of the μ-best ant, and

$$\Delta\tau^{*}(i,j) = \begin{cases} \frac{1}{L^{*}} & if\ (i,j)\ is\ part\ of\ the\ best\ slu \\ 0 & otherwise \end{cases} \tag{19}$$

where L^* is the value of best solution.

Best solution: pheromone is updated using only information of best solution of the previous iteration.

(b) Local actualization: It is done each time an ant moves from one client i to the next j to decrease the amount of pheromone of an used edge (i,j), in order to diversify solutions obtained by the ants.

$$\tau(i,j) = \varphi \cdot \tau(i,j) + (1-\varphi)\Delta\tau(i,j) \tag{20}$$

Several ways to determine $\Delta\tau(i,j)$ where tested:

Q-learning: inspired in Q-learning, a method for automatic learning, we define:

$\Delta\tau_k(i,j) = \gamma \cdot \max_{z\in\Gamma(j)} \tau(j,z), (0 \le \gamma < 1)$

Initial pheromone: we use $\Delta\tau(i,j) = \tau_0$ where is τ_0 the initial pheromone level.

Evaporation: $\Delta\tau(i,j) = 0$, that is $\Delta\tau(i,j) = \varphi \cdot \tau(i,j)$

(iv) Reduced neighbor list: when the problem is too big to explore all the potential moves of the ant, a reduced list of best candidates is used.

(v) Stopping rules: ACA procedure stops when there is not improvement on the solution after several iterations or when nmax number of iterations is reached.

The constructive procedure new ant solution builds the new solution with the ant *k*. It consists in building routes using t vehicles to satisfy customer demands.

V. Experimental Results

In this experiment, a version of the hybrid metaheuristic Algorithm that solves MVRPTW with other scientific community heuristics that solve the same variant was compared. During the experiment, 100 MVRPTW instances of the Solomon's benchmark were solved. The cases were composed by the six different problem types (which are R1, R2, C1, C2, RC1, and RC2). Each instance contains 100 locations.

The basic algorithm was implemented in C#, and each instance was solved 30 times. The experiments were executed under the next conditions: Xeon Processor at 3.06 GHz, with 3.87 Gb of RAM Memory and Windows Server 2003 as Operative System. The values for the parameters of the ACS algorithm for VRPTW were: ants used = 10, colonies = 5; generations = 40; $q_0 = 0.9$; $\beta = 1$; $\rho = 0.1$.

Table 1 shows the comparison between the ACS Algorithm and four of the best known algorithms that have solved MVRPTW. Columns two to six show the average number of vehicles used in the solutions by each kind of instance. The last column is the combination of the average number of vehicles shown in the results of a whole set of instances. The considered algorithms were: Reactive Variable Neighborhood Search [9], Hybrid Genetic Algorithm [10], Genetic Algorithm [11], and Multi-Objective Ant Colony System [3].

TABLE I. FOUR OF THE BEST KNOWN ALGORITHMS COMPARED WITH ACS VRPTW WITH RESPECT TO DISTANCE

Algorithm	**Total of Distance traveled**						**Average**
	R1	***R1***	***R1***	***R1***	***R1***	***R1***	
This Algorithm	1221	954	829	590	1383	1129	1018
[3]	1217	967	829	590	1383	1129	1019
[9]	1222	975	829	590	1390	1128	1022
[11]	1228	970	829	590	1392	1144	1025
[10]	1251	1056	829	590	1414	1258	1066

The results reported by the hybrid metaheuristic algorithm in Table 1 and 2 are based on an average taken from cases that were runned 50 times, while the rest of the values shown by the other algorithms belong to the best results found in all their runnings. The other researcher experiments are different in some aspects like: the number of times that a case is runned, the running time, the equipment where the instances were executed and the way that the results are reported. The researchers report their results based in a sample which experimental designs do not show the real behavior of an algorithm [12]. Due to this, it is not possible to compare the actual results of the ACS algorithm; however, it could be appreciated that the implementation used in this research is efficient. Due to this performance it was decided to implement the ACS algorithm in such a way that it could be used to solve more complex real-world instances, like the instances.

TABLE II. INSTANCE DESCRIPTION

Instance	**Description of the Instances**						
	Customers	***Vehicles***	***Depots***	***Arcs***	***Cus cap***	***Dep cap***	***Cus dem***
Case01	7	8	10	10	1	2	1268
Case02	6	8	10	10	1	2	1092
Case03	7	8	10	10	1	2	1502
Case04	7	8	10	10	1	2	1260
Case05	7	8	10	10	1	2	1468

VI. CONCLUSIONS

Routing and scheduling of deliveries are two crucial operational decisions in logistics distribution management. Better routing and scheduling can result in shorter delivery distance, or time, and thus, higher level of efficiency and lower delivery cost can be achieved. The VRP is used prevalently to aid the planning of these two decisions.

In this paper, the MDVRP was studied because the umber of depots is not limited to one in many real-world situations. Besides routing and scheduling, the grouping problem is also considered in the MDVRP. Because the DVRP integrates three hard optimization problems, a hybrid metaheuristic Algorithm was developed. A representative result and the analysis are given. The experiment indicates the validity of the technique to MDVSPTW with the above-mentioned conditions.

ACKNOWLEDGMENT

This work is partially support by The 11th Five Years Key Programs for Science and Technology Development of China (No.2008BADA8B03, No. 2006BAD08B01), Henan Program for New Century Excellent Talents in University (No. 2006HANCET-15).

REFERENCES

[1] Toth, P., Vigo, D., “The Vehicle Routing Problem. Monographs on Discrete Mathematics and Applications”, SIAM, Philadelphia (2001)

[2] Blasum, U., Hochst¨atter, W., “Application of the Branch and Cut Method to the Vehicle Routing Problem”, Universit¨at zu K¨oln, BTU Cotbus (May15, 2002)

[3] Gambardella, L., Taillar, E., Agazzi, G., “ MACS-VRPTW: A Multiple Ant Colony System for Vehicle Routing Problems with Time Windows”, Technical Report IDSIA.IDSIA-06-99. Lugano, Switzerland (1999)

[4] Mingozzi, A., Vallet, A., “An exact Algorithm for Period and Multi-Depot Vehicle Routing Problems”, Department of Mathematics, University of Bologna, Bolonga,Italy (2003)

[5] Archetti, C., Mansini, R., Speranza, M., “The Vehicle Routing Problem with capacity 2 and 3, General Distances and Multiple Customer visits”, In: ORP 2001, Paris(September 26-29, 2001)

[6] Zhang, Q W, Zhen ,T, Zhu, Y H. et al, “A Hybrid Intelligent Algorithm for the Vehicle Routing with Time Windows” , Progress in 2008 International Conference on Intelligent Computing (ICIC2008), LNCS 5226. Berlin: Springer-Verlag, 2008.pp. 47-54, doi: 10.1007/978-3-540-87442-3_7

[7] Psinger, D., Ropke, S., “ A general Heuristic for Vehicle Routing Problems”, DIKU, University of Copenhagen (February 25, 2005)

[8] Br¨aysy, O., “A Reactive Variable Neighborhood Search Algorithm for the VehicleRouting Problem with Time Windows”, Working paper, SINTEF Applied Mathematics,Department of Optimization, Norway (2001)

[9] Zhen, T, Zhang, Q W, Zhang,W S,“Hybrid Ant Colony Algorithm for the Vehicle Routing with Time Windows”, Progress in 2008 ISECS International Colloquium on Computing, Communication, Control, and Management(CCCM2008), IEEE Computer Society Press, 2008. Volume 1,pp.8-12, doi:10.1109/CCCM.2008.85.

[10] Wu T H, Low C, Bai J W., “Heuristic solutions to multi-depot location-routing problem”, Computers & Operations Research, 29 (10), (2002), pp.1393-1415

[11] Hung,K-S, Su, S-F, LeeZ-J., “Improving ant colony optimization algorithms for solving traveling salesman problems”, J Adv Comput Intell Inform 11(4), JACIII (2007), pp.433–442

[12] Nagy G, Salhi S., “Heuristic algorithms for the single and multiple depot vehicle routing problems with pickups and deliveries”,European Journal of Operational Research, 162(2005) pp.126-141