

A PROMOTED HYBRID HEURISTIC ALGORITHM FOR TWO-DIMENSIONAL MULTI-DEPOTS VEHICLE ROUTING PROBLEM

Zhu, X. N.*; Yan, R.** & Zhang, Q.*

* School of Economics and Management, University of Science and Technology Beijing, Beijing 100083, China

** School of Economics and Management, Beijing Information Science & Technology University, Beijing 100192, China

E-Mail: zhuxiaoning@ustb.edu.cn, yr1900@163.com

Abstract

In this paper, a multi-depots capacitated vehicle routing problem where client demand is composed of two-dimensional weighted items (2L-MDCVRP) is addressed. The problem calls for the minimization of the cost of transportation needed for the delivery of the goods demanded by the clients, and carried out by a fleet of vehicles based at several depots. The overall problem, denoted as 2L-MDCVRP, is NP-hard and it is very difficult to get a good performance solution in practice. We propose a quantum-behaved particle swarm optimization (QPSO) and exploration heuristic local search algorithm (EHLA). It has been proved that particle swarm optimization is a very efficient approach for the CVRP, and the quantum-behaved can help particle swarm optimization to escape effectively from local optimum. Furthermore, we propose an exploration heuristic local search to solve the loading constraints in 2L-MDCVRP. The effectiveness of the proposed algorithm is demonstrated through computational experiments on benchmark instances.

(Received, processed and accepted by the Chinese Representative Office.)

Key Words: Vehicle Routing, Multi-Depots, Two-Dimensional Packing, Quantum-Behaved Particle Swarm Optimization, Exploration Heuristic Local Search

1. INTRODUCTION

Both vehicle routing problem (VRP) and bin packing problem (BPP) are important, typical combinatorial optimization problems, and are frequently and independently studied in the past few decades. To resolve the VRP or BPP, numerous methodologies based on operations research and mathematical programming techniques or heuristic-based algorithms have been proposed in extant literature [1-4]. In recent years, some attention has been devoted to their combined optimization, called 2L-CVRP (two-dimensional capacitated vehicle routing problem), which is a combination of two important NP-hard optimization problems in distribution logistics. In the 2L-CVRP, clients demand are defined as sets of rectangular weighted items, while vehicles have a weight capacity and a rectangular two-dimensional loading surface. The objective of the 2L-CVRP is to serve all the customers and load the corresponding items into the vehicles, through a road network, with minimum total cost.

The 2L-CVRP is particularly important in both theory and practice. The logistics managers need to deal with both the packing and routing problems simultaneously. In addition, numerous real life applications involve the delivery of rectangular-shaped items that cannot be stacked one on the top of the other because of their fragility, weight or large height of freight such as household appliances, delicate pieces of furniture, etc. The 2L-CVRP was first proposed by Iori et al. [4]. For larger scale problems, Gendreau et al. propose a Tabu Search algorithm, in which the loading component of the problem is solved through heuristics, lower bounds and a truncated branch-and-bound procedure [5]. Zachariadis et al. [6] propose a metaheuristic algorithm which incorporates the rationale of Tabu Search and Guided Local

Search based on the research of Gendreau et al. [5], employing a guiding mechanism which can drastically diversifies the search conducted by trying to eliminate low-quality features from the final solution to control the objective of the problem. The Extended Guided Tabu Search (EGTS) and simulation annealing (SA), which introduce a new scoring-based heuristic to improve packing was proposed by Leung et al. [7]. For a population-based algorithm, the Ant Colony Optimization (ACO) was employed by Fuellerer et al. [8], and the performance was proven to be quite satisfactory. The Greedy Randomized Adaptive Search Procedure combined with Evolutionary Local Search (GRASP×ELS) algorithm was proposed by Duhamel et al. [9], and this algorithm outperforms all previous methods and obtains new better solutions for several instances. However, only the unrestricted 2L-CVRP problems were solved. Recently, Zachariadis et al. [10] proposed an innovative compact meta-heuristic, named as promise routing-memory packing (PRMP), which obtained excellent performance and improved best known solutions for many instances. Lately, Wei et al. [11] proposed a variable neighbourhood search to address the routing aspect, and adopt a skyline heuristic to examine the loading constraints. The effectiveness of their approach is verified through experiments on widely used benchmark instances involving two distinct versions of loading constraints (unrestricted and sequential 2L-CVRP problem).

The 2L-MDCVRP (multi-depots capacitated vehicle routing problem where client demand is composed of two-dimensional weighted items) is a new variant of the 2L-CVRP which considers vehicles start from several different depots, which is more tally with the actual situation comparing with the one depot assumption in 2L-CVPR. However, the issue of 2L-MDCVRP is seldom discussed during the existing research by far, which is the primary motivation of this research.

The reminder of the paper is organized as follows. The next section introduces the related literature about green supply chain. Section 2 describes the problem in detail. Section 3 presents the mathematic model of the 2L-CVRP. Following is the introduction of the proposed hybrid algorithm for solving the 2L-MDCVRP. Section 5 presents the extensive computational results on benchmark instances and comparisons with other algorithms. Finally, the conclusion of the 2L-MDCVRP is discussed along with the related managerial implications.

2. PROBLEM DESCRIPTION

According to the description of 2L-CVRP, the 2L-MDCVRP may be defined as follows. We are given a complete undirected graph $G=(V,E)$, where $V=V_n \cup V_T$ is a set of $n+T$ vertices corresponding to the depots ($V_T = \{1, \dots, T\}$) and the clients ($V_n = \{1, \dots, n\}$), and $E = \{e_{i' i} | i, i' = 1, \dots, n+T, i \neq i'\}$ is the set of edges between any pair of vertices. Associated to each edge e_{ij} there is a cost c_{ij} , which is the cost of traveling from i to i' . A set of K identical vehicles is available at each depot. Each vehicle has a weight capacity and a rectangular loading surface that is accessible from a single side for loading and unloading operations, whose length and width are equal to W and L , respectively. We also denote by $S=L \times W$ the total area of the loading surface.

Each client i ($i = 1, \dots, n$) demands a set of m_i rectangular items of total weight d_i . Each item I_{ik} ($i = 1, \dots, n, k = 1, \dots, m_i$) has a weight d_{ik} , length l_{ik} and width w_{ik} . In addition, we denote by $s_i = \sum_{k=1}^{m_i} s_{ik}$ the total area and $d_i = \sum_{k=1}^{m_i} d_{ik}$ the total weight of the items of client i .

The aim of 2L-MDCVRP is to assign clients $i = 1, \dots, n$ to one of the routes R_{tk} ($t = 1, \dots, m, k = 1, \dots, K$), with the minimum total transportation cost, and there is a feasible loading of items on vehicles' loading surfaces for each route. In the 2L-MDCVRP, the routes and the feasible loading must meet the following constraint conditions:

- Each route must start and end at the same depot; different routes can start and end at the different depots.
- Each client must be served by a single vehicle to avoid split deliveries.
- Each client must be visited once and only once.
- The total weight of the clients' items covered by a route cannot exceed the capacity of the vehicle.
- All items of each client must be completely loaded on the surface of the vehicle.
- All items have a fixed orientation and can be rotated by 90 degrees.
- All items must be loaded with their sides parallel to the sides of the loading surface.
- Each two items cannot overlap in the vehicle.

There are mainly two types of 2L-MDCVRP: unrestricted 2L-MDCVRP and sequential 2L-MDCVRP. The former one is defined by the eight constraints mentioned above, while for the sequential 2L-MDCVRP, a sequence constraint is added: when a client i is visited, all items of client i must be directly unloaded with movements parallel to the length dimension of the loading surface of the vehicle, from the open edge [6]. In other words, other items in the vehicle cannot be moved when items of one client are being unloaded. Fig. 1 illustrates the 2L-MDCVRP with two depots. Suppose that there are two routes (depot1 – 1 – 2 – 3 – 4 – 5 – depot1, depot2 – 6 – 7 – 8 – 9 – depot2) with 21 items demanded by nine clients. Loading I and Loading II are two different loading solutions depicted in Fig. 2 a and b, respectively. I_{ik} is item k of client i , and Loading I is unrestricted 2L-MDCVRP, while in Loading II, all items can be unloaded straight away, with movement parallel to the length dimension, so it is the sequential 2L-MDCVRP.

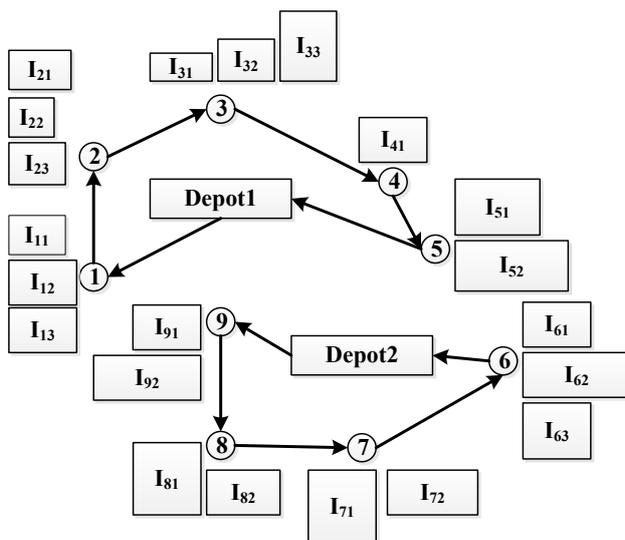


Figure 1: The demonstration of 2L-MDCVRP.

3. MATHEMATIC MODEL

It is assumed that each item in the vehicle is packed in a rectangular box, and each vehicle is regard as a rectangular container. We assume the dimensions of the boxes are integer, and they can only be placed orthogonally into the container with the fixed orientation.

A Cartesian coordinate system is adopted with its origin in the container's front-left corner, and let (x, y) be the possible coordinates where the front-left corner of a box can be placed. These positions along axes L and W of the container belong to the sets: $X = \{0, 1, \dots, L - \min(l_{ik})\}$ and $Y = \{0, 1, \dots, W - \min(w_{ik})\}$, respectively. So the container can be divided into many grids, as shown in Fig. 3.

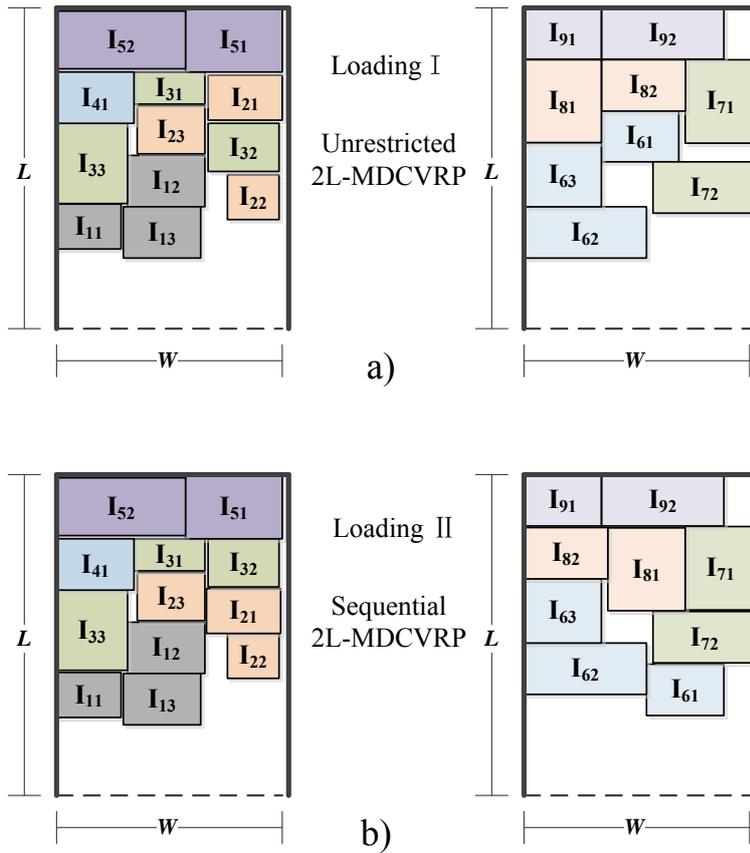


Figure 2: The unrestricted and sequential 2L-MDCVRP.

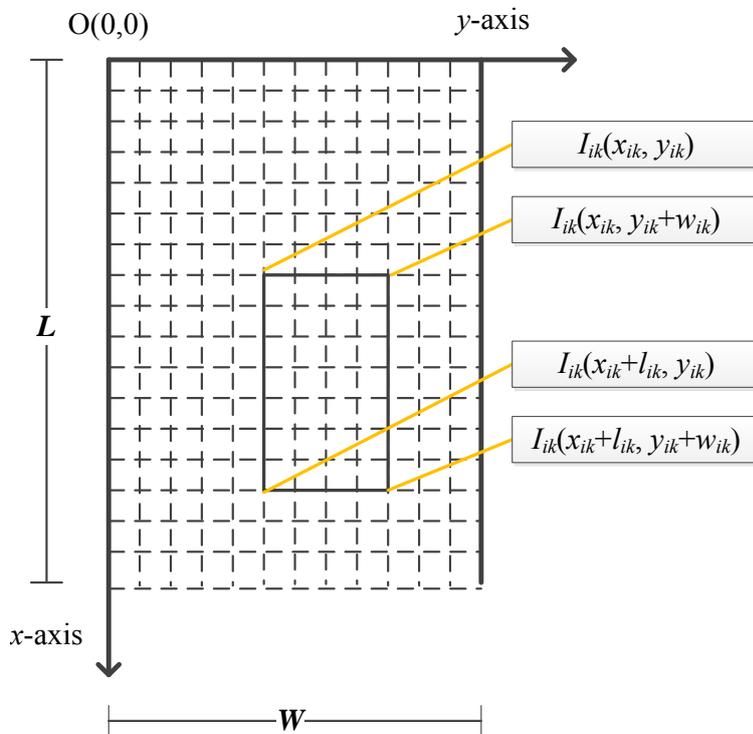


Figure 3: The Cartesian coordinate system and grid graph.

All the symbols used in the model are shown in Table I.

Table I: The symbols of the model.

Symbols	Explanation of symbols
V_T	set of depots, index $t=1, 2, \dots, T$
V_n	set of clients, index $i=1, 2, \dots, n$
P_t	set of vehicles of depot t , index $p=1, 2, \dots, v_t$, $v=v_1 + \dots + v_T$ is total number of vehicles
D	weight capacity of each vehicle
S	loading area of each vehicle
L	length of each vehicle
W	width of each vehicle
x_{ik}	x-axis position of the front-left of item I_{ik}
y_{ik}	y-axis position of the front-left of item I_{ik}
$c_{ii'}$	the distance between vertices i and i' (including clients and depots)
I_{ik}	item k of client i
m_i	items number demanded by client i
d_i	weight of items of client i
d_{ik}	weight of item I_{ik}
S_i	total area of items of client i
S_{ik}	area of item I_{ik} ,
l_{ik}	length of item I_{ik}
w_{ik}	width of item I_{ik}

The decision variable of the model is defined as following:

$$a_{ii}^{tp} = \begin{cases} 1, & \text{the vehicle } p \text{ of depot } t \text{ travel from node } i \text{ to node } i', \\ 0, & \text{others,} \end{cases}$$

$$b_i^{tp} = \begin{cases} 1, & \text{client } i \text{ is served by vehicle } p \text{ of depot } t, \\ 0, & \text{others,} \end{cases}$$

where tp means the vehicle p of depot t ($i, i'=1, \dots, n+T, t=1, \dots, T, p=1, 2, \dots, v_t$).

The mathematical formulation of model is shown as:

$$\min Z = \sum_{t=1}^T \sum_{p=1}^P \sum_{i=1}^{n+T} \sum_{i'=1}^{n+T} c_{ii'} \cdot a_{ii'}^{tp} \tag{1}$$

$$\text{s.t. } \sum_{t=1}^T \sum_{p=1}^P b_i^{tp} = 1, \quad i=1, \dots, n \tag{2}$$

$$\sum_{i=1}^{n+T} a_{ii}^{tp} = \sum_{i=1}^{n+T} a_{ii}^{tp}, \quad t=1, 2, \dots, T, p=1, 2, \dots, v_t \tag{3}$$

$$\sum_{i=1}^{n+T} a_{ii}^{tp} = b_i^{tp}, \quad i=1, \dots, n, t=1, 2, \dots, T, p=1, 2, \dots, v_t \tag{4}$$

$$\sum_{i=1}^{n+T} a_{ii}^{tp} = b_i^{tp}, \quad i=1, \dots, n, t=1, 2, \dots, T, p=1, 2, \dots, v_t \tag{5}$$

$$\sum_{i=1}^n b_i^{tp} \cdot d_i \leq D, \quad t=1, 2, \dots, T, p=1, 2, \dots, v_t \tag{6}$$

$$\sum_{i=1}^n b_i^{tp} \cdot s_i \leq S, t=1, 2, \dots, T, p=1, 2, \dots, v_t \quad (7)$$

$$0 \leq x_{ik} \leq W - w_{ik}, k=1, 2, \dots, m_i, i=1, 2, \dots, n \quad (8)$$

$$0 \leq y_{ik} \leq L - l_{ik}, k=1, 2, \dots, m_i, i=1, 2, \dots, n \quad (9)$$

$$x_{ik} + w_{ik} \leq x_{i'k'}, \text{ or } x_{i'k'} + w_{i'k'} \leq x_{ik}, k=1, 2, \dots, m_i, k'=1, 2, \dots, m_{i'}, i, i'=1, 2, \dots, n, i \neq i' \quad (10)$$

$$y_{ik} + l_{ik} \leq y_{i'k'}, \text{ or } y_{i'k'} + l_{i'k'} \leq y_{ik}, k=1, 2, \dots, m_i, k'=1, 2, \dots, m_{i'}, i, i'=1, 2, \dots, n, i \neq i' \quad (11)$$

$$\text{if } y_{ik} < y_{i'k'} + l_{i'k'} < y_{ik} + l_{ik}, \text{ then } x_{i'k'} + w_{ik} \leq x_{ik}, k=1, 2, \dots, m_i, k'=1, 2, \dots, m_{i'}, i, i'=1, 2, \dots, n, i \neq i' \quad (12)$$

Eq. (1) is the objective function, aims to minimize the total travel cost of all vehicles. Eqs. (2) to (5) are the loop constraints, where eq. (2) means each client just can be visit once by one vehicle, and eqs. (3) to (5) mean if a vehicle into a node then must leave from the node. Eqs. (6) and (7) are the capacity constraints, that limit the total weight and area of the clients' items covered by a route must not exceed the capacity and space of the vehicle. Eqs. (8) to (11) are loading constraints, where eqs. (8) and (9) limit the edges of boxes cannot exceed containers, and eqs. (10) and (11) avoid the overlapping of the boxes packed. Eq. (12) is sequential constrain, avoid to move items of other clients when serve one client. In case of unrestricted 2L-MDCVRP, Eq. (12) should be removed.

For the remaining part of this paper, we will use the term base model to refer to the model comprised by eqs. (1) to (12). Note that, in the computational experiments of Section 5, it was not possible to compile some of the models due to the computer memory required to pre-compute; however, these same models could be compiled and solved. It is also worth noting that the base model can be easily extended to consider special cases of (i) rotation of the boxes.

4. HYBRID ALGORITHM

The new hybrid algorithm for solving the 2L-MDCVRP consists of a quantum-behaved particle swarm optimization (QPSO) for routing vehicles and an exploration heuristic local search algorithm (EHLA) for loading the boxes of all clients of a route into the loading space of a vehicle. The proposed QPSO algorithm for the solution of the MDCVRP is given in section 4.1. Then a new decoding method and an initial solution generation are proposed for improving the solution. And the ELSA for the solution of the 2-dimensional loading problem is given in section 4.2. Finally, the hybrid algorithm combine by QPSO and EHLA is given in section 4.3.

4.1 Quantum-behaved particle swarm optimization

Particle swarm optimization (PSO) is a population-based searching method first proposed by Kennedy and Eberhart [12] who were inspired from the behaviour of natural group organisms (i.e. bees, fishes, and birds swarm). Then some scholars applied variant PSO to solve CVRP, and get good results [13-15]. In standard PSO algorithm, each particle is an individual and the swarm is composed of particles that the position of each particle is considered as a solution for the problem. Each particle leaves its position to a better position in the multi-dimensional search space with a specific velocity. The positions of particles are generated randomly and the velocity of particles is initialized by zero in the first iteration. The position and velocity of particles are updated by the following formulas in other iterations:

$$V_{i,j}(t+1) = w \cdot V_{i,j}(t) + c_1 \cdot r_{1,i}(t) \cdot [P_{i,j}(t) - X_{i,j}(t)] + c_2 \cdot r_{2,i}(t) \cdot [P_{g,j}(t) - X_{i,j}(t)] \quad (13)$$

$$X_{i,j}(t+1) = X_{i,j}(t) + V_{i,j}(t+1) \quad (14)$$

Liu et al. studied the research results of the particle convergence behaviour, and put forward a quantum-behaved particle swarm optimization algorithm (QPSO) based on the basic convergence characteristic of PSO algorithm [16]. QPSO algorithm improves the search strategy with fewer parameters since the evolutionary equation does not need velocity vector.

According to the literature [17], on the convergence analysis of the PSO algorithm, to ensure the convergence of the algorithm, each particle must converge to the particle's attractor C_i , attractor is as follows:

$$C_{i,j}(t) = (c_1 \cdot r_1 \cdot P_{i,j}(t) + c_2 \cdot r_2 \cdot P_{g,j}(t)) / (c_1 \cdot r_1 + c_2 \cdot r_2), \quad j = 1, 2, \dots, d, \quad c_1, c_2 \in U(0, 1) \quad (15)$$

$$C_{i,j}(t) = \alpha \cdot P_{i,j}(t) + (1 - \alpha) \cdot P_{g,j}(t), \quad \alpha \in U(0, 1) \quad (16)$$

P_i is the personal best particle and P_g is the global best particle, where $\alpha = (c_1 \cdot r_1) / (c_1 \cdot r_1 + c_2 \cdot r_2)$. It can be seen that the local attractor is a stochastic attractor of particle i that lies in a hyper-rectangle with P_i and P_g being two ends of its diagonal.

According to the literature [18], in quantum time-space framework, the quantum state of a particle is depicted by a wave function $\psi(\vec{x}, t)$. Then a global point called Mainstream Thought or mean best position of the population is introduced into QPSO [19]. The global point, denoted as $mbest$, is defined as the mean of the $Pbest$ positions of all particles. That is:

$$mbest(t) = (mbest_1(t), mbest_2(t), \dots, mbest_d(t)) = \left(\frac{1}{n} \sum_{i=1}^n P_{i,1}(t), \frac{1}{n} \sum_{i=1}^n P_{i,2}(t), \dots, \frac{1}{n} \sum_{i=1}^n P_{i,d}(t) \right) \quad (17)$$

Then, the new position of a particle can be calculated by:

$$X_{i,j}(t+1) = C_{i,j}(t) \pm \beta \cdot |mbest_j - X_{i,j}(t)| \cdot \ln(1/u), \quad u = rand(0, 1) \quad (18)$$

where parameter β is called contraction-expansion coefficient, which can be tuned to control the convergence speed of the algorithms.

The QPSO framework for solving the proposed problem is structured in four steps as follows: In the first step of algorithm the position of the particles are initialized randomly and P_i of each particle considered equal to initial position of particles also the velocity of particles are initialized to the zero. In the second step the objective function of particles is calculated, using decoding method, then P_i and P_g are calculated according to the objective function of particles, P_g is set to the best P_i among all of population. In third step the velocity and position of each particle are updated using eqs. (16) and (18) respectively.

Decoding method

It is assumed that there is more than one depot in the model, so a new particle encoding is designed with $3 \times n$ matrix to show the relations among the T depots, v vehicles and n clients (Table II). The first row X_d in the matrix denotes the depots, the second row X_v denotes the vehicles and the third row X_t indicates the distances of the vehicles.

Table II: Decoding example.

Client	1	2	3	4	5	6	7	8	9	10
X_d	1	1	2	1	2	2	2	1	2	2
X_v	1	1	3	2	4	4	4	2	5	5
X_t	0.3	1.2	0.6	2.3	4.8	1.1	2.9	3.9	2.8	1.7

For example, consider a case with two depots and ten clients. Depot1 has two vehicles, and depot2 has three vehicles, then the decoding is shown in Fig. 4.

X_t needs to be adjusted to the routes instead of distances according to the value of X_t , which means that we need to identify all the clients served by each vehicle, and then sort by the value of X_t to identify the route of each vehicle. In this way, the client can be served once and only once, so it can reduce the time of adjustment process of the feasible solution. The adjusted location vectors and the routes are shown in Fig. 4.

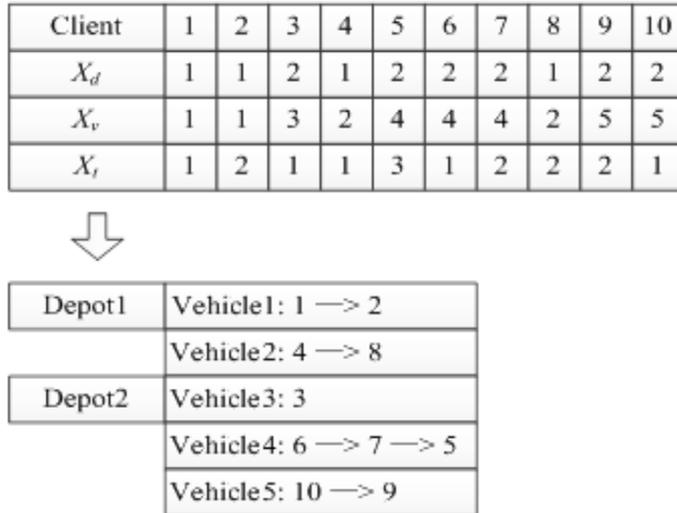


Figure 4: The decoding process.

Local search heuristics

In this subsection, the local improvement methods are explained. According to the literature [20], 2-opt heuristic and inner-tour swap are used to improve the solution.

2-opt heuristic applies in two tours. In the first step of this neighbourhood search algorithm selected tours are separated in two sections by selecting a node from each tour. In the second step first part of first tour join to second part of second tour and vice versa.

Inner-tour swap can be applied in single tour. In the first step of this local search two clients that are assigned to the same tour are selected and then the sequences of them are replaced. For applying this local search set count 1= from 1 to tour length-1, that tour length is the number of clients in the selected tour, then for count 2= from count 1 to tour length exchange the sequence of count 1st client and count 2nd client. If this change makes a tour with less cost, accept this change and continue, else reject the change and select another pair of customer according to the values of count 1 and count 2.

4.2 Exploration heuristic search algorithm

Through the QPSO algorithm in section 4.1 we can get the tours, and identify the clients covered by the vehicles, as well as vehicles in the routes. All the items demanded by the clients need to be loaded in the vehicles with the related constrains. In fact, there are two aspects that packing algorithm needs to consider. One is to determine the next loading item, and another is to determine the feasible loading position. So in this section we will give two methods for them.

Determine the items order

For a certain tour, the visit order of all clients on this tour is fixed. For sequential model that descript in section 2, other items in the vehicle cannot be moved when items of one customer are being unloaded. So the items in a vehicle are sorted in descending order of visit, called *OV*. Then the order of items belong to different clients are fixed, and the order of items of same client are not fixed. For unrestricted model, there no need to sort items in visit order.

For items of one client in sequential model or items of all clients in unrestricted model, we create two orders O1 and O2 to determine the final order of them. For O1 and O2, items are sorted in descending order of area $l \cdot w$, and length l , respectively. O1 is prior to O2 unless the bottom areas of the two items are the same. The orders of the items loading for the example of section2 can be identified according to the above rules:

route1: $I_{52}-I_{51}-I_{41}-I_{33}-I_{32}-I_{31}-I_{21}-I_{23}-I_{22}-I_{12}-I_{13}-I_{11}$,

route2: $I_{62}-I_{63}-I_{61}-I_{71}-I_{72}-I_{81}-I_{82}-I_{92}-I_{91}$.

Determine the feasible loading position

A new heuristic rule in terms of the sorting of the loading positions is proposed in this paper. The first selected item is located in the front-left corner in the vehicle, which is O(0, 0) in Fig. 5. It is important to note that the position of the item refers to the coordinates of its front-left. For example, if the item A with the length of 3 and the width of 2 is loaded in point O, then the space of (0, 0), (0, 3), (2, 0) and (2, 3) is occupied as shown in Fig. 5, and the new position list includes $a(0, W, L-l_A)$, $b(1, W-1, L-l_A)$, $c(2, W-2, L)$, after A is loaded according to the rule that the items should be near to the edges of the vehicle or other items.

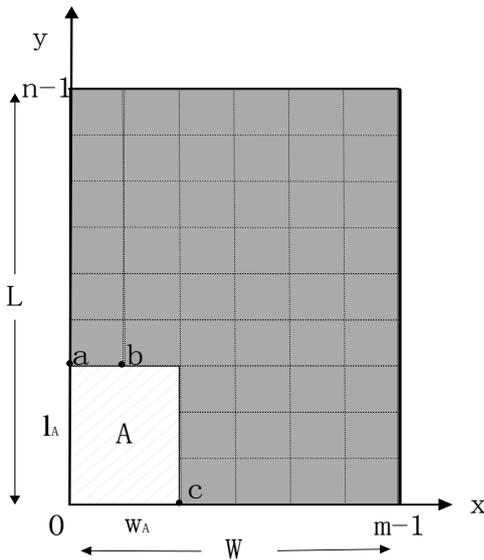


Figure 5: Feasible loading position.

Then we need to identify the most ideal position for the current item loading position among all the feasible loading positions to minimize the waste of the space, so the sorting rule for the feasible loading positions are as follows:

- a) both of the length and width of the item equal to the feasible loading position, then the fitness is 1, as shown in Fig. 6 a.
- b) the width of the item fit the feasible loading position, while the length of the item is less than the length of the feasible loading space, then the fitness is 2, as shown in Fig. 6 b.
- c) the length of the item fit the feasible loading position, while the width of the item is less than the width of the feasible loading space, then the fitness is 3, as shown in Fig. 6 c.
- d) both of the length and width of the item are less than the feasible loading position, then the fitness is 4, as shown in Fig. 6 d.
- e) either the length or the width of the item is larger than the feasible loading position, then the fitness is infinite.

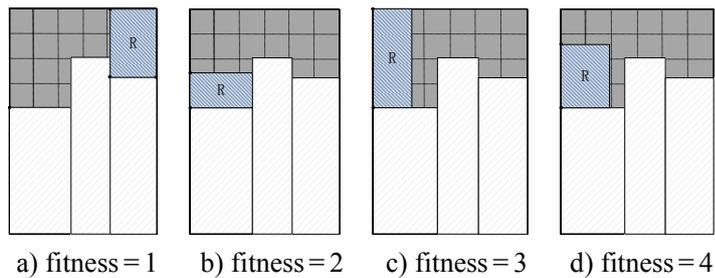


Figure 6: Fitness of loading position.

5. COMPUTATIONAL EXPERIMENTS

Gendreau et al. [5] generated the 180 2L-CVRP instances by extending the 36 well-known classical CVRP instances introduced by Toth and Vigo (Eds.) [21]. 30 of these 2L-CVRP instances are selected randomly, and loading surface (L, W) is fixed as (40, 20) for all instances. We compare our results with the best known solutions of Duhamel. As shown in Table III, “Best” refers to the best solution, “Time” (seconds) records the average CPU time to find the best solution, and “% Gap” refers to the percentage improvement between our algorithm and GRASP×ELS algorithm.

The 30 selected instances are changed to two depots instances to testify the proposed algorithm in this paper by adding a new depot. We calculate the centre of the clients, and then select a point between the first depot and the centre randomly as the second depot.

The proposed QPSO algorithm belongs to the swarm intelligence algorithm, so there is no advantage in terms of the CPU time to find the best solution. However, the proposed method outperforms the existing GRASP×ELS best solutions in 25 out of 30 Benchmark instances, and the same results for the rest five instances. The average percentage improvement is 0.84 % for our algorithm compared with GRASP×ELS algorithm. The numerical experiment shows that the proposed QPSO×EHLSA has a good performance for 2L-MDCVRP, as well as reducing the overall distances between the depots and the clients significantly.

6. CONCLUSIONS

In this paper, 2L-MDCVRP, which a new variant of the 2L-CVRP considering vehicles start from several different depots, is addressed. We propose a quantum-behaved particle swarm optimization (QPSO) and exploration heuristic local search algorithm (EHLSA). It has been proved that particle swarm optimization is a very efficient approach for the CVRP, and the quantum-behaved can help particle swarm optimization to escape effectively from local optimum. Furthermore, we propose an exploration heuristic local search to solve the loading constraints in 2L-MDCVRP. In addition, some efficient strategies are used to accelerate the procedure. 30 of the 2L-CVRP Benchmark instances are selected randomly, and the numerical experiments show that it could get better values for most of the Benchmark instances (25 out of 30). Thus, the proposed QPSO×EHLSA has a good performance for 2L-MDCVRP, as well as reducing the overall distances between the depots and the clients significantly.

ACKNOWLEDGEMENT

The work has been funded by the Fundamental Research Funds for the Central Universities (FRF-AS-13-003A, FRF-TP-14-058A2) and Research Funds for Beijing Social Science Fund Research Project (14JDJGC018).

Table III: Comparative results with other best known solutions.

No.	GRASP×ELS		QPSO×EHLA			QPSO×EHLA		
	One		One			Two		
Depot	Best	Time (s)	Best	Time (s)	% Gap	Best	Time (s)	% Gap
0201	334.96	0.0	327.24	1.10	2.30	274.65	1.22	18.01
0202	334.96	0.0	330.12	1.20	1.44	315.73	1.34	5.74
0203	352.16	0.1	340.84	1.10	3.21	296.57	1.25	15.79
0204	334.96	0.1	320.23	1.10	4.40	258.97	1.30	22.68
0205	334.96	0.0	329.50	1.30	1.63	307.69	1.28	8.14
0601	495.85	0.0	487.26	2.20	1.73	355.31	2.56	28.34
0602	495.85	0.4	488.51	1.40	1.48	375.22	1.73	24.33
0603	498.16	0.6	492.35	2.60	1.17	452.68	3.12	9.13
0604	498.32	0.5	498.32	3.20	0.00	417.46	3.22	16.23
0605	495.85	0.0	495.85	4.10	0.00	435.98	4.56	12.07
1201	610.00	0.2	599.87	3.20	1.66	481.39	3.78	21.08
1202	610.57	5.4	601.16	4.10	1.54	559.64	4.56	8.34
1203	610.00	54.1	609.43	15.80	0.09	567.32	20.46	6.99
1204	614.23	1.6	614.23	14.30	0.00	587.47	15.21	4.36
1205	610.23	6.4	610.04	19.40	0.03	603.54	19.74	1.09
1601	698.61	0.0	697.85	7.90	0.11	585.75	8.10	16.15
1602	698.61	0.9	698.61	10.20	0.00	657.42	13.47	5.98
1603	698.61	2.9	693.24	19.20	0.77	633.78	20.43	9.27
1604	703.35	12.0	699.20	22.50	0.59	699.20	26.75	0.59
1605	698.61	8.4	694.97	30.40	0.52	457.89	34.46	34.46
2301	835.26	3391.3	833.17	2211.10	0.25	649.85	1768.45	22.19
2302	1041.04	1226.9	1041.04	3289.50	0.00	864.95	3046.45	16.91
2303	1081.48	1288.2	1071.20	2974.30	0.95	971.84	2564.94	10.14
2304	1080.02	1523.1	1078.62	3037.40	0.13	894.81	3541.69	17.15
2305	950.09	1456.3	946.20	2976.30	0.41	863.26	2564.87	9.14
3301	1301.06	4842.1	1300.10	6928.40	0.07	1135.41	4678.95	12.73
3302	2285.94	5000.4	2279.50	5993.20	0.28	1578.97	4697.64	30.93
3303	2390.58	4853.6	2383.84	6259.80	0.28	1963.17	5012.75	17.88
3304	2308.40	3908.2	2308.23	6788.40	0.01	2048.75	5987.63	11.25
3305	2046.00	4713.7	2044.57	10892.20	0.07	1764.85	9845.64	13.74
Mean					0.84			14.36

REFERENCES

- [1] Laporte, G. (1992). The vehicle routing problem: an overview of exact and approximate algorithms, *European Journal of Operational Research*, Vol. 59, No. 3, 345-358, doi:10.1016/0377-2217(92)90192-c
- [2] Toth, P.; Vigo, D. (2002). An overview of vehicle routing problems, Toth, P.; Vigo, D. (Eds.) , *The Vehicle Routing Problem*, Society for Industrial and Applied Mathematics, Philadelphia, 1-26, doi:10.1137/1.9780898718515.ch1

- [3] Iori, M. (2005). Metaheuristic algorithms for combinatorial optimization problems, *4OR*, Vol. 3, No. 2, 163-166, doi:[10.1007/s10288-005-0052-3](https://doi.org/10.1007/s10288-005-0052-3)
- [4] Iori, M.; Salazar-Gonzalez, J.-J.; Vigo, D. (2007). An exact approach for the vehicle routing problem with two-dimensional loading constraints, *Transportation Science*, Vol. 41, No. 2, 253-264, doi:[10.1287/trsc.1060.0165](https://doi.org/10.1287/trsc.1060.0165)
- [5] Gendreau, M.; Iori, M.; Laporte, G.; Martello, S. (2008). A Tabu search heuristic for the vehicle routing problem with two-dimensional loading constraints, *Networks*, Vol. 51, No. 1, 4-18, doi:[10.1002/net.20192](https://doi.org/10.1002/net.20192)
- [6] Zachariadis, E. E.; Tarantilis, C. D.; Kiranoudis, C. T. (2009). A guided Tabu search for the vehicle routing problem with two-dimensional loading constraints, *European Journal of Operational Research*, Vol. 195, No. 3, 729-743, doi:[10.1016/j.ejor.2007.05.058](https://doi.org/10.1016/j.ejor.2007.05.058)
- [7] Leung, S. C. H.; Zheng, J.; Zhang, D.; Zhou, X. (2010). Simulated annealing for the vehicle routing problem with two-dimensional loading constraints, *Flexible Services & Manufacturing Journal*, Vol. 22, No. 1-2, 61-82, doi:[10.1007/s10696-010-9061-4](https://doi.org/10.1007/s10696-010-9061-4)
- [8] Fuellerer, G.; Doerner, K. F.; Hartl, R. F.; Iori, M. (2009). Ant colony optimization for the two-dimensional loading vehicle routing problem, *Computers & Operations Research*, Vol. 36, No. 3, 655-673, doi:[10.1016/j.cor.2007.10.021](https://doi.org/10.1016/j.cor.2007.10.021)
- [9] Duhamel, C.; Lacomme, P.; Quilliot, A.; Toussaint, H. (2011). A multi-start evolutionary local search for the two-dimensional loading capacitated vehicle routing problem, *Computers & Operations Research*, Vol. 38, No. 3, 617-640, doi:[10.1016/j.cor.2010.08.017](https://doi.org/10.1016/j.cor.2010.08.017)
- [10] Zachariadis, E. E.; Tarantilis, C. D.; Kiranoudis, C. T. (2013). Integrated distribution and loading planning via a compact metaheuristic algorithm, *European Journal of Operational Research*, Vol. 228, No. 1, 56-71, doi:[10.1016/j.ejor.2013.01.040](https://doi.org/10.1016/j.ejor.2013.01.040)
- [11] Wei, L.; Zhang, Z.; Zhang, D.; Lim, A. (2015). A variable neighborhood search for the capacitated vehicle routing problem with two-dimensional loading constraints, *European Journal of Operational Research*, Vol. 243, No. 3, 798-814, doi:[10.1016/j.ejor.2014.12.048](https://doi.org/10.1016/j.ejor.2014.12.048)
- [12] Kennedy, J.; Eberhart, R. C. (2001). *Swarm Intelligence*, Morgan Kaufmann Publishers, San Francisco
- [13] Ai, T. J.; Kachitvichyanukul, V. (2009). A particle swarm optimization for the vehicle routing problem with simultaneous pickup and delivery, *Computers & Operations Research*, Vol. 36, No. 5, 1693-1702, doi:[10.1016/j.cor.2008.04.003](https://doi.org/10.1016/j.cor.2008.04.003)
- [14] Marinakis, Y.; Marinaki, M.; Dounias, G. (2010). A hybrid particle swarm optimization algorithm for the vehicle routing problem, *Engineering Applications of Artificial Intelligence*, Vol. 23, No. 4, 463-472, doi:[10.1016/j.engappai.2010.02.002](https://doi.org/10.1016/j.engappai.2010.02.002)
- [15] Wu, Z. (2014). Optimization of distribution route selection based on particle swarm algorithm, *International Journal of Simulation Modelling*, Vol. 13, No. 2, 230-242, doi:[10.2507/IJSIMM13\(2\)CO9](https://doi.org/10.2507/IJSIMM13(2)CO9)
- [16] Liu, J.; Sun, J.; Xu, W.-B. (2006). Quantum-behaved particle swarm optimization with adaptive mutation operator, Jiao, L.; Wang, L.; Gao, X.-B.; Liu, J.; Wu, F. (Eds.), *Advances in Natural Computation*, Vol. 4221, Springer-Verlag, Berlin, 959-967, doi:[10.1007/11881070_126](https://doi.org/10.1007/11881070_126)
- [17] Jin, X.; Liang, Y.-Q.; Tian, D.-P.; Zhuang, F.-Z. (2013). Particle swarm optimization using dimension selection methods, *Applied Mathematics and Computation*, Vol. 219, No. 10, 5185-5197, doi:[10.1016/j.amc.2012.11.020](https://doi.org/10.1016/j.amc.2012.11.020)
- [18] Tang, D.-Y.; Cai, Y.-M.; Zhao, J.; Xue, Y. (2014). A quantum-behaved particle swarm optimization with memetic algorithm and memory for continuous non-linear large scale problems, *Information Sciences*, Vol. 289, 162-189, doi:[10.1016/j.ins.2014.08.030](https://doi.org/10.1016/j.ins.2014.08.030)
- [19] Sun, J.; Xu, W.-B.; Feng, B. (2004). A global search strategy of quantum-behaved particle swarm optimization, *IEEE Conference on Cybernetics and Intelligent Systems*, Vol. 1, 111-116, doi:[10.1109/iccis.2004.1460396](https://doi.org/10.1109/iccis.2004.1460396)
- [20] Norouzi, N.; Sadegh-Amalnick, M.; Alinaghiyan, M. (2015). Evaluating of the particle swarm optimization in a periodic vehicle routing problem, *Measurement*, Vol. 62, 162-189, doi:[10.1016/j.measurement.2014.10.024](https://doi.org/10.1016/j.measurement.2014.10.024)
- [21] Gendreau, M.; Laporte, G.; Potvin, J.-Y. (2002). Metaheuristics for the capacitated VRP, Toth, P.; Vigo, D. (Eds.), *The Vehicle Routing Problem*, Society for Industrial and Applied Mathematics, Philadelphia, 129-154, doi:[10.1137/1.9780898718515.ch6](https://doi.org/10.1137/1.9780898718515.ch6)