International Journal of Control and Automation
Vol.7, No.3 (2014), pp.299-316
http://dx.doi.org/10.14257/ijca.2014.7.3.29

An Improved Variable Neighborhood Search Algorithm for Multi Depot Heterogeneous Vehicle Routing Problem based on Hybrid Operators

Yingcheng Xu[1,2] and Wei Jiang[3*]

[1]*Quality Management Branch, China National Institute of Standardization, Beijing 100088, China*
[2]*Department of Industrial Engineering, Tsinghua University, Beijing 100084, China*
[3]*School of Computer and Information, Anqing Normal University, Anqing 246011,*

Abstract

This paper presents an Imporved Variable Neighborhood Search (IVNS) algorithm to solve Multi Depot Heterogeneous Vehicle Routing Problem with Time Windows (MDHVRPTW). The initial solution is obtained by customer allocation and path planning based on a cost network. The hybrid operators of insert and exchange are used to achieve the shaking process and the later optimization process is presented based on hybird operators to improve the solution space, the best-improvement strategy is adopted, which make the algorithm can achieve a better balance in the solution quality and running time. The idea of simulated annealing is introduced to take control of the acceptance of new solutions. Computational results are presented in benchmark instances, which show that our approach is competitive and even outperforms existing solution procedures proposed in the literature. And finally the proposed model and algorithm is applied to the large water project in China to solve the allocation of vehicles and routes.

Keywords: *improved variable neighborhood search; multi depot; heterogeneous; vehicle routing problem*

1. Introduction

We consider the multi depot heterogeneous vehicle routing problem with time windows(MDHVRPTW), a variant of the vehicle routing problem (VRP), where the vehicles do not necessary have the same capacity and they belong to different the distribution centers or depots. Therefore, the MDHVRPTW involves designing a set of vehicle routes, each starting and ending at the depot, for a heterogeneous fleet of vehicles which services a set of customers with known demands. Each customer is visited exactly once, and the total demand of a route does not exceed the capacity of the vehicle type assigned to it. The routing cost of a vehicle is the sum of its fixed cost and a variable cost incurred proportionately to the travel distance. The objective is to minimize the total of such routing costs. The number of available vehicles of each type is assumed to be unlimited. Variable neighborhood search (VNS) was initially proposed by Mladenovic and Hansen (1997, 2001) for solving combinatorial and global optimization problems. The main reasoning of this metaheuristic is based on the idea of a systematic change of neighborhoods within a local search method. In the past few decades, many scholars used VNS to solve the different VRP, such as PVRP (Pirkwieser,

[*] Corresponding author. Tel.: +86-10-5881-1138
Email address:xuycbuaa@gmail.com (Wei Jiang)

ISSN: 2005-4297 IJCA

2008), DVRP (Gendreau, 2006), HFVRP (Imran, 2009), and made some achievements. At present, VNS has become a hot topic and it is also applied to solve other problems, these can be found in the literatures (Avanthay 2003, Ribeiro 2002, Hansen 2007, Adibi 2010).

The paper has two main contributions. First, the proposed algorithm is improved in initial solution, shaking and local search based on classical VNS, the results show the algorithm is feasible and competitive with the existing other algorithms. Second, the proposed model and algorithm is applied to the large water project in China to solve the allocation of vehicles and routes. The remainder of the paper is organized as follows: The literatures reviews are illustrated in Section 2 and the problem definition and mathematical formulation are discussed in Section 3. Section 4 introduces the main ideas of the new variable neighborhood search. Computational results are presented and discussed in Section 5. The application research with the new variable neighborhood search is shown in Section 6. Section 7 concludes the paper with a resume of the approach.

2. Literature Reviews

In the past ten decades, a tremendous amount of work in the field of vehicle routing problems has been published, especially, there are many literatures based on VNS. The Braysy (2003) gave the internal design of the VND and RVNS algorithm in detail, analyzed VRPTW problem, indicated the VND algorithm was one of the most effective ways to solve VRPTW problems. Polacek (2004) designed VNS to solve MDVRPTW, the algorithm used the neighborhood structure of swap and cross to do shaking operation for the current solution, to do local search with a constrained 3-opt operator, to accept the part of the poor solution to avoid the algorithm into a local optimum by Threshold Accepting. Kytojoki *et al.* (2007) designed the guided VNS algorithm to handle the 32 existing large scale VRP problem and compared with TS algorithm. The result showed that VNS algorithm was more effective than TS algorithm in solving time. Goel and Gruhn (2008) introduced the RVNS to solve the general VRP problem including time windows, vehicle constraints, path constraints, travel departure time constraints, capacity constraints, the order models compatibility constraints, multi-supplier point of the orders, transport and service position constraints. Hemmelmayr *et al.* (2009) proposed the VNS algorithm for periodical VRP problem, adopted the saving algorithm for the construction of the initial solution, designed the move and cross neighborhood, used 3-opt operator as local search strategies, and contrasted it with other research results. Fleszar (2009) adopted VNS algorithm to solve the open-loop VRP problem, and tested 16 benchmark problems.

The hybrid Metaheuristics is a current research focus. To integrate other Metaheuristics into VNS is called the VNS algorithm based on Metaheuristics, such as variable neighborhood search algorithm of simulated annealing (Bouffard, 2007), variable Neighborhood search algorithms of Tabu search (Liao, 2007), and genetic and variable neighborhood search algorithm (Gao, 2008). Choi and Tcha (2007) used an efficient application of column generation technique which is enhanced by dynamic programming schemes. Lee *et al.* (2008) put forward an algorithm that uses tabu search and set partitioning. Recently, Brandao (2008) developed two tabu search variants incorporating GENI and some neighborhood reductions. Dondo and Cerda (2007) developed a three phase heuristic for the multi-depot HFVRPTW, the idea is to use clustering to reduce the size of the problem which is then solved optimally. Dondo, *et al.* (2009) proposed a hybrid local improvement algorithm to solve large scale MDVRPTW. Wen *et al.* (2008) developed an improved particle swarm algorithm for solving MDVRPTW; Ting *et al.* (2008) combined the ant colony algorithm and simulated annealing algorithm to solve MDVRPTW and got ideal experimental results; Ostertag (2008) integrated the VNS and MA into POPMUSIC algorithm framework respectively, and compared the solving results of two hybrid algorithm for the massive MDVRPTW. Andrea Bettinelli *et al.* (2011) presented a branch-and-cut-and-price algorithm for the exact solution of a variation of the vehicle routing problem with time windows in which the transportation fleet is made by vehicles with different capacities and fixed costs, based at different depots. Sutapa *et al.* (2011) analyzed the underlying complexities of MDPVRPTW and presented a heuristic approach to solve the problem, in this algorithm, two modification operators namely, crossover and mutation are designed specially to solve the MDPVRPTW. Anand (2012) proposed hybrid algorithm which was composed by an Iterated Local Search (ILS) based heuristic and a Set Partitioning (SP) formulation to solve the Heterogeneous Fleet Vehicle Routing Problem. Salhi *et al.* (2013) dealed with the fleet size and mix vehicle routing problem with backhauls (FSMVRPB) based on the ILP formulation.

Current VRP optimization models are useful for a variety of practical applications. However, due to the complexity of the problem, the current solving quality and efficiency for the large-scale problem of MDHVRPTW are far from the practical requirements. While most solution methods have assumed a single depot and a homogeneous fleet, real-world problems usually include multiple terminals and a finite set of vehicles with non-uniform capacity. So, there are many problems need to make in-depth research such as how to seek feasible solutions, how to prevent falling into local optimum, and how to control the solution within the acceptable range. This paper presents an improved variable neighborhood search algorithm to solve MDHVRPTW; it integrates local search operator, optimization process and the simulated annealing algorithm into the VNS algorithm framework. Through the comparison with other algorithms, it shows the proposed algorithm gets the better solution.

3. Problem Descriptions

3.1. Second-order headings

The number of customers is denoted by n and the number of depots is denoted by m. Thus, the problem is defined on a complete graph $G=(V,E)$, where $V=\{v_1,\cdots v_n,v_{n+1},\cdots,v_{m+n}\}$ is the vertex set and $E=\{(v_i,v_j):v_i,v_j\in V,i\neq j\}$ is the arc set. The customer set $C=\{v_1,\cdots,v_n\}$ represents n customers, while vertices set $D=\{v_{n+1},\cdots,v_{n+m}\}$ corresponds to m depots. Each vertex $v_i\in V$ has several non-negative weights associated with it, namely, a demand q_i, a service time s_i, as well as an earliest e_i and latest l_i possible start time for the service, which define a time window [e_i, l_i]. For the depots these time windows correspond to the opening hours. Furthermore, the depot vertices v_{n+1} to v_{n+m} feature no demands and service times, i.e. $q_i=s_i=0$; $\forall i\in\{1,\cdots,m\}$. Associated to C_{ijk}^l is the transportation cost from customer i to customer j for k vehicles of l type. The delivery vehicles set $T=\{T_{11},\cdots,T_{lk},\cdots,T_{LK}\}$ corresponds to the set of k vehicles for l type. Each l type vehicle k has associated a non-negative capacity D_k^l and non-negative maximum route duration S_k^l. w_k^l is the capacity of k vehicles for l type. d_{ij} is the linear distance from customer i to customer j. R_m is the capacity of m depots. ρ_i is penalty cost for unit-time violations of the specified time window for customer node i, ρ_k is penalty cost for unit-time violations of the maximum working time for vehicle k, Δa_i represents ith-time window violation due to early service, Δb_i ith-time window violation due to late service, ΔB_k corresponds to working time violation for vehicle k . A_i is vehicle arrival time at node i .

A feasible solution to the MDHVRPTW problem must satisfy the following constraints:

(1) the distribution of vehicles over the depots is fixed a priori and given as the input data;

(2) each vehicle starts and ends at its home depot;

(3) each customer is served by one and only one vehicle;

(4) the total load and duration of vehicle k does not exceed D_k and T_k respectively;

(5) the service at each customer i begins within the associated time window $[e_i, l_i]$ and each vehicle route starts and ends within the time window of its depot;

(6) the goal is to minimize the total transportation cost of by all vehicles.

3.2. Problem mathematical formulation

(1) Objective function

The problem objective (1) aims to minimize the overall service expenses, including traveling distance and time costs, waiting and service time costs and penalty costs.

$$\min \sum_{i\in D\bigcup C}\sum_{j\in D\bigcup C}\sum_{k\in T} C_{ijk}^{l} X_{ijk}^{l} d_{ij} + \rho_k \Delta B_k + \sum_{i\in C}\rho_i(\Delta a_i + \Delta b_i) \tag{1}$$

(2) Problem constraints

- Assignment of nodes to vehicles

Eq. (2) states that every customer node must be serviced by a single vehicle.

$$\sum_{l,k\in T}\sum_{i\in D\bigcup C} X_{ijk}^{l} = 1, \qquad j \in C \tag{2}$$

- Capacity constraints

Constraint (3) states that the overall load to deliver to customer sites serviced by a used vehicle v should never exceed its cargo-capacity w_k^l. The distribution center (depot) has also the capacity constraint shown Eq.(4).

$$\sum_{i\in C}\sum_{j\in D\bigcup C} q_i X_{ijk}^{l} \le w_k^l, \qquad l,k \in T \tag{3}$$

$$\sum_{i\in C} q_i Z_{ij} - \sum_{m\in D} R_m \le 0 \tag{4}$$

- Assignment of vehicles to nodes

Constraint (5) ensures that the vehicle can only reach a customer node for one time;

$$\sum_{i\in D\bigcup C} X_{ijk}^{l} = Y_{ik}^{l}, \qquad j \in D\bigcup C, \qquad l,k \in T \tag{5}$$

Constraints (6) states that certain vehicles can only depart from a customer node for one time;

$$\sum_{j\in D\bigcup C} X_{ijk}^{l} = Y_{jk}^{l}, \quad i \in D\bigcup C, \quad l,k \in T \tag{6}$$

- Assignment of vehicles to depots

Eq. (7) ensures that each vehicle only belongs to a distribution center(depot);

$$\sum_{m\in D}\sum_{i\in C} X_{mik}^{l} \le 1, \quad l,k \in T \tag{7}$$

- Relationship between the routes and depots

Constraint (8) states any routes contain only a distribution center (depot);

$$S_{ik} - S_{jk} + nX_{ijk}^{l} \le n-1, \quad i,j \in C, \quad l,k \in T \tag{8}$$

- Overall traveling time for vehicle k

Constraint (9) states each route does not exceed the maximum mileage of the vehicle.

$$\sum_{l,k\in T}\sum_{i\in D\bigcup C} X_{ijk}^{l} d_{ij} \le D_{k}^{l}, \quad j \in C \tag{9}$$

- Time constraint violations due to early/late services at customer sites.

$$\Delta a_i \ge e_i - A_i, \quad \forall i \in C \tag{10}$$

$$\Delta b_i \ge A_i - l_i, \quad \forall i \in C \tag{11}$$

- Other constraints

$$X_{ijk}^{l} = 0,1 \quad i,j \in D\bigcup C, \quad l,k \in T \tag{12}$$

$$Y_{ik}^{l} = 0,1, \quad i \in C, \quad l,k \in T \tag{13}$$

$$Z_{ij} = 0,1, \quad i \in C, \quad j \in D \tag{14}$$

4. An Improved Variable Neighborhood Search Algorithm

VNS is a metaheuristic for solving combinatorial and global optimization problems proposed by Hansen and Mladenovic (1999, 2001). Starting from any initial solution, a so called shaking step is performed by randomly selecting a solution from the first neighborhood. This is followed by applying an iterative improvement algorithm. This procedure is repeated as long as a new incumbent solution is found. If not, one switches to the next neighborhood (which is typically larger) and performs a shaking step followed by the iterative improvement. If a new incumbent solution is found one starts with the first neighborhood; otherwise one proceeds with the next neighborhood, etc. The description consists of the building of an initial solution, the shaking phase, the local search method, and the acceptance decision. The flow of VNS is shown in Figure 2.

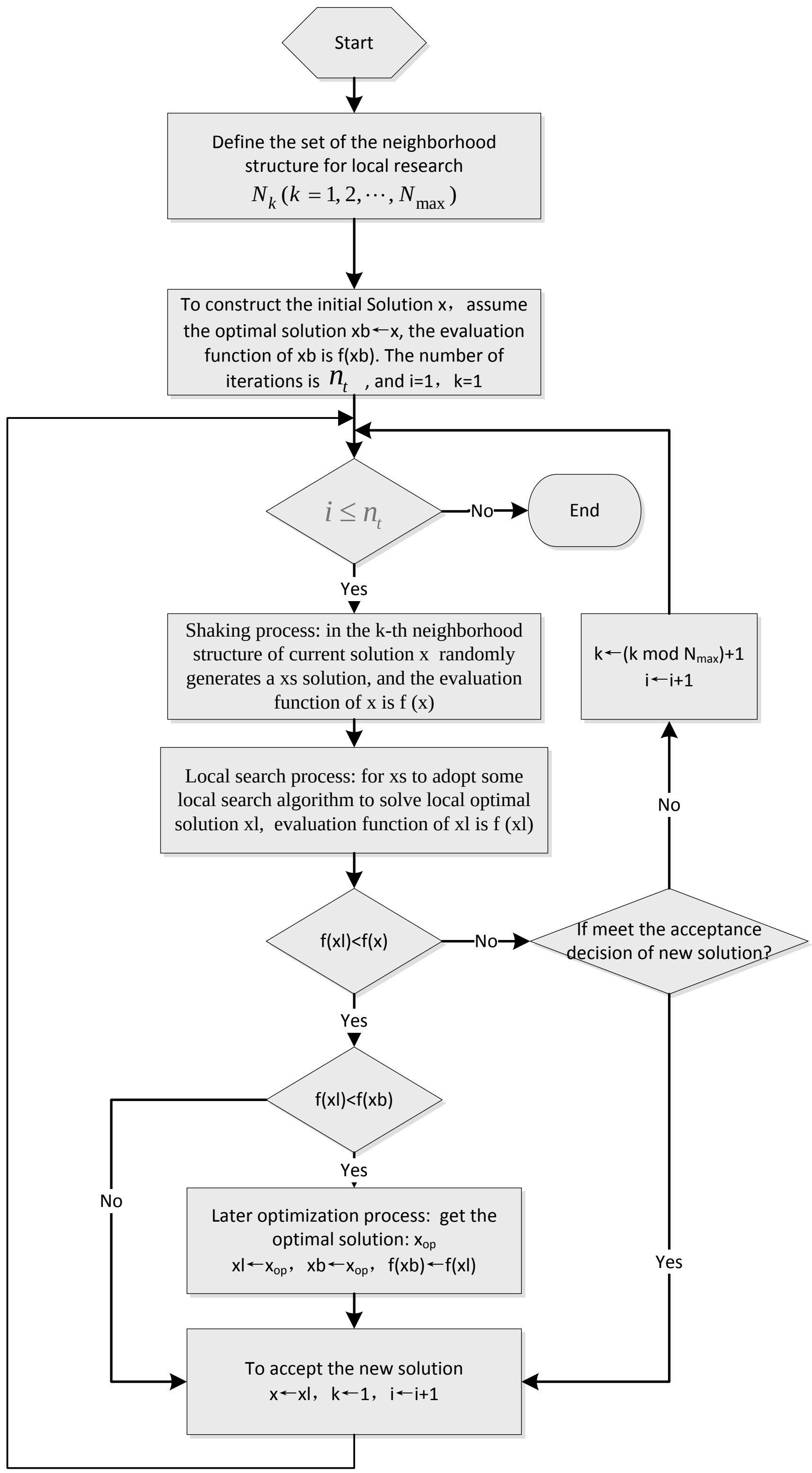

Figure 2. the flow of IVNS

4.1 Initial solution

Using variable neighborhood search algorithm, it first needs to build one or more initial feasible solution, an initial feasible solution mainly completes two tasks: customer allocation and path planning.

To assume h_{ij} represents the distance of from distribution center i to the customer j , the distance set is $H_{ij} = \{h_{ij} \mid i = 1, 2, \cdots, m; j = 1, 2, \cdots, n\}$. The minimum value is $\min H_{ij}$ in the distance set, the second minimum is $second \min H_{ij}$. Sum_k means the total number of customers that vehicles k delivers the goods, the set $U_k = \{y_{ik} \mid i = 0, 1, \cdots, Sum_k\}$ corresponds to the vehicle k to server the customers, y_{ik} presents the vehicle k as a transportation for customer i , y_{0k} means the distribution center(depot) is the initial point of the vehicle k . Steps of the initial feasible solution are as follows:

Algorithm 1 Steps of intital solution for the IVNS

Step 1, calculate $r_j = \min H_{ij} / second \min H_{ij}, 1 \le i \le m$, and select the appropriate $\varepsilon (0 < \varepsilon < 1)$;

Step 2, if $r_j < \varepsilon$, customer j will assign to the distribution center(depot) corresponding to $\min H_{ij}$

Step 3, repeat Step 1 and Step 2, traverses all customers point to form m customer groups;

Step 4, randomly take a customer group, the vehicle initial residual loading is $W_k^l = w_k^l, k = 0, Sum_k = 0, U_k = \Phi$;

Step 5, the demand of the customer i is q_i , and $k = 1$;

Step 6, if $w_k^l \le q_i$, $W_k^l = q_i - w_k^l$; else goto step 9;

Step 7, if $S_{i-1} + S_i \le S_k$, $U_k = U_k \bigcup \{i\}, Sum_k = Sum_k + 1$; otherwise goto Step 9;

Step 8, if $k > K$, $k = K$; else $k = k$;

Step 9, $k = k + 1$, goto step 6;

Step 10, $i = i + 1$, goto Step 5;

Step 11, repeat Step 5 to Step 10, K records the total of being used vehicle, U_k corresponds to the feasible route.

The initial solution obtained by the above method can basically meet the needs of the follow-up work, and built the foundation to get optimal feasible solution in following algorithm.

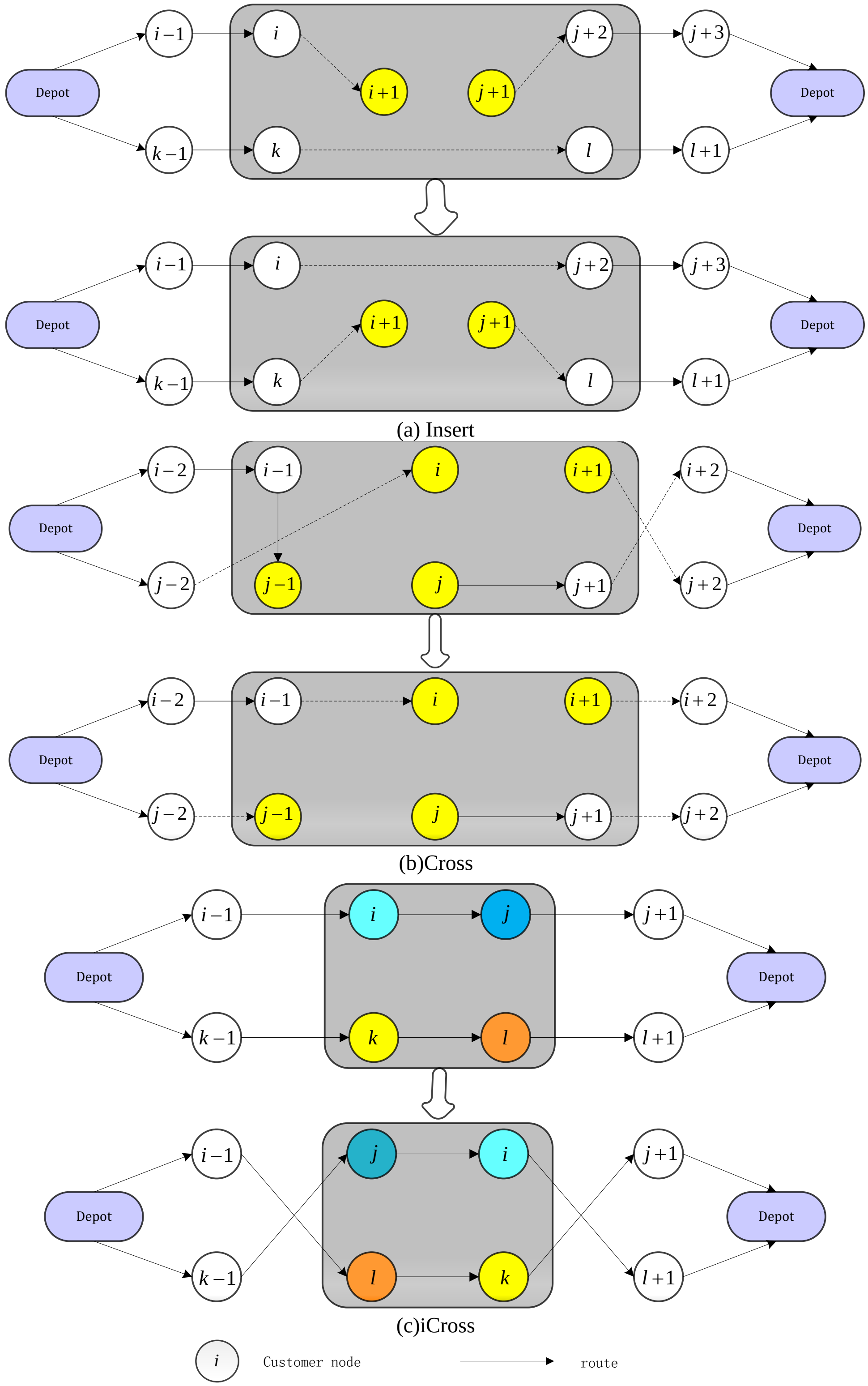

Figure 3. Insert and exchange operator

4.2 Shaking

Shaking is a key process in the variable neighborhood search algorithm design. The main purpose of the shaking process is to extend the current solution search space, to reduce the possibility the algorithm falls into the local optimal solution in the follow-solving process, and to get the better solution. The set of neighborhood structures used for shaking is the core of the VNS. The primary difficulty is to find a balance between effectiveness and the chance to get out of local optimal. In the shaking execution, it first selects a neighborhood structure N_k from the set of neighborhood structures of current solution x, then according to the definition of N_k, x corresponds to change and generate a new solution x^*.

There are two neighborhood structures to achieve the shaking: insert and exchange. Insert operator denotes a certain period of consecutive nodes move from the current path to another path; exchange operator refers to interchange the two-stage continuous nodes belonging to different paths. The insert and exchange operators are shown in Figure 3. The Cross exchange operator developed by Taillard, et al. (1997).The main idea of this exchange is to take two segments of different routes and exchange them. Compared with the VNS by Polacek, et al. (2004) the selection criterion is slightly changed. Now it is possible to select the same route twice. This allows exploring more customer visit combinations within one route. An extension to the CROSS exchange operator is introduced by Braysy (2003); this operator is called improved Cross exchange - iCross exchange for short. Both operators are used to define a set of neighborhood structures for the improved VNS.

In each neighborhood the insert operator is applied with a probability p_{insert} to both routes to further increase the extent of the perturbation, then the probability of the exchange operator is $1-p_{insert}$. IVNS selects randomly an exchange operator to change path for each shaking execution. The shaking process is somewhat similar to the crossover operation of the genetic algorithm. When the process is finished, the only two paths have the exchange of information; most of the features of the current solution will be preserved, to speed the convergence of the algorithm.

4.3 Local search

In a VNS algorithm, local search procedures will search the neighborhood of a new solution space obtained through shaking in order to achieve a locally optimal solution. Local search is the most time-consuming part in the entire VNS algorithm framework, and decides the final solution quality so computational efficiency must be considered in the design process of local search algorithm. Two main aspects are considered in the design of local search algorithms: local search operator and the search strategy. Based on the previous studies, this paper selects $2-opt$ and $3-opt$ as a local search operator in order to obtain the good quality local optimal solution in a short period. They can be called hybird operators. According to the probability, one of the two operators is selected in each local search process. The parameter p_{2-opt} represents the probability of selection for $2-opt$, similarly, the probability of selection for $3-opt$ can be expressed as $1-p_{2-opt}$. This mixed operator can develop optimization ability for $2-opt$ and $3-opt$, and expand the solution space of the algorithm.

There are mainly two search strategies: first-improvement and best-improvement in local search algorithm. The former refers to access the neighborhood solution of the current x solution successively in the solution process, if the current access neighborhood solution x_n is better than x, to make $x = x_n$ and update neighborhood solution. To repeat these steps until all the neighborhood solutions of x are accessed. Finally, x will be obtained as a local optimal solution. The latter refers to traverse all of the neighborhood solution of current x solution in the solution process, to select the optimum neighborhood solution x_n as a local optimal solution. In this paper, we adopt the best-improvement strategy, it enables the algorithm to achieve a better balance in the solution quality and run time.

4.4 Later optimization process

In order to accelerate the convergence speed and improve the solution quality, the later optimization process is proposed in the IVNS algorithm. After the local search is completed, if the local optimal solution xl is better than the global optimal solution xb, that is f (xl)<f (xb), the later optimization process will be continue to be implemented in order to seek a better global optimal solution(Zheng Wang, 2011). The algorithm of later optimization process which was proposed by Gendreau et al. is suitable for solving the traveling salesman problem and the vehicle routing problem with time windows.

4.5 Acceptance decision

To avoid that the VNS becomes too easily trapped in local optima, due to the cost function guiding towards feasible solutions and most likely complicating the escape of basins surrounded by infeasible solutions, we also allow to accept worse solutions under certain conditions. This is accomplished by utilizing a Metropolis criterion like in simulated annealing (Kirkpatrick, 1983) for inferior solutions x^* and accepts them with a probability of Eq. (15), depending on the cost difference to the actual solution x of the VNS process and the temperature T. We update T every n_T iterations by an amount of $T_{n+1} = \delta T_n$, where q_0 a random number on the interval [0,1], Where δ is settable cooling coefficient, and an initial temperature value is $T_0 = 10$.

$$SA(x^*, x) = \begin{cases} x^* & if \quad q_0 > \exp(\frac{f(x^*) - f(x)}{T}) \\ x & if \quad q_0 \le \exp(\frac{f(x^*) - f(x)}{T}) \end{cases} \tag{15}$$

5. Numerical Experiments

5.1 Problem data and experimental setting

In order to assess the performance of the improved variable neighborhood search algorithm to solve MHFVRPTW, the experiments analyze and compare with other existing algorithms. IVNS algorithm is implemented by the C # language, and the main configuration of the computer is Intel Core i3 1.8GHz, 2 GB RAM and Window XP. In this experiment, the data sets from the literature (Golden et al., 1984; Taillard, 1999; Choi and Tcha, 2007) are used. The benchmark problem is used to test the performance of the algorithm, and it is 12 of 20 issues proposed by Golden *et al.* Here, the largest instance has 100 customers. For each instance, we define the relative percentage deviation (RPD), and its computation equation is as follows:

$$RPD = \frac{\cos t_i - best_i}{best_i} \times 100 \qquad (16)$$

where $\cos t_i$ and $best_i$ denote, for the i^{th} instance, the cost found by our heuristic and the best known solution respectively. The average deviation is then computed over all instances in the data set.

The initial values of the various parameters for IVNS algorithm are set as follows:

(1) the parameter settings for simulated annealing accepted criteria are initial temperature $T_0 = 10$, every $n_T = n/10$ generation to update temperature $T_{n+1} = 0.9 \times T_n$, $n_t = 1000$ to end the algorithm.

(2) the parameters value of the Shaking operation are as follows: $p_{insert} = 0.2$, $p_{cross} = 0.15$, $p_{icross} = 0.1$.

(3) the p_{2-opt} value is 0.5 in local search.

5.2 Numerical results

Table 1. Comparison of solution quality from the different methods

No	n	Best	Gendreau (1999)		Choi & Tcha (2007)		Lee (2008)		Imran, *et al.* (2009)		IVNS	
			Cost	CPU	Cost	CPU	Cost	CPU	Cost	CPU	Cost	CPU
G3	20	**961.03**	**961.03**	164	**961.03**	0	**961.03**	59	**961.03**	21	**961.03**	0
G4	20	**6437.33**	6445.10	253	**6437.33**	1	**6437.33**	79	**6437.33**	18	**6437.33**	1
G5	20	**1007.05**	**1007.05**	164	**1007.05**	1	**1007.05**	41	**1007.05**	13	**1007.05**	2
G6	20	**6516.47**	**6516.47**	309	**6516.47**	0	**6516.47**	89	**6516.47**	22	**6516.47**	2
G13	50	**2406.36**	**2406.36**	724	**2406.36**	10	2408.41	258	**2406.36**	252	**2406.36**	21
G14	50	**9119.03**	9125.65	1033	**9119.03**	51	9160.42	544	**9119.03**	274	**9119.03**	19
G15	50	**2586.37**	2606.72	901	**2586.37**	10	**2586.37**	908	**2586.37**	303	2586.72	23
G16	50	**2720.43**	**2720.43**	815	2728.14	11	2724.33	859	2741.50	253	**2720.43**	20
G17	75	**1734.53**	**1734.53**	1022	**1734.53**	207	1745.45	1488	1745.33	745	1743.76	52
G18	75	**2369.65**	2412.56	691	**2369.65**	70	2373.63	2058	**2369.65**	897	**2369.65**	50
G19	100	**8659.74**	8685.71	1687	8661.81	1179	8699.98	2503	8665.12	1613	8664.81	175
G20	100	**4039.49**	4166.73	1421	4042.59	264	4043.47	2261	4066.94	1595	**4039.49**	108
Average		4046.46	4065.70	765.33	4047.53	150.33	4055.33	928.92	4051.85	500.50	4047.68	39.42
BestNum*			6		9		5		8		9	
ARPD**			0.03		0.03		0.22		0.13		0.12	

* ARPD-average relative percentage deviation
** BestNum-the number of best solution

The remaining 12 questions proposed by Golden are considered to be as the benchmark problem. The experiment results are shown in Table 1, where Best denotes best solution, row Average is the average solution of all the problems, row ARPD represents average RPD, the last row BestNum given algorithm to obtain the number of the best solution. The results in Table 5 show that we proposed algorithms can get nine best solutions, ARPD is 0.12. The algorithm of Taillard and Lee obtains five best solutions, ARPD are 0.14 and 0.22 respectively, and they are similar to our algorithm. Grendreau, Wassan and Osman with their algorithms have found six known best solution of the problem. The ARPD of Brandao, Wassan and Osman respectively are 0.03 and 0.51. Choi and Tcha, Brandao, Imran find the nine best solutions of the questions, ARPD are 0.05, 0.72, and 0.13, and one is slightly better than ours; two others are somewhat higher than our algorithm. However, on the issue of problem G20, the difference between the results of the three algorithms for Taillard, Imran and IVNS are small, ARPD 0.14, 0.13 and 0.12, respectively.

Due to the operating environment of each algorithm, and therefore can not be directly compared to the CPU time of the algorithm. According to Linpack benchmark of Donbarra, we can get a roughly relative speed for different microcomputer through Mflops (Million Floationg Point / Second) standard, then roughly compare the algorithm time. Finally, it is clear that our heuristic methods require a reasonable amount of CPU time.

6. Applications

There is a large water project in China, it divides 10 periods, needs 64.5 months to complete. The requirement of the filling materials for the whole project is $3426.86 \times 10^4 m^3$, where rockfill materials is $2766.86 \times 10^4 m^3$, filter material is $191.49 \times 10^4 m^3$, and core-wall materials is $468.51 \times 10^4 m^3$. Large water project is a material flow equilibrium including excavation sites, filling sites, transferring yard, excavation waste dump sites, material yard, the distribution center, and equipment parking etc. Figure 4 shows the construction plan, includes two depots, sixteen sites with different demand. How to minimize the transportation cost and reduce the project period is very important to consider for project administrative department.

In order to assess whether the planning cost is optimal or not, we have some assumption to calculate the optimal cost with our proposed algorithm. For simplicity, as an example, the optimization process is described in fourth stage. There are two distribution centers (depots) and 16 customer points, two distribution center coordinates are (9,10) and (35,27.5) respectively(units: km). The coordinates of each customer points and distribution volume are as shown in Table 2. There are two kind of vehicles with different load capacity, for A and B, their transport capacities are 70t and 100t respectively. The goal is to arrange the delivery vehicles reasonably, to minimize the total distribution costs.

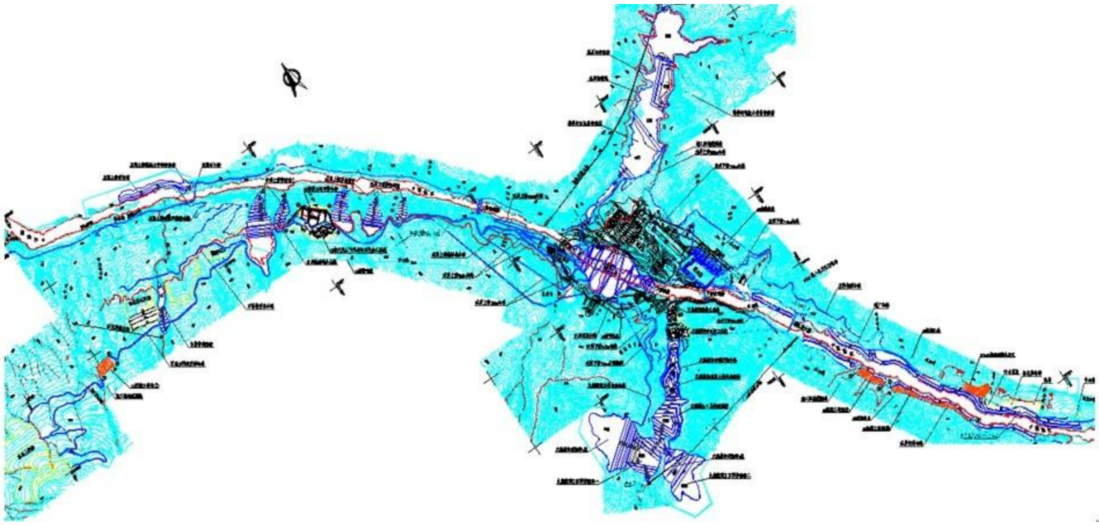

Figure 4. The construction plan of X project

Table 2. The position of customer points and demand

No	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
X	5	25	27.5	30	15	35	10	15	37.5	27.5	32.5	30	25	22.5	42.5	22.5
Y	22.5	35	25	20	32.5	12.5	30	22.5	22.5	35	38.5	17.5	20	14	17.5	30
Demand	20	10	10	30	30	10	10	10	20	10	20	20	20	30	10	30

Using the proposed algorithm on the same machine with the same parameters, to randomly solve the problem for 10 times, and the results are shown in Table 3.

Table 3. The result of MHVRPTW with IVNS

No	Total Mileage /km	Vehicle Type	
		A	B
1	181.69	2	3
2	190.51	3	2
3	174.12	1	3
4	195.34	4	1
5	194.25	3	2
6	174.51	1	3
7	190.01	2	2
8	181.79	2	3
9	174.26	1	2
10	182.09	2	3
Average	183.86	4.5	
Standard deviation	7.19	0.43	

As can be seen from Table 3, it gets the higher quality of the solution for 10 times, the average of total mileage and vehicle respectively is183.86km and 4.5. The calculation results of the algorithm is fairly stable, the total mileage of the worst solution is only 11.87% more than the best solution. On the computational efficiency, three times reach the best solution; three times reach the second best solution. Optimal total mileage is 173.4km, shown in Figure 5. A specific solution is shown in Table 4. Table 5 shows the comparison results for the planning and optimization. Through analyzing the 10 stages comparison results, the results are improved in some different degree, in five stages, the reduction ratio is 7.35%, the mileage is reduced from 252.9km to 234.3km. Overall, the total mileage is reduced from 2963.6km to 2860.2km, the average reduction ratio is 3.49%. So, the proposed algorithm can reduce the mileage and save cost for the large water project, the results show our model is effective and feasible.

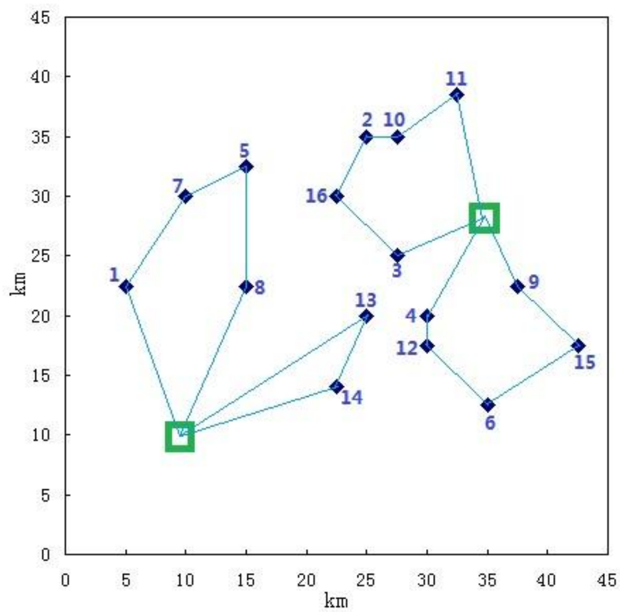

Figure 5. the optimal route with IVNS

Table 4. The optimal solution of delivery for MHVRPTW

Depot	Route	Delivery/t	Vehicle Type	Mileage/km
(35,27.5)	0-11-10-2-16-3-0	100-80-70-60-30-20	B	38.73
(35,27.5)	0-4-12-6-15-9-0	100-70-50-40-30-10	B	53.785
(9,10)	0-1-7-5-8-0	70-50-40-10-0	A	51.1
(9,10)	0-13-14-0	70-50-20	A	29.91
The number of vehicle		4		
The total mileage /km		173.4		

Table 5. The comparison of planning and optimization (Unit: km)

Stage1	1	2	3	4	5	6	7	8	9	10	Total
Planning	143.1	187.5	182.8	212.4	252.9	342.2	368.3	385.7	402.6	486.1	2963.6
Improved	132.7	176.2	173.4	210.3	234.3	323.1	362.8	379.3	398.5	463.6	2860.2
reduction %	3.07	6.03	5.14	0.99	7.35	5.58	1.49	1.66	1.02	4.63	3.49

7. Conclusions

An attempt has been made here to solve the Multi Depot Heterogeneous Vehicle Routing Problem with Time Windows (MDHVRPTW) via an Improved Variable Neighborhood Search (IVNS) metaheuristic. In the algorithm, a clustering algorithm is utilized to allocate customers in the initial solution construction phase, a hybrid operator of insert and exchange are used to achieve the shaking process, the Best-improvement strategy is adopted, and it can make the algorithm to achieve a better balance in the solution quality and running time. Computational experience with the benchmark test instances confirms that our approach utperforms all the existing algorithms both in terms of the quality of solutions generated and the solution time. Finally, this study shows that our method can be applied successfully in the large water project problems.

In summary, the proposed algorithm in this paper has good global searching ability, faster convergence speed, and at the same time it can overcome premature convergence and get the higher solving quality. Through the application of IVNS in the large water project in China, it shows that the proposed algorithm enables enterprises to shorten delivery mileage, save distribution vehicle, and reduce construction costs and enhance economic efficiency in some way. Future research lines include the development of a mathematical framework for further improving the solution provided by the hybrid approach and the extension of the strategy to more difficult problems such as the Multi Depot Heterogeneous Vehicle Routing Problem with the pick-up and delivery.

Acknowledgments

Funding for this research is supported by the national Natural Science Foundation of China (Grant Nos.71301152, 71271013 and 71301011), National Social Science Foundation of China (Grant No. 11AZD096), the National Key Technology R&D Program of the Ministry of Science and Technology (Grant No. 2013BAK04B02, 2013BAK04B04), Quality Inspection Project (Grant No. 2014424309), and China Postdoctoral Science Foundation (Grants No. 2013T60091, 2012M520008).

References

[1] A. Subramanian, P. H. Vaz Penna, E. Uchoa and L. S. Ochi, "A Hybrid Algorithm for the Heterogenous Fleet Vehicle Routing Problem", European Journal of Operational Research, vol. 221, no. 2, **(2012)**, pp. 285-295.

[2] A. Bettinelli, A. Ceselli and G. Righini, "A branch-cut-price algorithm for the multi-depot heterogeneous vehicle routing problem with time windows", Transportation Research Part C: Emerging Technologies, vol. 19, no. 5, **(2011)**, pp. 723-740.

[3] A. Imran, S. Salhi and N. A. Wassan, "A Variable Neighborhood-Based Heuristic for the Heterogeneous Fleet Vehicle Routing Problem", European Journal of Operational Research, vol. 197, no. 2, **(2009)**, pp. 509-518.

[4] C. Avanthay, A. Hertz and N. Zufferey, "A variable neighborhood search for graph coloring", European Journal of Operational Research, vol. 151, no. 2, **(2003)**, pp. 379-388.

[5] V. Bouffard and J. A. Ferland, "Improving simulated annealing with variable neighborhood search to solve the resource-constrained scheduling problem", Journal of Scheduling, vol. 10, no. 6, **(2007)**, pp. 375-386.

[6] J. Brandao, "A Deterministic Tabu Search Algorithm for the Fleet Size and Mix Vehicle Routing Problem", European Journal of Operational Research, vol. 195, no. 3, **(2009)**, pp. 716-728.

[7] O. Braysy, "A reactive variable neighborhood search for the vehicle routing problem with time windows", Informs Journal on Computing, vol. 15, no. 4, **(2003)**, pp. 347-368.

[8] E. Choi and D. W. Tcha, "A Column Generation Approach to the Heterogeneous Fleet Vehicle Routing Problem", European Journal of Operational Research, vol. 34, **(2007)**, pp. 2080-2095.

[9] R. Dondo and J. Cerda, “A Cluster-Based Optimization Approach for Multi-Depot Heterogeneous Fleet Vehicle Routing Problem with Time Windows”, European Journal of Operational Research, vol. 176, **(2007)**, pp. 1478-1507.
[10] R. Dondo and J. Cerda, “A hybrid local improvement algorithm for large-scale multi depot vehicle routing problems with time windows”, Computers and Chemical Engineering, vol. 33, no. 2, **(2009)**, pp. 513-530.
[11] K. Fleszar, I. H. Osman and K. S. Hindi, “A variable neighborhood search algorithm for the open vehicle routing problem”, European Journal of Operational Research, vol. 195, no. 3, **(2009)**, pp. 803-809.
[12] J. Gao, L. Sun and M. Gen, “A hybrid genetic and variable neighborhood descent algorithm for flexible job shop scheduling problems”, Computer of Operational Research, vol. 35, no. 9, **(2008)**, pp. 2892-2907.
[13] A. Goel and V. Gruhn, “A general vehicle routing problem”, European Journal of Operational Research, vol. 191, no. 3, **(2008)**, pp. 650-660.
[14] P. Hansen, J. Brimberg, *et al.*, "Primal-dual variable neighborhood search for the simple plant-location problem", Informs Journal on Computing, vol. 19, no. 4, **(2007)**, pp. 552-564.
[15] P. Hansen, N. Mladenovic and D. Perez, “Variable neighborhood decomposition search”, Journal of Heuristics, vol. 7, no. 4, **(2001)**, pp. 335-350.
[16] P. Hansen and N. Mladenovic, “Variable neighborhood search for the p-median”, Location Science, vol. 5, **(1997)**, pp. 207-226.
[17] V. C. Hemmelmayr, K. F. Doerner and R. F. Hartl, “A variable neighborhood search heuristic for periodic routing problems”, European Journal of Operational Research, vol. 195, no. 3, **(2009)**, pp. 791-802.
[18] J. Kytojoki, T. Nuortio, O. Braysy, *et al.*, "An efficient variable neighborhood search heuristic for very large scale vehicle routing problems", Computer and Operational Research, vol. 34, no. 9, **(2007)**, pp. 2743-2757.
[19] Y. H. Lee, J. I. Kim, K. H. Kang and K. H. Kim, “A Heuristic for Vehicle Fleet Mix Problem Using Tabu Search and Set Partitioning”, Journal of the Operational Research Society, vol. 59, **(2008)**, pp. 833-841.
[20] C. J. Liao and C. C. Cheng, “A variable neighborhood search for minimizing single machine weighted earliness and tardiness with common due date”, Computers and Industrial Engineering, vol. 52, no. 4, **(2007)**, pp. 404-413.
[21] M. A. Adibia, M. Zandiehb and M. Amiri, “Multi-objective scheduling of dynamic job shop using variable neighborhood search”, Expert Systems with Applications, vol. 37, no. 1, **(2010)**, pp. 282-287.
[22] M. Gendreau, F. Guertin, *et al.*, "Neighborhood search heuristics for a dynamic vehicle dispatching problem with pick-ups and deliveries", Transportation Research Part C, vol. 14, **(2006)**, pp. 157–174.
[23] A. Ostertag, K. F. Doerner, *et al.*, "A variable neighborhood search integrated in the popmusic framework for solving large scale vehicle routing problems", Proceeding of the 5th International Workshop on Hybrid Metaheuristic, Malaga, Spain, **(2008)**, pp. 29-42,
[24] M. Polacek, R. F. Hartl, *et al.*, "A variable neighborhood search for the multi depot vehicle routing problem with time windows", Journal of Heuristics, vol. 10, no. 6, **(2004)**, pp. 613-627.
[25] C. C. Ribeiro and M. C. Souza, “Variable neighborhood search for the degree-constrained minimum spanning tree problem”, Discrete Applied Mathematics, vol. 118, no. 2, **(2002)**, pp. 43-54.
[26] S. Kirkpatrick, C. D. Gelatt Jr. and M. P. Vecchi, "Optimization by simulated annealing", Science, vol. 220, no. 4598, **(1983)**, pp. 671-680.
[27] S. Pirkwieser and G. R. Raidl, "A Variable Neighborhood Search for the Periodic Vehicle Routing Problem with Time Windows", EU/Meeting, **(2008)**, Troyes, France.
[28] S. Salhi, N. Wassan and M. Hajarat, “The Fleet Size and Mix Vehicle Routing Problem with Backhauls: Formulation and Set Partitioning-based Heuristics”, Transportation Research Part E: Logistics and Transportation Review, vol. 56, **(2013)**, pp. 22-35.
[29] S. Sutapa and M. K. Jha, "Multi Depot Probabilistic Vehicle Routing Problems with a Time Window: Theory, Solution and Application", International Journal of Operations Research and Information Systems, vol. 2, no. 2, **(2011)**, pp. 17-41.
[30] C. J. Ting and C. H. Chen, “Combination of multiple ant colony system and simulated annealing for the multi depot vehicle routing problem with time windows”, Transportation Research Record: Journal of the Transportation Research Board, vol. 9, no. 1, **(2008)**, pp. 85-92.
[31] L. Wen and F. H. Meng, "An improved PSO for the multi depot vehicle routing problem with time windows", Proceeding of the 2008 IEEE Pacific-Asia Workshop on Computational Intelligent and Industrial Application, IEEE Computer Society, **(2008)**, pp. 852-856.
[32] X. Wang and Y. Li, “Study on multi-depot and multi-type vehicles vehicle routing problem with backhauls”, Control and decision, vol. 24, no. 12, **(2009)**, pp. 1769-1705.

Authors

Yingcheng Xu received the Ph.D. degree from Beijing University of Aeronautics and Astronautics, Beijing, China.

He is currently a Post-Doctoral Fellow with China National Institute of Standardization, Beijing, China. His research interests include information systems, system simulation and modeling, supply chain management, and logistics.

Wei Jiang is currently an Associate Professor with School of Computer and Information, Anqing Normal University, China. His research interests include heuristic algorithms, management information system, e-business, project management, and logistics and supply chain management.

