Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2016, Article ID 9071394, 10 pages
http://dx.doi.org/10.1155/2016/9071394

Research Article

Vehicle Coordinated Strategy for Vehicle Routing Problem with Fuzzy Demands

Chang-shi Liu,[1,2,3,4] Gang Kou,[5] and Fu-hua Huang[1]

[1] *School of Management, Hunan University of Commerce, Changsha 410205, China*
[2] *Mobile E-Business Collaborative Innovation Center of Hunan Province, Hunan University of Commerce, Changsha 410205, China*
[3] *Key Laboratory of Hunan Province for Mobile Business Intelligence, Hunan University of Commerce, Changsha 410205, China*
[4] *Institute of Big Data and Internet Innovation, Hunan University of Commerce, Changsha 410205, China*
[5] *School of Business Administration, Southwestern University of Finance and Economics, Chengdu 611130, China*

Correspondence should be addressed to Gang Kou; kougang@swufe.edu.cn

Received 24 December 2015; Accepted 8 August 2016

Academic Editor: Thomas Hanne

The vehicle routing problem with fuzzy demands (VRPFD) is considered. A fuzzy reasoning constrained program model is formulated for VRPFD, and a hybrid ant colony algorithm is proposed to minimize total travel distance. Specifically, the two-vehicle-paired loop coordinated strategy is presented to reduce the additional distance, unloading times, and waste capacity caused by the service failure due to the uncertain demands. Finally, numerical examples are presented to demonstrate the effectiveness of the proposed approaches.

1. Introduction

The vehicle routing problem (VRP) was first proposed by Dantzig and Ramser [1]. The VRP has since been the topic of many operational studies. VRP consists of designing efficient routes to serve a number of nodes with a fleet of vehicles. Each node is visited exactly once by one vehicle. The activity of the vehicle is bounded by certain constraints. Each vehicle starts at the depot and returns to the same depot after completing its task. Most VRP studies employ the vehicle uncoordinated strategy; that is, there is no coordination between the vehicles, and each vehicle completes only its own task. There are many significant VRP results based on this case, including those of Clarke and Wright [2], Solomon [3], Laporte [4], Figliozzi [5], Sprenger and Mönch [6], Pillac et al. [7], Kou et al. [8], and Kou et al. [9].

The vehicle routing problem with fuzzy demands (VRPFD) is an extension of the VRP; that is, the demand of each node is uncertain, subjective, ambiguous, and/or vague [10]. The VRPFD is widely employed for many real applications due to their numerous uncertainties, including garbage collection systems, product recall systems, and raw milk collection systems (collecting raw milk from dairy farmers for milk powder production enterprises). There are also several classical studies that refer to the VRPFD, such as Bertsimas [11], Cao and Lai [12], Kuo et al. [13], Kou and Lin [14], Kou et al. [15], Allahviranloo et al. [16], and Hu et al. [17]. The VRPFD typically assumes that the real value of a node's demand is known when the vehicle reaches the node, whereas the vehicle's route is planned in advance. After serving v nodes, the vehicle might not be able to service the $v+1$ node once it arrives due to insufficient capacity. In such situations, if the vehicle uncoordinated strategy is employed, the vehicle must return to the depot and unload what it has picked up thus far and then return to the node where it had a "service failure" and continue to serve the remaining nodes. Thus, "additional distance" and "additional unloading times" are introduced due to the "service failure." However, there are also vehicles with surplus capacity after completing their own tasks, introducing "waste capacity." All of these cases result in increasing logistics cost. To the authors' knowledge, few researchers have addressed the problem of minimizing the "additional distance" and "waste capacity," let alone "additional unloading times," in the VRPFD.

In this paper, vehicle coordinated strategy (VCS) is defined such that each vehicle finishes its own assigned task first; then, if there is a vehicle with surplus capacity, the vehicle must help any vehicle that has not completed its own task according to the specified vehicle coordination rules [18, 19]. Only a few VRP studies have considered VCS. Shang and Cuff [20] considered a multiobjective vehicle routing heuristic for a pickup and delivery problem. They assumed that the fleet size is not predetermined and that customers are allowed to transfer between vehicles. These transfers can occur at any location and between any two vehicles. Yang et al. [21] proposed a mixed-integer programming formulation for the offline version of the real-time VRP and compared five rolling horizon strategies for the real-time version. To some extent their work is relevant to vehicle coordination. Liu et al. [18] proposed a simple general VCS for the VRP with deterministic demands. Lin [19] designed a VCS with single or multiple vehicle uses. The VCS is defined as allowing vehicles to travel to transfer items to another vehicle returning to the depot, provided that no time window constraints are violated. Sprenger and Mönch [6] studied a methodology to solve a cooperative transportation planning problem motivated by a real-world scenario found in the German food industry. Several manufacturers with the same customers but complementary food products share their vehicle fleets to deliver to their customers. They designed a heuristic to solve the problem. The results of extensive simulation experiments demonstrated that the cooperative setting outperforms the noncooperative one. Hu et al. [22] presented a feasible routing solution to accommodate the changes (such as customer's demand changes, delivery time window changes, disabled roads induced by traffic accidents or traffic jams, and vehicle breakdowns) and to minimize the negative impacts on the existing distribution process in real-time VRP. They handle these disruptions by readjusting vehicle routes in real time to improve vehicles' efficiency and enhance service quality. To a certain extent their work is relevant to vehicle coordination. However, all investigations assumed that the customers were uniformly distributed in certain regions and that the demands were deterministic. To the authors' knowledge, few studies have employed the VCS in the VRPFD.

Thus, in this paper, the fuzzy reasoning constrained program model for VRPFD is formulated, and the hybrid ant colony algorithm is designed to minimize total travel distance. In particular, the two-vehicle-paired loop coordinated strategy (TVPLCS) is presented to reduce the "additional distance," "additional unloading times," and "waste capacity" caused by the service failure due to the uncertain demands. Finally, numerical examples are presented to demonstrate the effectiveness of the proposed approaches.

The remainder of this paper is organized as follows. In Section 2, the fuzzy reasoning constrained program model for VRPFD is formulated. In Section 3, we design a hybrid ant colony algorithm for VRPFD. In Section 4, in particular, we present the TVPLCS to minimize the "additional distance," "additional unloading times," and "waste capacity". In Section 5, we give numerical examples to demonstrate the effectiveness of the proposed approaches. Finally, we summarize the contributions of this paper.

2. VRPFD Assumptions and Model

2.1. VRPFD Assumptions. In this paper, the VRPFD assumes that there is only one depot denoted by 0, and there are n nodes with fuzzy demands served by m vehicles. The locations of the depot and nodes are known. The fuzzy demand of each node i is uncertain and only characterized by a triangular fuzzy number D_i, $D_i = (d_{i1}, d_{i2}, d_{i3})$. d_{i1} is the minimum of the demand of node i, d_{i3} is the maximum of the demand of node i, and d_{i2} is the most likely value. The distance c_{ij} between nodes i and node j is known. Each node is served exactly once by one vehicle. For simplicity, the capacity Q of each vehicle is the same, and the activity of the vehicle is only bounded by capacity constraints. Each vehicle starts at the depot and returns to the same depot after completing its task. The objective is to design a set of vehicle routes that minimizes the total logistics costs.

2.2. Deciding whether the Vehicle Serves the Next Node or Returns to the Depot. When the demand of each node is deterministic, it is easy for us to decide whether the vehicle is able to serve the next node after serving v nodes. However, while the demand at each node is uncertain and only characterized by a triangular fuzzy number $D_i = (d_{i1}, d_{i2}, d_{i3})$, it is difficult for us to decide whether the vehicle should serve the next node $v + 1$ or return to the depot. We only know that the greater the vehicle's remaining capacity and the lesser the demand at the next node, the greater the vehicle's "chances" of being able to serve the next node. In this paper, we solve this problem by triangular fuzzy number theory proposed by Liu [23], which described as follows.

The membership function of triangular fuzzy number $D = (d_1, d_2, d_3)$ is defined as

$$\mu_D(x) = \begin{cases} 0, & x \le d_1 \text{ or } x \ge d_3 \\ \dfrac{(x - d_1)}{(d_2 - d_1)}, & d_1 \le x < d_2 \\ \dfrac{(d_3 - x)}{(d_3 - d_2)}, & d_2 \le x < d_3. \end{cases} \tag{1}$$

Let $\text{pos}\{\varepsilon\}$ be the occurrence possibility of event ε. For triangular fuzzy number $A = (a_1, a_2, a_3)$ and $B = \{b_1, b_2, b_3\}$, $\text{pos}\{A \le B\}$ is defined as

$$\begin{aligned} \text{pos}\{A \le B\} &= \sup\{\min\{\mu_A(x), \mu_B(y)\} \mid x \le y\} \\ &= \begin{cases} 1, & a_2 \le b_2 \\ \dfrac{(b_3 - a_1)}{(b_3 - b_2) + (a_2 - a_1)}, & a_2 > b_2,\ a_1 < b_3 \\ 0, & a_1 \ge b_3. \end{cases} \end{aligned} \tag{2}$$

Now, we can deduce that the occupied capacity Q_{vo} of the vehicle which had served v nodes is

$$Q_{vo} = \sum_{i=1}^{v} D_i = \left(\sum_{i=1}^{v} d_{i1}, \sum_{i=1}^{v} d_{i2}, \sum_{i=1}^{v} d_{i3} \right). \tag{3}$$

Also, the capacity Q of each vehicle can be presented as a triangular fuzzy number $Q = (Q, Q, Q)$. So, the remaining capacity Q_{vr} of the vehicle is

$$\begin{aligned} Q_{vr} = Q - Q_{vo} &= \left(Q - \sum_{i=1}^{v} d_{i3}, Q - \sum_{i=1}^{v} d_{i2}, Q - \sum_{i=1}^{v} d_{i1}\right) \\ &= \left(Q_{vr1,} Q_{vr2}, Q_{vr3}\right). \end{aligned} \tag{4}$$

Thus, the possibility $\operatorname{pos}(D_{v+1} \le Q_{vr})$, which means that the demand of node $v + 1$ is less than the remaining capacity, is

$$\operatorname{pos}\left(D_{v+1} \le Q_{vr}\right) = \begin{cases} 1, & d_{v+1,2} \le Q_{vr2} \\ \dfrac{\left(Q_{vr3} - d_{v+1,1}\right)}{\left(Q_{vr3} - Q_{vr2}\right) + \left(d_{v+1,2} - d_{v+1,1}\right)}, & d_{v+1,2} > Q_{vr2},\ d_{v+1,1} < Q_{vr3} \\ 0, & d_{v+1,1} \ge Q_{vr3}. \end{cases} \tag{5}$$

Let $P^* \in [0, 1]$ be the decision maker's preference. A large value of P^* indicates that the decision maker is risk averse, and the decision maker aims to ensure service. In this case, $P^* \ge 0.6$. The possibility of "service success" is relatively high. In contrast, a small value of P^* means that the decision maker has an insatiable appetite for risk and tries to serve more nodes with each vehicle. In that case, $P^* \le 0.5$. The possibility of "service success" is relatively low.

Now, after serving v nodes, we can make a decision whether the vehicle should serve the next node $v+1$ or return to the depot 0. The decision was made as follows.

If $\operatorname{pos}(D_{v+1} \le Q_{vr}) \ge P^*$, the vehicle should serve the next node $v + 1$; else, the vehicle should return to the depot 0.

2.3. VRPFD Model.

The notations used in the formulation of the VRPFD are described as follows.

i: node index ($i = 0$ stands for the depot).

n: number of nodes.

N: set of nodes.

S: nonempty proper subset of the set N.

D_i: fuzzy demand of each node i, $D_i = (d_{i1}, d_{i2}, d_{i3})$.

c_{ij}: distance between node i and node j.

k: vehicle index.

m: number of vehicles.

K: set of vehicles.

Q: capacity of the vehicle.

P^*: decision maker's preference, $P^* \in [0, 1]$.

The decision variables used in the formulation of the VRPFD are described as follows:

y_{ik}: {if node i is served by vehicle k, $y_{ik} = 1$; otherwise, $y_{ik} = 0$}.

x_{ijk}: {if vehicle k moves from node i to node j, $x_{ijk} = 1$; otherwise, $x_{ijk} = 0$}.

Thus, the fuzzy reasoning constrained program model of the VRPFD is mathematically formulated as follows:

$$\min \quad \sum_{k=1}^{m} \sum_{i=0}^{n} \sum_{j=0}^{n} c_{ij} x_{ijk} \tag{6}$$

$$\text{Subject to} \quad \operatorname{pos}\left(\sum_{i \in N} D_i y_{ik} \le Q\right) \ge P^*, \quad \forall k \in K \tag{7}$$

$$\sum_{k \in K} y_{ik} = 1, \quad \forall i \in N \tag{8}$$

$$\sum_{i \in N} x_{ijk} = y_{jk}, \quad \forall j \in N,\ k \in K \tag{9}$$

$$\sum_{j \in N} x_{ijk} = y_{ik}, \quad \forall i \in N,\ k \in K \tag{10}$$

$$\sum_{j \in N} x_{ijk} = \sum_{j \in N} x_{jik} = y_{ik}, \quad \forall i \in N,\ k \in K \tag{11}$$

$$\sum_{k \in K} y_{0k} = m \tag{12}$$

$$\sum_{j \in N} x_{0jk} = \sum_{i \in N} x_{i0k} = 1, \quad \forall k \in K \tag{13}$$

$$\sum_{k \in K} \sum_{i \in S} \sum_{j \in S, j \ne i} x_{ijk} \le |S| - 1, \quad \forall S \subset N. \tag{14}$$

The object of the proposed VRPFD is to minimize the total distance. Constraint (7) ensures that all nodes are served within the vehicle's capacity at the values of the decision maker's preference. Constraint (8) ensures that each node is visited by one vehicle. Constraints (9) and (10) define the relationships between x_{ijk} and y_{ik}, respectively. Constraint (11) guarantees that a vehicle must enter and leave each node exactly once. Constraint (12) ensures that at most m vehicles are used. Constraint (13) ensures that vehicle routes start from the depot 0 and terminate at the same depot. Constraint (14) represents the subtour elimination constraint where $|S|$ stands for the cardinality of set S.

3. Hybrid Ant Colony Algorithm for VRPFD

The ant colony algorithm (ACA) is one of the most popular swarm-inspired methods in the field of computational intelligence. The first ACA was developed by Clolrni et al. [24]. It was successfully applied to the traveling salesman problem. The first ant system for the VRP was proposed by Bulleneimer et al. [25]. Doerner et al. [26] further improved this ant colony system using a savings-based heuristic. Recently, ACA has been applied to the VRP with different constraints, for example, Ellabib et al. [27], Gajpal and Abad [28], Yu et al. [29], and Fleming et al. [30]. By looking at success of above hybridised ant colony algorithms on VRP, we decided to develop hybrid ant colony algorithm (HACA) for VRPFD too.

Let Ω be the set of all candidate nodes in the dataset, let $U(h)$ be the set of nodes yet to be served by ant h, and let $\mathfrak{I}(h)$ be the set of nodes already served by ant h.

3.1. Transfer Probabilities. The probability p_{ij}^{h} that ant h chooses to serve node j having served node i is given by

$$p_{ij}^{h} = \begin{cases} \dfrac{\left[\tau_{ij}\right]^{\alpha} \times \left[\eta_{ij}\right]^{\beta}}{\sum_{r\in U(h)} \left[\tau_{ir}\right]^{\alpha} \times \left[\eta_{ir}\right]^{\beta}}, & j \in U(h) \\ 0, & \text{otherwise,} \end{cases} \tag{15}$$

where τ_{ij} is the pheromone density of edge (i, j); η_{ij} is the visibility of edge (i, j); α is the relative influence of the pheromone trails; and β is the relative influence of the visibility.

3.2. Pheromone Updating

3.2.1. Local Pheromone Updating Rule

Definition 1 (ant attraction). The ant attraction ∂/χ of edge (i, j) is the ratio of number ∂ of ants that have travelled edge (i, j) to the number χ of ants that have visited node i.

Each ant leaves constant quantity θ of pheromone on the edge it travels, and larger ant attraction χ/θ of edge (i, j) results in greater amount of ant travel on edge (i, j). Thus, more frequent local pheromone updating will result in a larger pheromone quantity τ_{ij} between all edges. The global searching of the ACO algorithm represents a handicap. To address this problem a local pheromone updating rule has been designed.

Let χ_h be the number of ants that visited node i before arrival of ant h at node i, and let ∂_h be the number of ants that travelled edge (i, j) before ant h travelled edge (i, j). The local pheromone update quantity of edge (i, j) caused by ant h can be calculated as

$$\Delta\tau_{ij}^{h}(t) = \begin{cases} \theta \times \left(1 - \dfrac{\partial_h}{\chi_h}\right), & \text{if ant } h \text{ traveled from node } i \text{ to } j \\ 0, & \text{otherwise.} \end{cases} \tag{16}$$

Let $\tau_{ij}^{h}(t)$ be the local pheromone quantity of edge (i, j) before updating; let $\tau_{ij}^{h}(t+1)$ be the local pheromone quantity of edge (i, j) after updating; and let ρ be the pheromone volatilization coefficient. Then, the local pheromone updating rule can be defined as

$$\tau_{ij}^{h}(t+1) = \rho \times \tau_{ij}^{h}(t) + \Delta\tau_{ij}(t), \quad \forall i,\ \forall j, i \neq j, \tag{17}$$

$$\Delta\tau_{ij}(t) = \sum_{h=1}^{m} \Delta\tau_{ij}^{h}(t). \tag{18}$$

3.2.2. Global Pheromone Updating Rule. If ant h has already served all n nodes, the global pheromone updating rule is employed. It is defined as

$$\tau_{ij}(t+n) = (1-\rho) \cdot \tau_{ij}(t) + \Delta\tau_{ij}(t), \tag{19}$$

$$\Delta\tau_{ij}(t) = \sum_{k=1}^{m} \Delta\tau_{ij}^{h}(t). \tag{20}$$

3.3. The Steps of HACA. The steps of the proposed HACA are depicted below.

Step 1 (algorithm initialization). (1) Set the values of the current iteration number nc, the maximum iteration number $\max ir$, the capacity value Q, the ant number m, and the decision maker's preference P^{*}. (2) Set all ants at the central depot 0, and let each ant start from the depot. (3) Let $Q_{vo} = 0$ be the initial occupied load of ant h, and the remaining capacity $Q_{vr} = Q - Q_{vo}$.

Step 2 (route construction). (1) Calculate the transfer probability p_{ij}^{h}. (2) Select node j according to the sequence of p_{ij}^{h} arranged in decreasing order. (3) If $\text{pos}(D_j \leq Q_{vr}) \geq P^{*}$, $j \in U(h)$, ant h must move to node j from the current node i, and the current node of ant h is changed to be j, $j \notin U(h)$, $j \in \mathfrak{I}(h)$, and the occupied load $Q_{vo} = Q_{vo} + d_j$; otherwise, ant h should return to the depot, $Q_{vo} = 0$, and move to the next node j. (4) Repeat this selection until $U(h) = \phi$.

Step 3 (pheromone updating). If $U(h) = \phi$, that is, ant h has already served all n nodes, the global pheromone updating rule is employed; otherwise, the local pheromone updating rule is employed.

Step 4 (judgment). If the total number of searching ants is smaller than m, return to step 2; otherwise, find the best solution by the path set $L = \{L_1, L_2, \ldots, L_m\}$ obtained with $\mathfrak{I}(h)$.

Step 5 (the 2-opt local search). (1) The obtained route is broken at random into three segments. (2) The middle segment must not contain the depot. (3) The route is then reconstructed by reversing the middle segment. (4) The route is updated whenever there is an improvement. (5) The process is repeated until there is no further improvement in the solution [28]. (6) If the new solution is better than the current solution, the new solution will replace the current solution.

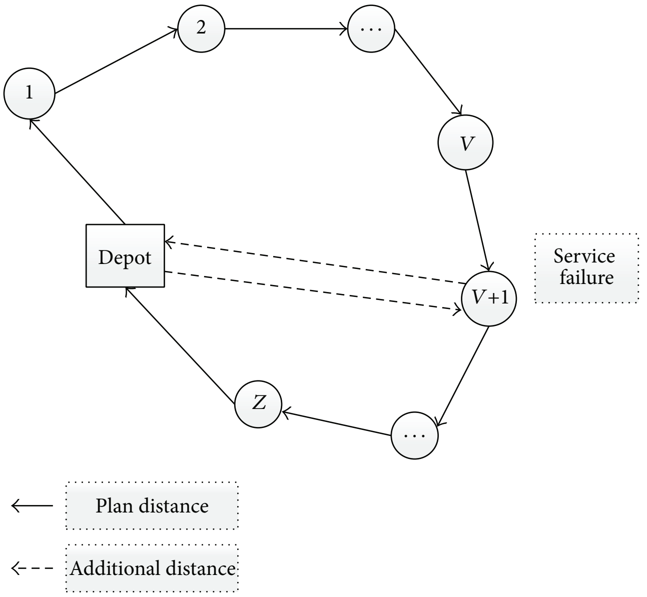


Figure 1: Service failure.

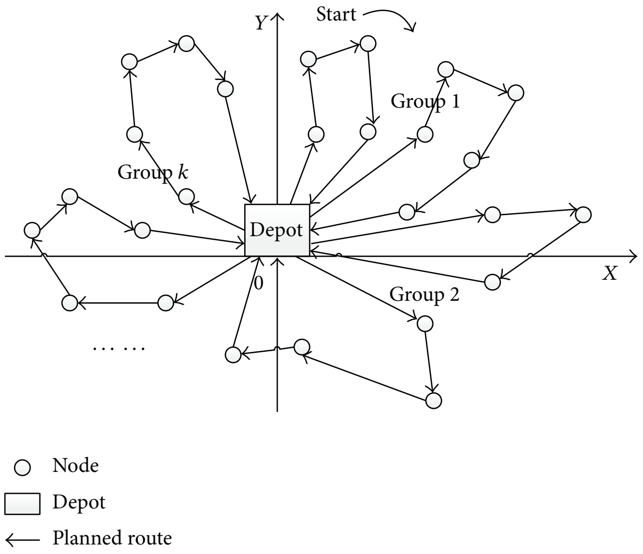


Figure 2: Vehicle grouping.

Step 6 (termination rule of the algorithm). If $nc < \max ir$, $nc = nc + 1$, return to step 2, and repeat the above steps; otherwise, terminate the HACA.

4. Two-Vehicle-Paired Loop Coordinated Strategy

As mentioned above, the VRPFD typically assumes that the "actual" value of a node's demand is known when the vehicle reaches the node, and the vehicle route is planned in advance. After serving v nodes, a vehicle might not be able to service the $v + 1$ node once it arrives due to its insufficient capacity. In such situations, if the vehicle uncoordinated strategy is employed, the vehicle must return to the depot, unload what it has picked up thus far, return to the node where it had a "service failure," and continue to service the remaining nodes (e.g., in raw milk collection systems). Thus, "additional distance" and "additional unloading times" are introduced due to the "service failure" (Figure 1). In contrast, there are also vehicles with surplus capacity after completing their own tasks; thus, "waste capacity" is created. All of these cases increase the logistics costs. To the authors' knowledge, few studies have considered the problem of how to effectively minimize the "additional distance" and "waste capacity," let alone reduce "additional unloading times," in the VRPFD.

After the optimal routes are obtained by the HACA (it is assumed that each route only served exactly by one vehicle), the two-vehicle-paired loop coordinated strategy (TVPLCS) is presented to minimize the "additional distance," "additional unloading times," and "waste capacity" in the VRPFD. The essence of the TVPLCS is that the vehicle with "surplus capacity" must help the vehicle with "insufficient capacity" according to the specified coordination rules after finish its own assigned task. The coordination rules of the TVPLCS are described as follows.

Coordination Rule 1. Assume that there are m planned vehicle routes and one depot 0 in two-dimensional coordinates. Put the depot on the origin of the two-dimensional coordinates. Starting from y-axis, divide the 2 adjacent routes into a coordinated group according to clockwise rotation (Figure 2). If the number of the routes is odd, there is one remaining route which is not assigned, and the vehicle will complete its own assigned task.

Coordination Rule 2. Each vehicle of the same coordinated group should finish its own assigned task first, and each vehicle first serves its "outer" nodes and then its "inner" nodes (Figure 3(a)). In this manner, if a vehicle has a "service failure," the "failure nodes" are near the other vehicle in the same coordinated group, thus promoting coordination between the two vehicles (Figure 3(b)).

Coordination Rule 2. If a vehicle completes its own task and has no surplus capacity, it should return to the depot and inform the other vehicle of its "task status."

Coordination Rule 4. If a vehicle completes its own task, has surplus capacity, and does not receive information from the other vehicle, the vehicle should wait and inform the other vehicle of its "task status" (Figure 4(a)). If the vehicle receives information from the other vehicle that the other vehicle is completing its own task, the first vehicle should return the depot (Figure 4(b)). If the vehicle receives information from the other vehicle that the other vehicle cannot complete its own task, the first vehicle should go to the node where the other vehicle had a "service failure" and continue to serve the remaining nodes. If the vehicle completes the remaining tasks, it should return the depot and inform the depot (Figure 4(c)); otherwise, the vehicle should return to the depot and inform the depot of its "remaining task status" (Figure 4(d)).

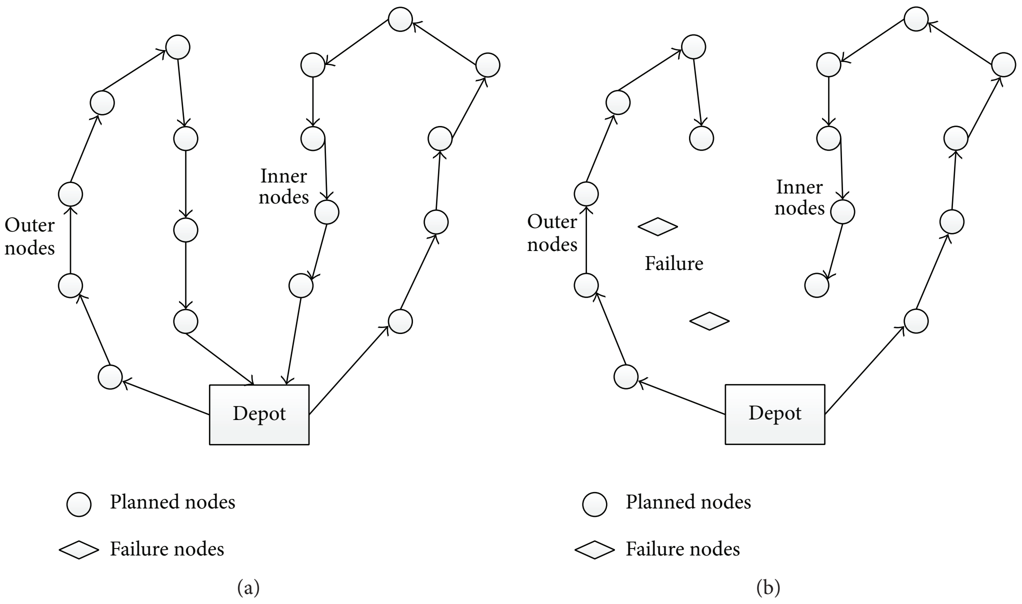


Figure 3: Coordination rule 2.

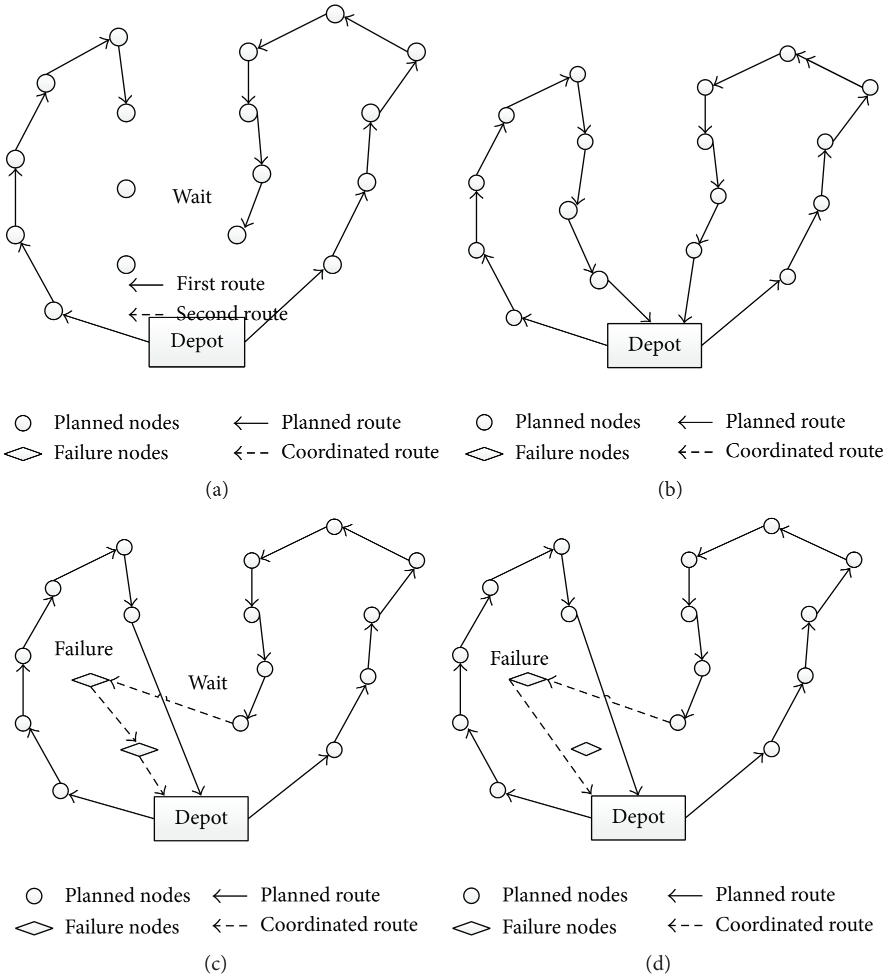


Figure 4: Coordination rule 4.

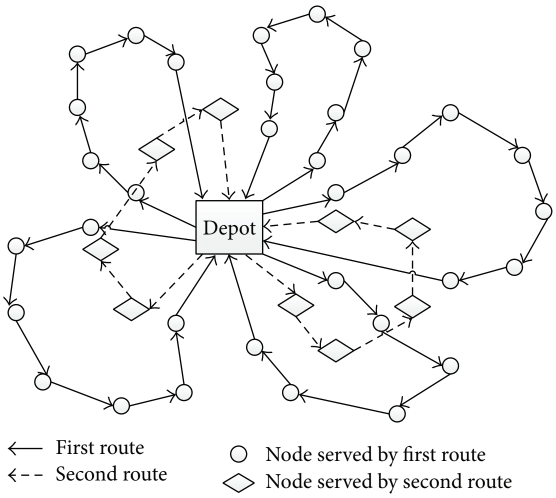


Figure 5: Coordination rule 5.

Coordination Rule 5. If all vehicles of all coordinated groups have returned to the depot and there are "remaining nodes" because of "service failure," the "remaining nodes" will be served by the vehicles of the second scheduling optimization, and the vehicle coordinated strategy will also be employed in the second service (Figure 5).

There are also some deficiencies in the TVPLCS, such as the "waiting" and "informing" between the two vehicles. "Waiting" means that if the vehicle completes its task and has surplus capacity, it must wait for the other vehicle's information. It may wait a long time due to different traffic circumstances, thus increasing the time cost. "Informing" means that when the vehicle completes its task, it must inform the other vehicle of its "task status." The "informing" problem can be easily solved using the advanced communication technology now available.

5. Experimental Results

The program for HACA was developed in Matlab 7.0. All computer experiments were performed on a PC (CPU 1.86 GHz, Memory 2 GB). The parameters for the HACA are set as follows: the ant number $m = 30$; the vehicle capacity value $Q = 150$ items; the maximum iteration number $\max ir = 200$; the pheromone volatilization coefficient $\rho = 0.9$; the relative influence of the pheromone trails $\alpha = 1$; the relative influence of the visibility $\beta = 2$; and the pheromone quantity $\theta = 15$.

Because the standard test instances for VRPFD are unavailable, the two-dimensional coordinates of the nodes and depot are generated randomly in [100 × 100] in this paper. The fuzzy demands of the nodes were also determined arbitrarily.

5.1. The Running Time of the HACA.

5.1. The Running Time of the HACA. In this experiment, the number of the test nodes is within the interval of 100–500 with a step of 100. Each instance runs 20 times. The average running time of each instance is used to be the solution. This is not real time problem and hence CPU times of HACA are acceptable. The results of running times are shown in Table 1.

Table 1: Running times of HACA.

Number of test nodes	100	200	300	400	500
Running time (seconds)	289.37	484.62	698.16	875.84	983.67

To facilitate the comparison, HACA is also simulated in several benchmark datasets. The source of the datasets is http://www.coin-or.org/SYMPHONY/branchandcut/VRP/data/#V. These datasets are modified to the VRPFD datasets by generating fuzzy demand. The fuzzy demand $D_i = (d_{i1}, d_{i2}, d_{i3})$ is randomly generated for each dataset, where the original dataset is used as d_{i2} for each node. In this simulation, solution obtained by HACA is compared with HPSOGA designed by Kuo et al. [13]. Each instance runs 20 times. The average running time of each instance is shown in Table 2. The results in Table 2 prove that HACA has promising performance in solving VRPFD. HACA outperforms HPSOGA for all dataset.

Table 2: Running times of HACA and HPSOGA.

Dataset	A-n33-k5	E-n51-k5	M-n101-k10	M-n151-k12
Running time of HACA (seconds)	20.16	81.24	296.71	437.62
Running time of HPSOGA (seconds)	20.64	88.09	388.32	635.59

5.2. The Efficiency of the TVPLCS. To determine efficiency of the TVPLCS we have developed an approach to estimate the "waste capacity," "additional distance," and "additional unloading times." The steps of the proposed calculation approach are described as follows.

Step 1. The method for processing fuzzy number into deterministic number proposed by the CA [31] is employed to estimate the "actual" demands d_i^a of each node.

Step 2. For each route planned by the HACA, the vehicle moves along the route, accumulating the sum Q^a of the "actual" demands of all nodes; that is, $Q^a = \sum_{i \in \text{route}} d_i^a$.

Step 3. If $Q^a \leq Q$, the vehicle can finish the task of this route, then the "waste capacity" is $Q - Q^a$; if $Q^a > Q$, the vehicle cannot finish the task of this route. The vehicle must return to the depot, unload what it has picked up thus far, return to the node where it had a "service failure," and continue to serve the remaining nodes. Now, we can calculate the "additional distance" and "additional unloading times" due to the "service failure." The "additional distance" is equal to length of second tour of the vehicle caused because of service failure in the first tour. The "additional unloading times" are the "service

Table 3: Travel distance results within 100 nodes.

P^*	TVPLCSHACA			HPSOGA		
	TPD	TAD	TD	TPD	TAD	TD
0.1	2,285.66	1,817.80	4,103.46	2,376.18	2,013.74	4,380.92
0.2	2,367.16	1,682.57	4,049.73	2,426.91	1,802.32	4,229.23
0.3	2,497.35	1,176.48	3,673.83	2,573.84	1,472.69	4,046.53
0.4	2,734.83	714.54	3,449.37	2,821.73	1058.36	3,880.09
0.5	2,610.92	673.27	3,284.19	2,783.41	952.71	3,736.12
0.6	3,016.73	283.24	3,298.97	3,133.28	509.16	3,642.44
0.7	3,236.51	0	3,236.51	3,292.41	407.65	3,700.06
0.8	3,347.26	0	3,347.26	3,384.74	368.52	3,753.26
0.9	3,469.80	0	3,449.80	3,794.37	0	3,794.37
1	3,577.56	0	3,577.56	3,836.91	0	3,836.91

failure” times. The “total distance” is the sum of the “total planned distance” and “total additional distance.”

5.2.1. Test of the Travel Distance. For simplicity and perceptual intuition, we only test 100 nodes. To show the efficiency of the proposed approaches, we compare our TVPLCS based on the HACA (TVPLCSHACA) with the hybrid particle swarm optimization with genetic algorithm (HPSOGA) proposed by Kuo et al. [13]. HPSOGA employs vehicle uncoordinated strategy, and it is designed for VRPFD. This is to say, coordinating strategies/rules defined in Section 4 all were used in the experiment. At first, the coordination rule 1 is used. Then the coordination rule 2 is used. Also, the coordination rule 3 and coordination rule 4 are used; if the task is not finished, the coordination rule 5 must be used; else, the coordination rule 5 may not be used.

All test are calculated according to the decision maker’s confidence P^*, where P^* varies within the interval of 0-1 with a step of 0.1. The average computational results of 10 times are calculated.

The “total planned distance” (TPD), “total additional distance” (TAD), and “total distance” (TD) are calculated. The results of TVPLCSHACA and HPSOGA are listed in Table 3.

From the results in Table 3, we can see the following: (1) with regard to TPD the result gained from the HACA is smaller than the result obtained by the HPSOGA. When $P^* = 0.9$, the former is 9.35% less than the latter. Even more, the former is 10.33% less than the latter while $P^* = 1$. (2) As mentioned above, TAD is due to the “service failures.” The TAD result gained from the TVPLCSHACA is also smaller than the result obtained by the HPSOGA. Especially, when $0.4 \le P^* \le 0.6$, the former is close to almost half of the latter. This is to say, the TVPLCSHACA is useful for VRPFD, and TVPLCS can effectively reduce the TAD. When $P^* \ge 0.7$, TAD gained from the HACA is 0, which means that there is no “service failure” in each route. However, TAD gained from the HPSOGA is 0 while $P^* \ge 0.9$. That is, our approach is better than HPSOGA. (3) The TD of TVPLCSHACA is also smaller than the result obtained by the HPSOGA. The results can show the effectiveness of our proposed approaches.

Table 4: Results of the total vehicle number, waste capacity, and additional unloading times.

P^*	TVPLCSHACA				HDEA			
	PVN	TVN	TWC	TAUT	PVN	TVN	TWC	TAUT
0.1	13	22	0	9	13	26	166	13
0.2	13	21	0	8	14	25	182	12
0.3	14	21	0	7	14	25	186	11
0.4	15	21	0	6	15	25	198	9
0.5	16	20	51	4	16	25	223	8
0.6	17	20	87	3	17	24	257	7
0.7	18	18	125	0	18	23	261	5
0.8	18	18	114	0	19	23	273	4
0.9	19	19	89	0	20	20	127	0
1	19	19	81	0	20	20	118	0

5.2.2. Test of the Total Vehicle Number, Waste Capacity, and Additional Unloading Times. Also, we test the same 100 nodes which tested above. To show the efficiency of the proposed TVPLCS, we compare our TVPLCSHACA with the HDEA proposed by Erbao and Mingyong [12]. The algorithm employs vehicle uncoordinated strategy, and it is designed for VRPFD. All test are calculated according to the decision maker’s confidence P^*, where P^* varies within the interval of 0-1 with a step of 0.1. The average computational results of 10 times are calculated. The “plan vehicle number” (PVN), “total vehicle number” (TVN), “total waste capacity” (TWC), and “total additional unloading times” (TAUT) are tested. The results of our TVPLCSHACA and the results of HDEA (HDEA employs vehicle uncoordinated strategy) are listed in Table 4. In Table 4, capacity unit is item.

The results in Table 4 indicate the following: (1) For PVN, the value gained by HACA is close to the result planed by HDEA, and the former is better than the latter. (2) With regard to TAUT, the value gained by TVPLCSHACA is much better than the result calculated by HDEA. When $P^* \le 0.3$, the number of the former is 30% less than that of the latter. When $0.4 \le P^* \le 0.6$, the number of the former is 50% less than that of the latter. When $P^* \ge 0.7$, TAUT gained by TVPLCSHACA is 0. However, TAUT gained by HDEA is 0 till $P^* \ge 0.9$. So, TVPLCS can effectively cut down “additional unloading times” in VRPFD, especially when the value of decision maker’s preference is relatively smaller. (3) About TWC, when $P^* \le 0.4$, all numerical values gained by TVPLCSHACA are 0, but the results of HDEA are close to 200. In other words, when a vehicle has surplus capacity, the vehicle should serve the “failure nodes” of the other failure route according to the specified TVPLCS rules. TVPLCS could be usefully employed in reduction of the “waste capacity” in VRPFD, especially when the value of decision maker’s preference is relatively smaller. (4) As for TVN, the value gained by TVPLCSHACA based on the HACA is less than the result planed by HDEA. When $P^* \le 0.8$, the former is nearly 25% less than the latter. Thus, the TVPLCS is very useful for the VRPFD, especially when P^* is given a relatively smaller value. That is, the decision maker

Table 5: Results of total distance, vehicle number, and logistics cost.

P^*	TVPLCSHACA			HDEA		
	TD	TVN	TLC	TD	TVN	TLC
0.1	4,103.46	22	31,517.3	4,380.92	26	34,904.6
0.2	4,049.73	21	30,748.65	4,229.23	25	33,646.15
0.3	3,673.83	21	28,869.15	4,046.53	25	32,732.65
0.4	3,449.37	21	27,746.85	3,880.09	25	31,900.45
0.5	3,284.19	20	26,420.95	3,736.12	25	31,180.6
0.6	3,298.97	20	26,494.85	3,642.44	24	30,212.2
0.7	3,236.51	18	25,182.55	3,700.06	23	30,000.3
0.8	3,347.26	18	25,736.3	3,753.26	23	30,266.3
0.9	3,449.80	19	26,749	3,794.37	20	28,971.85
1	3,577.56	19	27,387.8	3,836.91	20	29,184.55

has an insatiable appetite for risk and wants to serve more nodes with each vehicle.

5.3. Determining the Reasonable Value of the Decision Maker's Preference. Now, we determine the reasonable value of the decision maker's preference based on the results listed in Tables 3 and 4. The TD in Table 3 and the TVN in Table 4 are selected to calculate the total logistics cost (TLC) for VRPFD. In this paper, it is assumed that the freight of TD is 5 yuan RMB per unit distance, and the fixed charge of TVN is 500 yuan RMB per vehicle. TLC is the sum of TD and TVN. The results of TLC are listed in Table 5.

From the results in Table 5 we can see the following: (1) When decision maker's preference $P^* \geq 0.6$, the decision maker is risk averse and aims to minimize service failures. The optimal value of P^* should be 0.6 or 0.7 according to TD. The best value of P^* should be 0.7 according to TLC and TVN. However, it is not easy for us to obtain a highly credible decision maker's preference due to the variability of customer demands in practice, especially when the operations of a distributing system are only starting up. (2) While $P^* \leq 0.5$, the decision maker is risk preference and wants to serve more nodes with each vehicle. All values of TD, TVN, and TLC tell us that the best value of P^* is 0.5. (3) The reasonable value of the decision maker's preference deduced from TVPLCSHACA is exactly the same as that inferred from HDEA.

6. Conclusions

This paper contributes to the research on the VRPFD in the following respects: (1) The fuzzy reasoning constrained program model and HACA are designed to optimize the vehicle routes, and the most appropriate values for the decision maker's confidence level P^* were obtained by simulation. That is, if decision maker is risk averse and aims to ensure service, the best value of P^* should be 0.7. If the decision maker has an insatiable appetite for risk and wants to serve more nodes with each vehicle, the best value of P^* should be 0.5. (2) In particular, the TVPLCS is presented to reduce the "additional distance," "unloading times," and the "waste capacity" caused by "service failure." Numerical examples are presented to demonstrate the effectiveness of our proposed approaches. Particularly, the TVPLCS is very useful for the VRPFD when the decision maker has an insatiable appetite for risk, especially when the operations of a distributing system are only starting up and the customers' demands are difficult to estimate.

For the future research, we may consider other nondeterministic side constraints in VRPFD, such as stochastic vehicle travel time, important customer with emergent service, and time window constraint that should be considered in order to fit the practical applications.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This research has been partially supported by grants from the National Natural Science Foundation of China (no. 71373216, no. 71471149 and no. 71222108), Major Project of the National Social Science Foundation of China (no. 15ZDB153), grants from the National Philosophy and Social Science Foundation of China (no. 15AJL006), Excellent Youth Project of Hunan Provincial Education Department of China (no. 15B131), and Key R & D program of Hunan Provincial Science and Technology Department of China (no. 2015ZK3049).

References

[1] G. B. Dantzig and J. H. Ramser, "The truck dispatching problem," *Management Science*, vol. 6, no. 1, pp. 80–91, 1959.

[2] G. Clarke and J. W. Wright, "Scheduling of vehicles from a central depot to a number of delivery points," *Operations Research*, vol. 12, no. 4, pp. 568–581, 1964.

[3] M. M. Solomon, "Algorithms for the vehicle routing and scheduling problems with time window constraints," *Operations Research*, vol. 35, no. 2, pp. 254–265, 1987.

[4] G. Laporte, "The vehicle routing problem: an overview of exact and approximate algorithms," *European Journal of Operational Research*, vol. 59, no. 3, pp. 345–358, 1992.

[5] M. A. Figliozzi, "Planning approximations to the average length of vehicle routing problems with time window constraints," *Transportation Research Part B: Methodological*, vol. 43, no. 4, pp. 438–447, 2009.

[6] R. Sprenger and L. Mönch, "A methodology to solve large-scale cooperative transportation planning problems," *European Journal of Operational Research*, vol. 223, no. 3, pp. 626–636, 2012.

[7] V. Pillac, M. Gendreau, C. Guéret, and A. L. Medaglia, "A review of dynamic vehicle routing problems," *European Journal of Operational Research*, vol. 225, no. 1, pp. 1–11, 2013.

[8] G. Kou, D. Ergu, and J. Shang, "Enhancing data consistency in decision matrix: adapting Hadamard model to mitigate judgment contradiction," *European Journal of Operational Research*, vol. 236, no. 1, pp. 261–271, 2014.

[9] G. Kou, Y. Lu, Y. Peng, and Y. Shi, "Evaluation of classification algorithms using MCDM and rank correlation," *International*

Journal of Information Technology & Decision Making, vol. 11, no. 1, pp. 197–225, 2012.

[10] E. Cao and M. Lai, “The open vehicle routing problem with fuzzy demands,” *Expert Systems with Applications*, vol. 37, no. 3, pp. 2405–2411, 2010.

[11] D. J. Bertsimas, “A vehicle routing problem with stochastic demand,” *Operations Research*, vol. 40, no. 3, pp. 574–585, 1992.

[12] C. Erbao and L. Mingyong, “A hybrid differential evolution algorithm to vehicle routing problem with fuzzy demands,” *Journal of Computational and Applied Mathematics*, vol. 231, no. 1, pp. 302–310, 2009.

[13] R. J. Kuo, F. E. Zulvia, and K. Suryadi, “Hybrid particle swarm optimization with genetic algorithm for solving capacitated vehicle routing problem with fuzzy demand—a case study on garbage collection system,” *Applied Mathematics and Computation*, vol. 219, no. 5, pp. 2574–2588, 2012.

[14] G. Kou and C. Lin, “A cosine maximization method for the priority vector derivation in AHP,” *European Journal of Operational Research*, vol. 235, no. 1, pp. 225–232, 2014.

[15] G. Kou, Y. Peng, and G. Wang, “Evaluation of clustering algorithms for financial risk analysis using MCDM methods,” *Information Sciences*, vol. 275, pp. 1–12, 2014.

[16] M. Allahviranloo, J. Y. J. Chow, and W. W. Recker, “Selective vehicle routing problems under uncertainty without recourse,” *Transportation Research Part E: Logistics and Transportation Review*, vol. 62, pp. 68–88, 2014.

[17] Z.-H. Hu, J.-B. Sheu, L. Zhao, and C.-C. Lu, “A dynamic closed-loop vehicle routing problem with uncertainty and incompatible goods,” *Transportation Research Part C: Emerging Technologies*, vol. 55, pp. 273–297, 2015.

[18] X. Liu, G. G. He, and W. W. Gao, “Modeling and algorithm for the multiple vehicles coordinated stochastic vehicle routing with time-constrain,” *System Engineering*, vol. 23, pp. 105–109, 2005 (Chinese).

[19] C. K. Y. Lin, “A cooperative strategy for a vehicle routing problem with pickup and delivery time windows,” *Computers and Industrial Engineering*, vol. 55, no. 4, pp. 766–782, 2008.

[20] J. S. Shang and C. K. Cuff, “Multicriteria pickup and delivery problem with transfer opportunity,” *Computers & Industrial Engineering*, vol. 30, no. 4, pp. 631–645, 1996.

[21] J. Yang, P. Jaillet, and H. Mahmassani, “Real-time multivehicle truckload pickup and delivery problems,” *Transportation Science*, vol. 38, no. 2, pp. 135–148, 2004.

[22] X. P. Hu, L. J. Sun, and L. L. Liu, “A PAM approach to handling disruptions in real-time vehicle routing problems,” *Decision Support Systems*, vol. 54, no. 3, pp. 1380–1393, 2013.

[23] B. Liu, “Uncertain theory: an introduce to its axiomatic foundations,” in *Fuzzy Variables, Fuzzy Sets and Systems*, S. Nahmias, Ed., vol. 1, pp. 97–110, Springer, Berlin, Germany, 2004.

[24] A. Clolrni, M. Dorigo, and V. Maniezzo, “Distributed optimization by ant colonies,” in *Proceedings of the European Conference on Artificial Life (ECAL ’91)*, pp. 134–142, Paris, France, 1991.

[25] B. Bulleneimer, R. F. Hartl, and C. Strauss, “Strauss, applying the ant system to the vehicle routing problem,” in *Proceedings of the 2nd International Conference on Metaheuristics*, pp. 1–12, 1997.

[26] K. Doerner, M. Gronalt, R. F. Hartl, M. Reimann, C. Strauss, and M. Stummer, “Savings ants for the vehicle routing problem,” in *Application of Evolutionary Computing*, pp. 11–20, Springer, Berlin, Germany, 2002.

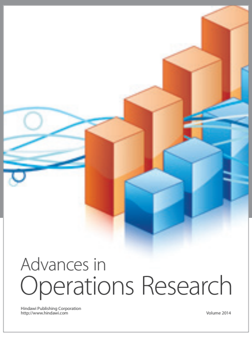
[27] I. Ellabib, P. Calamai, and O. Basir, “Exchange strategies for multiple Ant Colony System,” *Information Sciences*, vol. 177, no. 5, pp. 1248–1264, 2007.

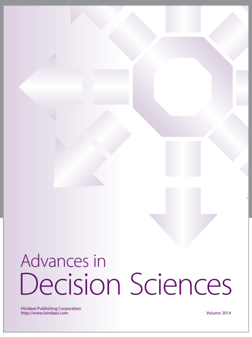
[28] Y. Gajpal and P. L. Abad, “Multi-ant colony system (MACS) for a vehicle routing problem with backhauls,” *European Journal of Operational Research*, vol. 196, no. 1, pp. 102–117, 2009.

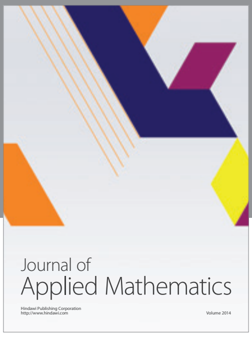
[29] B. Yu, Z. Z. Yang, and B. Z. Yao, “A hybrid algorithm for vehicle routing problem with time windows,” *Expert Systems with Applications*, vol. 38, no. 1, pp. 435–441, 2011.

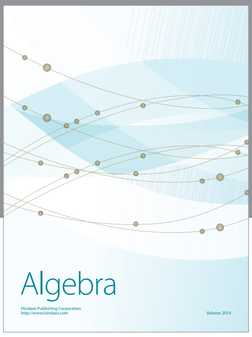
[30] C. L. Fleming, S. E. Griffis, and J. E. Bell, “The effects of triangle inequality on the vehicle routing problem,” *European Journal of Operational Research*, vol. 224, no. 1, pp. 1–7, 2013.

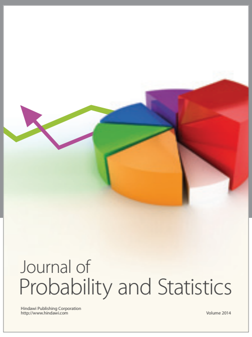
[31] C.-C. Chou, “The canonical representation of multiplication operation on triangular fuzzy numbers,” *Computers & Mathematics with Applications*, vol. 45, no. 10-11, pp. 1601–1610, 2003.


Advances in
Operations Research

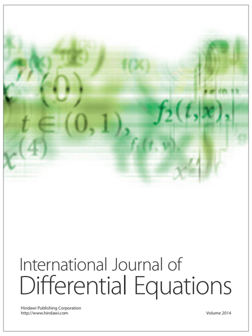

Advances in
Decision Sciences


Journal of
Applied Mathematics

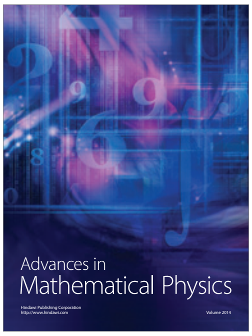

Algebra

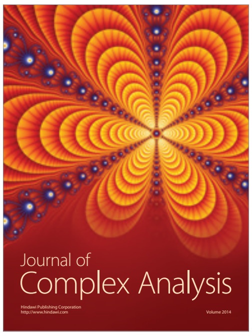

Journal of
Probability and Statistics

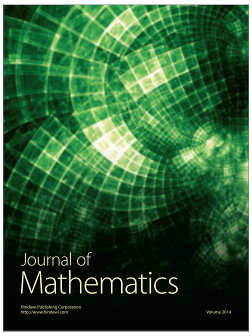
The Scientific
World Journal

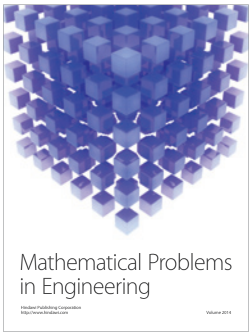

International Journal of
Differential Equations

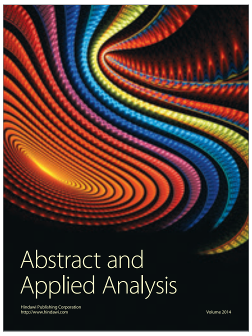
International Journal of
Combinatorics

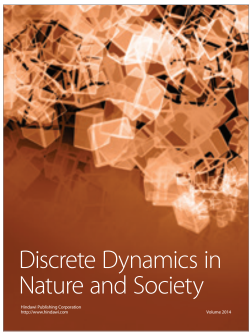

Advances in
Mathematical Physics

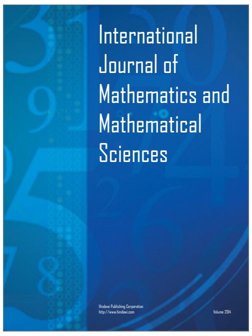

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

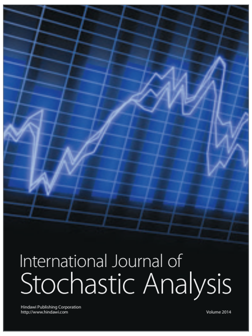

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization