Hindawi Publishing Corporation
Discrete Dynamics in Nature and Society
Volume 2016, Article ID 7841297, 10 pages
http://dx.doi.org/10.1155/2016/7841297

Research Article

A Three-Stage Saving-Based Heuristic for Vehicle Routing Problem with Time Windows and Stochastic Travel Times

Zheng Wang and Chunyue Zhou

School of Software Technology, Dalian University of Technology, Dalian 116024, China

Correspondence should be addressed to Zheng Wang; drwz@dlut.edu.cn

Received 7 October 2015; Revised 10 January 2016; Accepted 26 January 2016

Academic Editor: Alicia Cordero

This paper presents a saving-based heuristic for the vehicle routing problem with time windows and stochastic travel times (VRPTWSTT). One of the basic ideas of the heuristic is to advance the latest service start time of each customer by a certain period of time. In this way, the reserved time can be used to cope with unexpected travel time delay when necessary. Another important idea is to transform the VRPTWSTT to a set of vehicle routing problems with time windows (VRPTW), each of which is defined by a given percentage used to calculate the reserved time for customers. Based on the above two key ideas, a three-stage heuristic that includes the "problem transformation" stage, the "solution construction" stage, and the "solution improvement" stage is developed. After the problem transformation in the first stage, the work of the next two stages is to first construct an initial solution for each transformed VRPTW by improving the idea of the classical Clarke-Wright heuristic and then further improve the solution. Finally, a number of numerical experiments are conducted to evaluate the efficiency of the described methodology under different uncertainty levels.

1. Introduction

Many real-life logistics and distribution problems can be attributed to a kind of vehicle routing problems with time windows which has been widely studied. Various kinds of algorithms have been developed to deal with the problem [1], which narrows the gap between theory and practical application. In the extant theory approaches, it is usually assumed that the travel time between a pair of points is a definite constant. However, in real traffic conditions of city areas, there are a large number of uncertainties resulting in that the travel time of a pair of points will change with the different traffic conditions and thus is not always the same. Ignoring the travel time fluctuations when developing route plans for pickup and/or delivery vehicles can result in inefficient and suboptimal solutions [2]. The extant researches that incorporate variant travel times are very limited in the assumption of travel time distribution and computational efficiency. Therefore, this study is undertaken to better handle the stochasticity of travel times.

The VRPSTT is arguably one of the most challenging and practical variants of the VRP [3], and only a few studies in extant literature have addressed the VRPSTT. Laporte et al. proposed a variant of VRPSTT chance-constrained programming model and compensation model [4], where a penalty function was introduced to prorate the delay time. Park and Song subsequently constructed three new heuristic algorithms based on an extension of VRP algorithms [5]. Kenyon and Morton developed two stochastic programming models [6]. The first model minimizes the expected completion time, whereas the second model maximizes the probability of completing the project by a prespecified deadline. More recently, Lecluyse et al. introduced the variability in traffic flow into the model [7], which was used to evaluate the routes based on the uncertainty involved. At the same time, the objective function was extended by standard deviation of the travel times. Connors and Sumalee [8] and Chen and Zhou [9] studied the stochasticity of travel times from the view of travelers' equilibrium. Zhang et al. studied the stochastic travel time vehicle routing problem with simultaneous pickups and deliveries and developed a new scatter search approach for the problem by incorporating a new chance-constrained programming method [3]. Wang and Lin presented a flexible solution methodology for the capacitated vehicle routing problem with stochastic travel times [10]. Although these studies have advanced the research on VRPSTT, practical algorithms and solution approaches are still needed because the stochasticity of travel times poses a very challenging combinatorial optimization problem.

To solve the VRPTWSTT, we present a three-stage saving-based heuristic, in which the issue of solving VRPTWSTT is transformed into a new issue that consists of solving multiple VRPTW. Also, a number of experiments are conducted to test the efficiency of the heuristic. The remainder of this paper is arranged as follows. First, the VRPTWSTT is described and analyzed in Section 2. Second, the main solution framework and detailed solution procedures are developed in Section 3. After the experimental settings and the results are presented in Section 4, conclusions are provided in the end.

2. Problem Description

This paper focuses on the VRPTWSTT which can be described in the following.

Let $G = (V, E)$ be a graph, where $V = \{v_0, v_1, \ldots, v_n\}$ is the set of customer nodes $\{v_1, \ldots, v_n\}$ and depot node v_0 and $E = \{(v_i, v_j) \colon v_i, v_j \in V, i \neq j\}$ is the set of edges. Each customer node v_i $(i = 1, \ldots, n)$ is associated with a positive demand q_i, a service time sv_i, and a time window $[et_i, lt_i]$ with $0 \leq et_i \leq lt_i$, where et_i and lt_i denote the earliest service start time and the latest service start time that customer i has preferred. Each edge $(v_i, v_j) \in E$ $(i \neq j)$ is associated with a positive stochastic travel time t_{ij} whose distribution function is known and can follow any known statistical distribution, either theoretical or empirical, whose expected value exists. Define K as a set of identical vehicles available to deliver goods from the depot to the customers. Each vehicle has the same capacity C $(C \gg \max\{q_i/1 \leq i \leq n\})$. Some additional constraints associated with the problem are as follows:

(1) Each vehicle can perform only one route in a scheduling period (e.g., a day).

(2) Each customer node can be visited only once by exactly one vehicle.

(3) All vehicles begin and end their routes at the depot node v_0.

(4) No vehicle can be loaded exceeding its maximum capacity C.

(5) Let at_i be a vehicle's arrival time at customer i if the vehicle services customer i and let bt_i be the service beginning time of customer i. Then the two rules defined in formula (1) must be observed: the vehicle has to wait until the time et_i if it arrives earlier than et_i; otherwise, the vehicle can immediately begin its service when it arrives at customer i. If $at_i > lt_i$, a penalty that is charged to the logistics service provider will be incurred and customer i is called a "poorly serviced customer." The penalty can be considered as compensation for customer i as a result of delayed service:

$$bt_i = \begin{cases} et_i, & \text{if } at_i < et_i, \\ at_i, & \text{if } at_i \geq et_i. \end{cases} \tag{1}$$

It should be noted that the problem addressed is essentially different from the vehicle routing problem with soft time windows (VRPSTW), in which customer time window limitations may be violated even in an original schedule so that the number of vehicles required and/or the total travel distance can be significantly reduced. In the VRPTWSTT, customer time window limitations must be satisfied in original routing plans given expected values for all edges but may be violated during the plan execution process, because there are many possibilities that the estimated travel time in the original plan may be delayed in an actual travel process.

(6) Once a routing plan is made, vehicles must run according to the plan no matter what disruption events occur until they return to depot in order to meet delivery commitments for customers.

The route of vehicle k, which is defined as $(v_0^k, v_1^k, \ldots, v_n^k, v_{n+1}^k)$, where v_0^k and v_{n+1}^k represent the depot and the other nodes represent customers, is a cycle starting and ending at the depot and serving a given subset of sequential customers. Let $x_{ij}^k = 1$ if the edge (v_i, v_j) is included in the route of vehicle k $(v_0^k, \ldots, v_i^k, v_j^k, \ldots, v_{n+1}^k)$ or 0 otherwise. The route of vehicle k is called "deficient route" if there is at least one poorly serviced customer among all its serviced customers.

The VRPTWSTT studied in this paper is a single-objective problem in which the total cost, including the travel cost of employed vehicles and the penalty cost of all poorly serviced customers, should be minimized. The objective function for VRPTWSTT can be defined as formula (2), where c_1 and c_2 are used to transform the travel time and the vehicle's lag time of poorly serviced customers to travel cost and penalty cost, respectively. Generally, $c_2/c_1 > 1$, implying a penalty of violating the customer time window limitation, and hence formula (2) can be naturally simplified to formula (3), where $c = c_2/c_1$ indicating a penalty coefficient when the customer time window limitation is violated. The value of c can be much larger than 1, for example, 10, 20:

$$z_0 = c_1 \sum_{k\in K} \sum_{(v_i,v_j)\in E} t_{ij} x_{ij}^k + c_2 \sum_{v_j \in V/v_0} \max\left\{bt_j - lt_j, 0\right\}, \tag{2}$$

$$z = \sum_{k\in K} \sum_{(v_i,v_j)\in E} t_{ij} x_{ij}^k + c \sum_{v_j \in V/v_0} \max\left\{bt_j - lt_j, 0\right\}. \tag{3}$$

Formula (3) can also be understood as the total cost when $c_1 = 1$. Thus the total cost contains two components, the travel cost defined as $\sum_{k\in K} \sum_{(v_i,v_j)\in E} t_{ij} x_{ij}^k$ and the penalty cost defined as $c \sum_{v_j \in V/v_0} \max\{bt_j - lt_j, 0\}$. Let $\mathrm{ps}_i = 1$ if customer i is a poorly serviced customer or 0 otherwise.

Then the reliability of a solution can be defined as $1 - (\sum_{v_i \in V/v_0} \mathrm{ps}_i)/(|V|-1)$, indicating what percentage of the total number of customers the non-poorly serviced customers account for, and it is an important index in analyzing a solution.

Note that t_{ij} in formula (3) is not a constant but a stochastic value whose statistical distribution and expected value $E(t_{ij})$ are known. Thus the z value depends not only on the routing solution, indicated by x_{ij}^k and bt_i, but also on the actual travel times, indicated by t_{ij}. Different travel times have different z values even for the same routing solution. For example, any accidental travel time delay may result in large z value while the unobstructed traffic situation can lead to small z value. In order to exclude the impact of accidental factors, the objective of the problem can be described as minimizing the expected z value, which is formulated and transformed in

$$\begin{aligned} \min E(z) &= E\left\{\sum_{k \in K} \sum_{(v_i,v_j) \in E} t_{ij} x_{ij}^k + c \sum_{v_j \in V/v_0} \max\left\{bt_j - lt_j, 0\right\}\right\} \\ &= \sum_{k \in K} \sum_{(v_i,v_j) \in E} E\left(t_{ij}\right) x_{ij}^k \\ &\quad + c \sum_{v_j \in V/v_0} E\left\{\max\left\{bt_j - lt_j, 0\right\}\right\}. \end{aligned} \tag{4}$$

3. The Three-Stage Saving-Based Heuristic

The VRPTWSTT is NP-hard because it can be reduced to the general VRP, which is a well-known NP-hard problem [11]. Furthermore, according to the problem's definition, how to balance the expected travel cost and the expected penalty cost is an important factor in constructing a solution. In general, a solution with less expected travel cost usually has more expected penalty cost. This is mainly because a route in a solution with less expected travel cost is often fully engaged in delivery tasks and hence there is often not much flexible time left to cope with unexpected transportation congestion. Once the unexpected event occurs, the penalty cost will probably be incurred. Therefore, an algorithm that is devoted to minimizing the expected total cost $E(z)$ should be able to integrate the stochasticity of travel time and generate a solution that can well balance the expected travel cost and the expected penalty cost and hence minimize the total cost. As a result of the high complexity level of the VRPTWSTT, solution techniques capable of producing high-quality solutions in a limited time, that is, heuristics, are of prime importance.

3.1. Main Framework. Our heuristic for the VRPTWSTT consists of three stages: the "problem transformation" stage, the "solution construction" stage, and the "solution improvement" stage. The objective of the first stage is to transform the VRPTWSTT into multiple VRPTW in which the latest service time of each customer is advanced by a period of time in order to cope with unexpected travel time delays. As a result of the problem transformation, penalty costs of solutions to the transformed VRPTW can be greatly reduced. But on the other hand, if too much time is reserved, travel costs will also increase rapidly. Therefore, another problem of how much time should be reserved for each customer under different routing schemes in order to minimize the expected total costs is derived from the first stage. To solve the problem, we develop the next two stages to first construct an initial solution for each transformed VRPTW and then to further improve the solution. Finally, for all transformed problems, we can get their improved solutions, the best of which indicates the best balance between the travel cost and the penalty cost and will be reported to decision-makers.

The proposed three-stage heuristic (TSH) is described in Algorithm 1, where procedures of the three stages are described in the following subsections, respectively.

3.2. The Problem Transformation Stage. In the problem transformation stage, the VRPTWSTT is transformed into a deterministic problem which is different from VRPTWSTT in the following two points.

(1) The stochastic travel time between any pair of nodes is replaced with its expected value according to its statistical distribution.

(2) The latest service start time of each customer is supposed to be advanced by a period of time, and thus vehicles should arrive at customers no later than their advanced times.

The reason for the second point comes from the fact that decision-makers of logistics service providers are always reluctant to bear the consequences of penalties but would rather schedule vehicles to arrive at customers earlier than their latest service start time in the original plans so that there are enough reserved times left for vehicles to cope with any unexpected travel time delay during practical travel process, such as road congestion. Otherwise, once the travel time delays are encountered, the penalty cost incurred will be far greater than the cost of earlier arrivals.

Furthermore, it can be easily concluded from common sense that the earlier the arrivals at customers, the smaller the penalty cost but the greater the travel cost. This conclusion can be also validated from our experiments in Section 4. Therefore, how long should be advanced for each customer so that the total cost can be minimized is an important problem.

To decide the advanced time for each customer, we use the variable r ($0 < r \leq 1$), the loop variable in the main framework of Algorithm 1, to represent the advanced percentage of time and calculate the new latest service start time for each customer, for example, customer i, by the formula $lt_i' = lt_i - r \times \mathrm{avgLen}_i$, where avgLen_i represents the average route length from all the other nodes to customer i. Figure 1 is an illustration of advancing lt_i to lt_i' for customer i.

Note that lt_i represents the latest time that a vehicle should arrive at customer i without incurring any penalty cost. However, in our transformation version of the problem, lt_i will be advanced to lt_i' so that vehicles should arrive at

```
Set $Solutions = \emptyset$;
For $r = r_0$ to $r_{\max}$ step $r_{\text{step}}$, do:
    Transform VRPTWSTT to a deterministic VRPTW (Section 3.2);
    Solve the VRPTW($r$) by using an improved saving-based heuristic in order to get an
    initial solution $IS$ (Section 3.3);
    Improve the initial solution $IS$ by a solution improvement procedure and then multiple
    solutions can be obtained, the minimum expected total cost of which is inserted into
    the set $Solutions$ (Section 3.4);
Next $r$;
Select and return the solution that has the minimum expected total cost from $Solutions$.
```

Algorithm 1: Main framework of the suggested heuristic TSH.

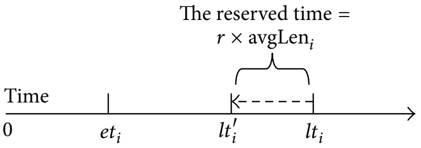


Figure 1: An illustration of advancing lt_i to lt_i' for customer i.

customer i before lt_i' (not lt_i) in the original plan and the time that is equal to $r \times \text{avgLen}_i$ ($lt_i - lt_i'$) is reserved to cope with any unexpected travel time delay during practical travel process of vehicles. In the formula $r \times \text{avgLen}_i$, r is a percentage of the average route length, avgLen_i, from all the other nodes to customer i and its value may be 0.1, 0.2, and so forth. The rationale of the formula comes from the observation that the longer the average route length is, the more likely it is to have accidents during the travel process of vehicles and thus the greater the expected penalty cost is. Meanwhile, our heuristic is essentially a simulation-based method, in which various percentages r are tried and the one that generates the solution with the least total cost will be reported to decision-makers.

It should also be noted that we do not guarantee lt_i' must be larger than et_i. There are possibilities that $lt_i' \le et_i$. Once $lt_i' \le et_i$, the vehicle that serves customer i also needs to arrive at customer i before lt_i' in the original plan. And in the practical travel process, if the vehicle arrives at customer i earlier than et_i, the vehicle also needs to wait until the time et_i.

3.3. The Solution Construction Stage. In this subsection, a new procedure is presented to construct an initial solution to the transformed problem. The procedure is based on the classical Clarke and Wright Savings (CWS) algorithm [10]. Algorithm 2 describes the procedure, which changes the saving selection strategy of the classical CWS to a long-term perspective.

The concept of saving in this paper expresses the saving of the travel cost and penalty cost obtained by joining two routes into one route.

There is a deficiency in the saving selection work of the CWS, which prevents the CWS from generating high-quality solutions. This is because the saving selection strategy of the CWS is a greedy one, which suggests selecting customers pairs in the descending order of saving values and hence may not accurately identify the best customers pair in some cases. To deal with this problem, a new saving value selection strategy that can guide computers to high-quality solutions is developed.

Different from the descending order of saving values in CWS, our evaluation strategy is illustrated in Figure 2, where a forestlike data structure is demonstrated. In the data structure, each point represents a saving and may have its child saving points, in which each customers pair can be connected without violating any problem constraints if the pair of customers of its father point has been connected. For example, point 1 is the father point of point 1.1 and point 1.1 is its child point. The relationship of the two points indicates that the two customers of point 1.1 are feasible to be connected if the two customers of point 1 have been connected. As illustrated in Figure 2, points in the same generation belong to the same layer, for example, points 1, 2, and 3 in layer 1.

From the illustration of Figure 2, the evaluation values of point 1 can be calculated based on the evaluation values of its children whose evaluation values can be further calculated based on their children. Therefore, the evaluation values of the first layer's points may be calculated layer by layer from the bottom layer to the top layer, which can be called a "roll-up evaluation strategy."

However, the deeper the layer that the calculation begins from is, the more complex the calculation procedure is. Even for a small-scale problem with only hundreds of savings, the number of points in the forest is extremely large and hence it is very time-consuming and impossible to calculate the evaluation values of the top-layer points from the bottom layer. Furthermore, we do not exactly know the depth of the forest. Therefore, calculating the evaluation values of the top-layer points from an intermediate layer, for example, layer 2 or 3, not the bottom layer, becomes a feasible choice.

The evaluation values of a top-layer point and an intermediate-layer point can be calculated by formula (5), where EV_i, SV_i, and ic represent the evaluation value, the saving value, and the children number of point i, respectively. And p in formula (5) represents the weight that the saving value accounts for the evaluation value. The evaluation value

```
Set savingList = ∅, Solution = ∅;
While at least one of customers has not been included in Solution, do:
    Calculate the savings for all pairs of customers that can be directly connected, insert
      all the savings into savingList;
    Fetch a saving, denoted as Sav, from savingList according to the roll-up evaluation
      strategy which is detailed in Section 3.3;
    Connect the two routes that are corresponding to the two customers in Sav, so that a
      new route can be obtained;
    Insert the new combined route into Solution and delete the route that is
      corresponding to the two customers in Sav from Solution if it has been included in Solution;
    Clear savingList;
Endwhile;
Return Solution.
```

Algorithm 2: The procedure to construct an initial solution.

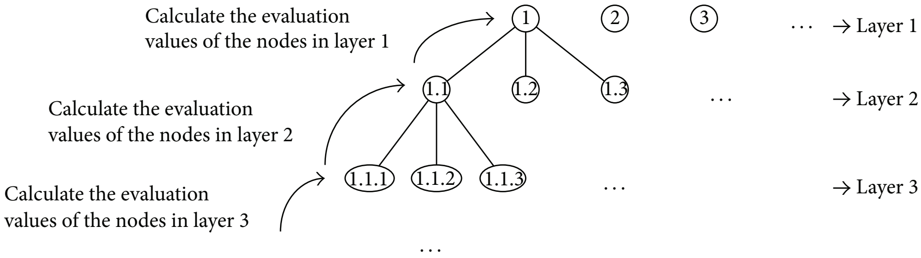

Calculate the evaluation values of the nodes in layer 1
1
2
3
· · · → Layer 1
Calculate the evaluation values of the nodes in layer 2
1.1
1.2
1.3
· · · → Layer 2
Calculate the evaluation values of the nodes in layer 3
1.1.1
1.1.2
1.1.3
· · · → Layer 3
· · ·

Figure 2: The illustration of the evaluation rule of savings.

of a point in the layer from which the calculation begins is equal to its saving value:

$$\mathrm{EV}_i = p \times \mathrm{SV}_i + \frac{(1-p)}{ic} \times \sum_{j=1}^{ic} \mathrm{EV}_j. \qquad (5)$$

The roll-up evaluation strategy uses a long-term perspective in that it can evaluate a pair of customers not only on its saving value but also on the saving values of its feasible subsequent connections if the pair of customers is connected, while, differently, the CWS only suggests selecting customers pairs in the descending order of their saving values, which may be regarded as a greedy strategy.

The procedure which adopts the recursion idea to calculate the evaluation values of a top-layer point is detailed in Algorithm 3.

3.4. The Solution Improvement Stage. The solution obtained from the solution construction stage can be improved because we have found from a large number of observations that the service start times of some customers in the obtained solution are easily delayed due to occasional travel time delays while others are relatively difficult. The reason that produces the result of the imbalance can be explained in the following. If avgLen_i is small but an edge (j, i) that has a large length is selected into the route of a solution, the reserved time for customer i is smaller than expected and customers like i are prone to be delayed. On the contrary, if avgLen_i is large but an edge (j, i) that has a small length is selected into the route of a solution, the reserved time for customer i is larger than expected and customers like i are not prone to be delayed. This imbalance can result in large penalty costs that are mainly incurred by the customers prone to be delayed. From a number of experiments, we find that the smaller the number of the customers that are prone to be delayed, the lower the penalty cost.

To reduce the number of the customers that are prone to be delayed, we develop a method to balance the possibilities of delaying service start time among multiple customers so that the penalty cost and the associated total cost of a solution could be reduced simultaneously. The key idea behind the method is to iterate the process of deleting from a route a customer node, which is prone to be delayed, inserting it into a relatively idle route, and evaluating the new solution by formula (4) until the total cost cannot be reduced. The method is described in Algorithm 4.

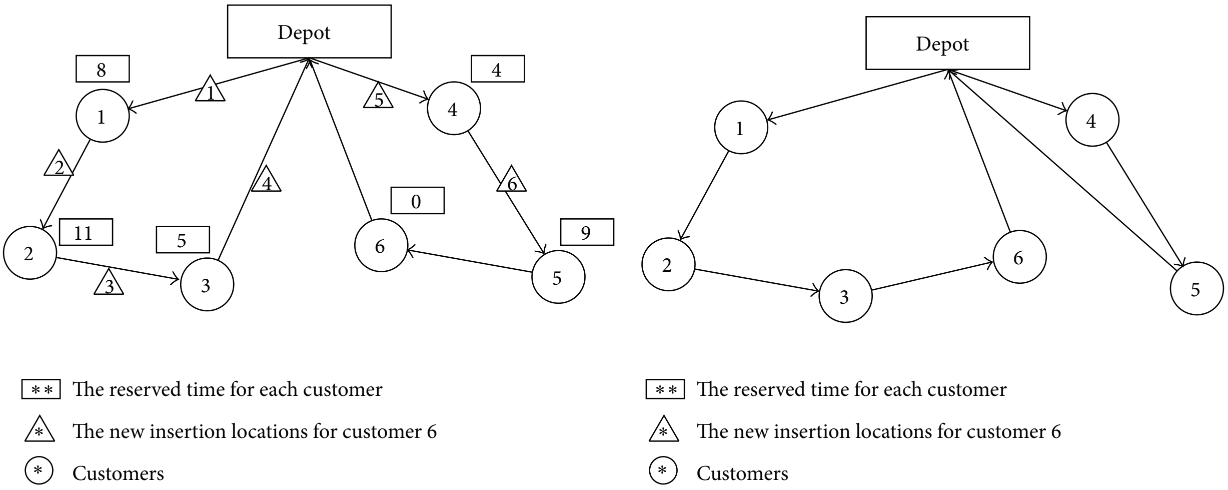
In Algorithm 4, we apply a procedure to iteratively delete a customer with the minimum reserved time and insert the customer into the other routes or the other locations of the same route in order to obtain a solution with less total cost. In each iteration, this procedure can be repeated. Figure 3 gives an example of one iteration, where there are two routes and each route has three customers. Customer 6 in Figure 3 has the minimum reserved time 0 and should be deleted for reinsertion. The reserved time 0 indicates that the service of customer 6 is very easy to be delayed once travel times are larger than expected during the process of delivery. There are 6 new insertion locations for customer 6 to be inserted, among which location 4 is the best one because the new solution obtained from inserting customer 4 into location 4

```
Function calculateEvaluationValue(Saving, Depth):
    Set $Depth = Depth - 1$;
    Connect the pair of customers in Saving;
    Set $TotalValue = 0$, $TotalValueNumber = 0$;
    While there is any saving in the saving list that has not been considered, do:
        Fetch a saving from saving list into Sav;
        If the pair of customers in Sav can be connected without violating any problem constraint, do:
            If $depth > 0$, do:
                Call function calculateEvaluationValue(Sav, depth) and get the evaluation value of Sav, denoted by SavEv;
                Set $TotalValue = TotalValue + SavEv$, $TotalValueNumber = TotalValueNumber + 1$;
            Elseif $depth = 0$, do:
                Set the evaluation value of Sav, denoted by SavEv, to its saving value;
                Set $TotalValue = TotalValue + SavEv$, $TotalValueNumber = TotalValueNumber + 1$;
            Endif;
    Endwhile;
    If $TotalValueNumber > 0$, set $SavingEv = TotalValue/TotalValueNumber$;
    Disconnect the pair of customers in Saving;
Return SavingEv.
```

Algorithm 3: The procedure to calculate the evaluation value of a top-layer point denoted by *saving* from the *depth* layer of the data structure.

```
Set the incumbent solution IS to the solution obtained from the solution construction stage;
Fabricate a solution $IS'$ whose expected total cost is set to an infinite value that cannot be
    exceeded by the total cost of any solution;
While the expected total cost of IS is less than that of $IS'$, do:
    Set $IS' = IS$, $Solutions = \emptyset$;
    Fetch the customer denoted by $j$ that has the minimum reserved time under the expected travel times;
    For each route ro, do:
        For each position ep of ro, into which a customer node can be inserted, do:
            Delete customer $j$ from its original route and insert it into ro at the position ep;
            Insert the solution obtained from the above step into Solutions;
        Next ep;
    Next ro;
    Fetch a solution with the least expected total cost $E(z)$ from Solutions into IS;
Endwhile;
Return the solution $IS'$.
```

Algorithm 4: The procedure of improving a solution generated by the solution construction stage.


Depot
8
1
1
5
4
4
2
1
4
2
11
5
0
6
6
9
3
2
3
5
** The reserved time for each customer
* The new insertion locations for customer 6
* Customers
Depot
1
4
2
6
3
5
** The reserved time for each customer
* The new insertion locations for customer 6
* Customers

(a) Initial solution

(b) New solution

Figure 3: An example of deleting a customer from a route and inserting them to another route.

has the minimum total cost. Therefore, customer 6 should be moved from their route to another and the new solution can be illustrated in Figure 3(b).

4. Computational Results

In this section, we present the computational results of the TSH, which has been coded in Java and run on a laptop with an Intel® Core™ 2 Duo CPU T7100 at 1.8 GHz and 2 GB RAM.

4.1. Test Instances and Parameter Setting. Because there are no standard benchmarks for SVRP, different sets of instances have been presented in the papers of the SVRP literature [3, 4, 6], which makes it difficult to compare the different methodologies that have been presented for SVRP. Therefore, we generalized the well-known Solomon instances and obtained 49 VRPTWSTT instances by replacing the fixed travel time of each pair of customers in VRPTW with a random travel time for VRPTWSTT. These generalized instances are solved by the method presented in this paper and compared with the best-known results of VRPTW. The details of Solomon instances and their best-known solutions can be found at the website http://neo.lcc.uma.es/vrp and the method used to randomly generate travel times is given in the following paragraphs.

Since the travel times in VRPTWSTT can follow any distribution and our heuristic does not have any limitation as long as its mean value is known, we choose the log-normal distribution to model the travel time for each edge as a result of its nonnegative values. The two parameters of the distribution, the location parameter μ_i and the scale parameter σ_i, are formulated by the following two expressions, respectively:

$$\mu_i = \ln\left(E\left[t_{ij}\right]\right) - \frac{1}{2}\ln\left(1 + \frac{\operatorname{Var}\left[t_{ij}\right]}{E\left[t_{ij}\right]^2}\right),$$
$$\sigma_i^2 = \ln\left(1 + \frac{\operatorname{Var}\left[t_{ij}\right]}{E\left[t_{ij}\right]^2}\right). \tag{6}$$

For each instance, we changed the travel time of an edge (i, j) $(v_i, v_j \in V,\ i \neq j)$ in VRPTW instances to a random value t_{ij} by setting $E[t_{ij}]$ to the Euclidean distance between the two nodes according to their coordinates. The original VRPTW instances can be particularly defined with $\operatorname{Var}[t_{ij}] = 0$. But for VRPTWSTT instances, we have the inequation $\operatorname{Var}[t_{ij}] > 0$. A larger variance of an edge indicates a higher uncertainty when traveling the edge. Our heuristic is tested under three different variance levels, the relatively low variances, the medium variances, and the relatively high variances. And these variance levels are further illustrated in Table 1, in which the relationship between the variance and the mean value for each variance level is given.

Table 1: The relationship between the variance and the mean value for each variance level.

Variance level	The relationship between the variance and the mean value
The relatively low variances	$\operatorname{Var}[t_{ij}] = 0.05E\left[t_{ij}\right]^2$
The medium variances	$\operatorname{Var}[t_{ij}] = 0.5E\left[t_{ij}\right]^2$
The relatively high variances	$\operatorname{Var}[t_{ij}] = 1.5E\left[t_{ij}\right]^2$

Note that there are 56 Solomon VRPTW instances, among which only 49 ones are selected. Some instances are not selected because penalty costs of solutions to these instances are almost close to 0. We test the penalty costs of all the best-known VRPTW solutions under the medium variance level and filter out 7 instances (C109, C201, C202, C203, C204, C205, and C208) by using the condition penalty cost/travel cost $<$ 0.005, which indicates time windows of customers in these instances are wide enough to tolerate travel time delays and hence the time window limitations of these instances are greatly weakened.

In order to decide the parameter values of our heuristic, preliminary experiments were performed on some instances with different parameter configurations to determine which configuration would be effective in solving the VRPTWSTT. The preliminary experiments showed a satisfactory performance when the parameter values were set to $r_0 = 0$, $r_{\text{step}} = 0.05$, $r_{\max} = 0.5$, $p = 0.5$, $c = 10$, and depth $= 3$, where depth represents the number of the layer from which the calculation of the evaluation of savings begins.

4.2. Detailed Results for One Instance. In this subsection, we present the numerical results for only one of the 49 instances in order to better illustrate our heuristic. Take the generalized version of the classical RC101 instance under the medium variance level $(\operatorname{Var}[t_{ij}] = 0.5E[t_{ij}]^2)$ for an example, and we run our heuristic 11 $((0.5 - 0)/0.05 + 1 = 11)$ times in which different runs have different r_0 and $r_{\max}$ values. But in one run we set $r_{\max} = r_0$ in order to make the heuristic generate a solution for only one percentage of the reserved time. Table 2 shows the alternative solutions (for the generalized instance with medium variances) that we obtained for different values of r_0 and $r_{\max}$. For each of the obtained solutions, the following data is provided: travel costs, penalty costs, total costs, and reliabilities.

In order to better demonstrate the character of alternative solutions to the generalized RC101 instance, a graphical representation is given in Figure 4 to illustrate the changing trends of travel costs, penalty costs, total costs, and reliabilities for all values of r. Definitions and solving procedures of these four indexes are given in Sections 2 and 3, respectively. From Figure 4, the following five points can be observed and discussed.

(1) The index of travel costs shows a continuously growing trend. This is mainly because, with the growth of the r value, time windows of customers are continuously tightened in order to reserve more time for vehicles to cope with unexpected travel time delays, but generally there may be some customers whose time window constraints are violated after it has been tightened. Therefore, a vehicle cannot complete the tasks that have been assigned when the time windows

Table 2: Results of alternative solutions to the RC101 instance with medium variances.

r_0 and $r_{\max}$	Travel cost	Penalty cost	Total cost	Reliability
0	1770.00	104.57	2815.69	0.94
0.05	1776.48	103.49	2811.36	0.94
0.1	1789.21	90.59	2695.07	0.96
0.15	1806.70	78.37	2590.42	0.97
0.2	1822.58	72.36	2546.20	0.97
0.25	1869.71	72.09	2590.58	0.97
0.3	1895.62	64.68	2542.45	0.98
0.35	1947.98	62.44	2572.35	0.98
0.4	1952.90	56.83	2521.21	0.98
0.45	**2007.21**	**48.12**	**2488.41**	**0.99**
0.5	2126.60	48.08	2614.35	0.99

are not tightened, and hence some good routes with more tasks have to be modified.

(2) On the contrary to the trend of travel costs, the index of penalty costs shows a continuously decreasing trend. This also conforms to the common sense that the more the time reserved is, the less likely it is to violate time window limitations.

(3) As the weighted sum of travel costs and penalty costs, the trend of total costs is not particularly obvious. The trend of the total costs can be divided into three stages: when r value is less than 0.2, the total costs demonstrate a decreasing trend, which indicates the penalty costs play a greater role in the contribution to total costs; when r value is between 0.25 and 0.45, the total costs fluctuate; when r value is greater than 0.45, the total costs grow as a result of the rapid growing trend of travel costs and a relatively stable reduction in penalty costs in this range of r values. Note that when r is 0.45, neither the travel cost, 2007.21, nor the penalty cost, 48.12, is the minimum, but their weighted sum, the total cost, reaches the minimum value of 2488.41.

(4) The two indices of both penalty costs and reliabilities can illustrate the level of a solution's probability of suffering deficient routes. As is expected, the index of reliabilities shows a stable growing trend, which is just the opposite to the overall trend of penalty costs. This is because the index of reliabilities denotes the percentage of the total number of poorly serviced customers while the penalty cost is the sum of penalty costs of these customers. As the r value grows, the reserved time and the reliabilities increase simultaneously. When the reliability reaches its maximum value (0.99), there are two solutions of different r values (0.45 and 0.5). Relative to the penalty cost index, a solution's reliability provides decision-makers with another index to evaluate the reliability of a solution. If a decision-maker cares about the cost of penalties, he can consider the index of penalty cost; and otherwise, if a decision-maker wants to minimize the number of poorly serviced customers, the index of reliabilities may come in handy.

(5) As for the running time of our heuristic, the process of generating the results for one alternative solution to the generalized RC101 instance can be completed in just fourteen seconds (average 14359 milliseconds on our laptop whose configuration is relatively low). Therefore, it seems completely feasible to implement the process for most real cases.

4.3. Results for 49 Instances.

4.3. Results for 49 Instances. The average results obtained for all instance sets (c1, c2, r1, r2, rc1, and rc2) under three variance levels are given in Table 3. The meaning of each column is as follows:

(1) BKS expected total cost: the expected total cost of best-known solution (BKS) obtained from the website http://neo.lcc.uma.es/vrp/, calculated according to formula (4).

(2) TSH expected total cost: the expected total cost of the solution generated by the TSH formulated by (4).

(3) Gap: the gaps in expected total costs between BKS and TSH, calculated as ((BKS−TSH)/TSH)×100%, where BKS and TSH stand for the values of their expected total costs.

The results of each generalized instance under any of the three uncertainty levels have similar statistical characteristics to those of the generalized RC101 instance discussed in Section 4.2.

From Table 3, the following four points can be observed and discussed.

(1) In a low-variability level defined by $\mathrm{Var}[t_{ij}] = 0.05E[t_{ij}]^2$, results show that the expected total costs of TSH tend to provide a better average reliability level of 0.9929 when applied to the VRPTWSTT. As is discussed before, higher reliability levels imply fewer numbers of poorly serviced customers and thus less expected penalty costs. The average gap is 2.07% between expected total costs of BKS and TSH. The expected total costs of BKS are the costs when travel times are deterministic instead of stochastic. From the table, the average r value is 0.292; that is, averagely, the latest service start time of each customer can be advanced by 29.2% of its average routing time from all the other nodes to it to attend unexpected congestion.

(2) As for the case of the medium-variability level with $\mathrm{Var}[t_{ij}] = 0.5E[t_{ij}]^2$, the average reliability level of the BKS is about 0.9923, which is lower than that of the low-variability level, indicating a higher degree of uncertainty. And the average gap in expected total costs between BKS and TSH is about 63.04%, which is much larger than 2.07% of the low-variance level. Important information about all positive gaps is also given, implying an absolute advantage for TSH over BKS under this level.

Table 3: Average results for each instance set under three variance levels.

Instance set	Results of best-known VRPTW solutions		Results of TSH			Gap	Variance level
	BKS expected total cost	Reliability	TSH expected total cost	Reliability	r		
c1	831.83	0.999	830.85	0.999	0.369	0.12%	The relatively low variances
c2	601.37	0.997	603.34	1.000	0.025	−0.32%	
r1	1363.81	0.962	1351.26	0.985	0.383	0.90%	
r2	1072.93	0.983	1028.65	0.998	0.323	4.14%	
rc1	1651.61	0.952	1601.59	0.978	0.338	3.19%	
rc2	1269.20	0.980	1215.64	0.998	0.313	4.37%	
Avg.	1131.79	0.979	1105.22	0.9929	0.292	2.07%	
c1	1332.58	0.980	1062.00	0.993	0.419	24.17%	The medium variances
c2	653.11	0.996	630.26	1.000	0.300	3.62%	
r1	3009.56	0.892	1640.47	0.985	0.413	81.73%	
r2	2083.48	0.954	1138.65	0.998	0.336	80.16%	
rc1	3959.32	0.881	2016.18	0.980	0.375	91.06%	
rc2	2630.62	0.945	1312.55	0.998	0.344	97.50%	
Avg.	2278.11	0.941	1300.02	0.9923	0.364	63.04%	
c1	2580.59	0.961	1091.76	0.993	0.425	132.27%	The relatively high variances
c2	757.45	0.995	660.64	1.000	0.450	14.89%	
r1	5630.55	0.863	1656.88	0.986	0.421	238.48%	
r2	4493.30	0.930	1156.05	0.998	0.327	280.99%	
rc1	7279.30	0.854	2108.23	0.977	0.338	238.74%	
rc2	5920.61	0.916	1356.47	0.998	0.363	319.61%	
Avg.	4443.63	0.920	1338.34	0.9920	0.387	204.16%	

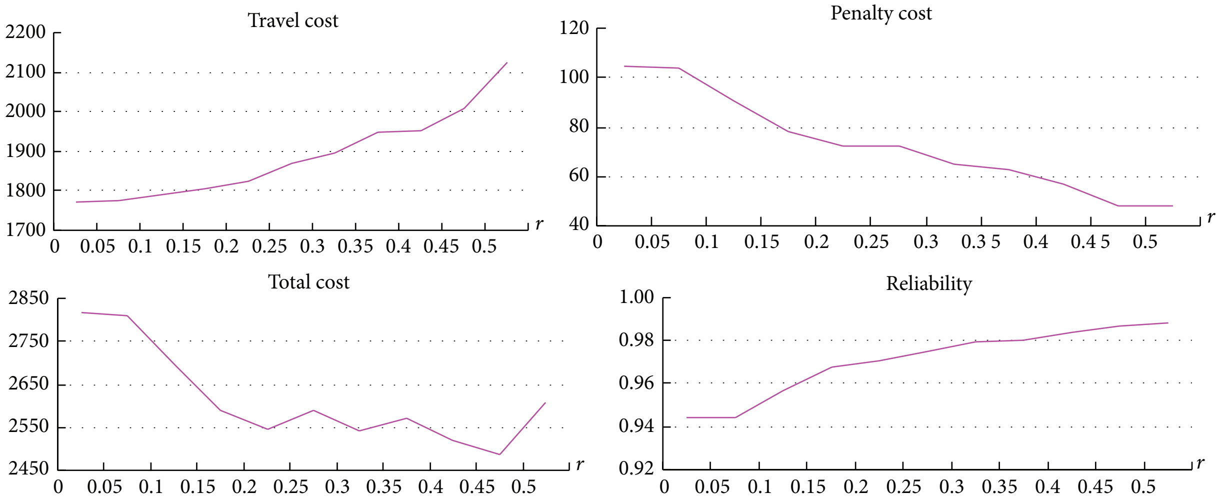


Figure 4: The changing trends of travel costs, penalty costs, total costs, and reliabilities for all values of r.

(3) The average reliability is only 0.920 for BKS under the high-variability level and a big average gap of 204.16% exists in expected total costs between BKS and TSH. The average reliability index is 0.9920 for TSH. Also, the average r value is 0.387, which is larger than the previous levels.

(4) From the table, we can also observe that all reliabilities of TSH are higher than those of BKS. A histogram is given in Figure 5 to graphically demonstrate and compare the average reliabilities of BKS and TSH solutions under three uncertainty levels.

5. Conclusions

In this paper, we present a three-stage heuristic for the vehicle routing problem with time window and stochastic travel time (VRPTWSTT). The heuristic, which does not require

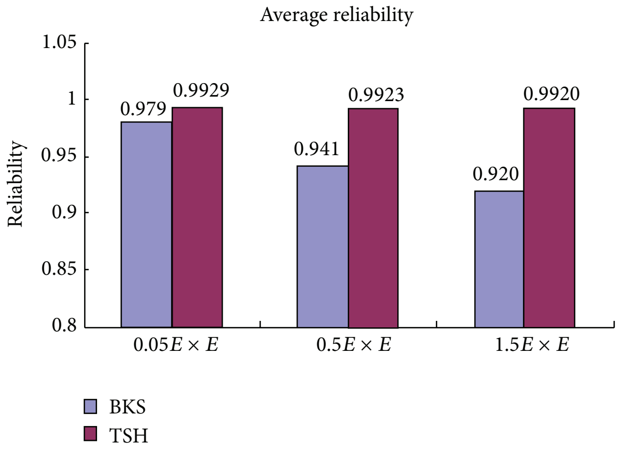


FIGURE 5: Average reliabilities of solutions under different uncertainty levels.

complex fine-tuning processes, solves VRPTWSTT by transforming it into multiple VRPTW, solving the VRPTW, and selecting the best one from the solutions of the VRPTW. The main contributions of this paper include the following.

(1) The heuristic reduces a complex VRPTWSTT, where no efficient heuristics have been developed yet, to a set of VRPTW, which can be solved by many excellent heuristics, by advancing the latest service start time of each customer ahead of its schedule. The time reserved for a customer is used for vehicles to cope with any unexpected travel time delay during their real delivery process. This idea can be applied to any variant of vehicle routing problems with stochastic travel times after it is appropriately adjusted.

(2) The heuristic is designed as a three-stage procedure to enhance the probability of generating high-quality solutions. The three stages are devoted to transforming the problem, constructing its initial solution, and improving the solution, respectively, so that a good balance between the travel cost and the penalty cost of a solution can be achieved and the total cost of a solution can be minimized. A number of conducted experiments show that the results produced by the heuristic for the generalized instances are better than the best-known solutions of the corresponding VRPTW in which the travel time between any pair of nodes is regarded as its expected value.

(3) The heuristic is flexible in that it can apply to a VRPTWSTT with any distribution of travel times with a known mean and the travel times of different edges can have different distributions and even possible dependence. Moreover, our heuristic can provide decision-makers with various solutions to practical problems, each of which considers a different period of time by which the latest service start time of each customer is advanced, indicating different levels of risk or reliability.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work is partially supported by the grants from the National Natural Science Foundation of China (nos. 71271037, 71201055, 71201014, and 71272093) and the Fundamental Research Funds for the Central Universities (no. DUT15QY38). Those supports are gratefully acknowledged.

References

[1] O. Bräysy and M. Gendreau, "Vehicle routing problem with time windows, part I: route construction and local search algorithms," *Transportation Science*, vol. 39, no. 1, pp. 104–118, 2005.

[2] M. A. Figliozzi, "The time dependent vehicle routing problem with time windows: benchmark problems, an efficient solution algorithm, and solution characteristics," *Transportation Research Part E: Logistics and Transportation Review*, vol. 48, no. 3, pp. 616–636, 2012.

[3] T. Zhang, W. A. Chaovalitwongse, and Y. Zhang, "Scatter search for the stochastic travel-time vehicle routing problem with simultaneous pick-ups and deliveries," *Computers & Operations Research*, vol. 39, no. 10, pp. 2277–2290, 2012.

[4] G. Laporte, F. Louveaux, and H. Mercure, "The vehicle routing problem with stochastic travel times," *Transportation Science*, vol. 26, no. 3, pp. 161–170, 1992.

[5] Y.-B. Park and S.-H. Song, "Vehicle scheduling problems with time-varying speed," *Computers & Industrial Engineering*, vol. 33, no. 3-4, pp. 853–856, 1997.

[6] A. S. Kenyon and D. P. Morton, "Stochastic vehicle routing with random travel times," *Transportation Science*, vol. 37, no. 1, pp. 69–82, 2003.

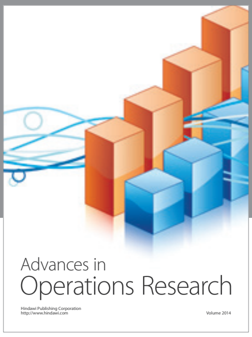
[7] C. Lecluyse, T. Van Wosensel, and H. Peremans, "Vehicle routing with stochastic time-dependent travel times," *4OR: A Quarterly Journal of Operations Research*, vol. 7, no. 4, article 363, 2009.

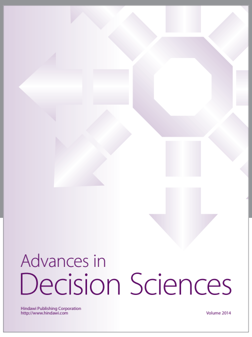
[8] R. D. Connors and A. Sumalee, "A network equilibrium model with travellers' perception of stochastic travel times," *Transportation Research Part B: Methodological*, vol. 43, no. 6, pp. 614–624, 2009.

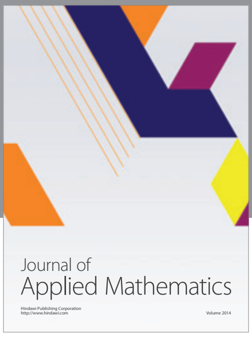
[9] A. Chen and Z. Zhou, "The α-reliable mean-excess traffic equilibrium model with stochastic travel times," *Transportation Research Part B: Methodological*, vol. 44, no. 4, pp. 493–513, 2010.

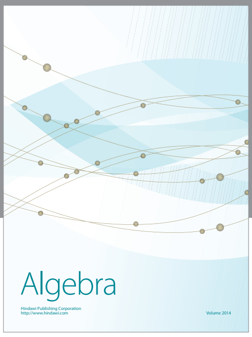
[10] Z. Wang and L. Lin, "A simulation-based algorithm for the capacitated vehicle routing problem with stochastic travel times," *Journal of Applied Mathematics*, vol. 2013, Article ID 127156, 10 pages, 2013.

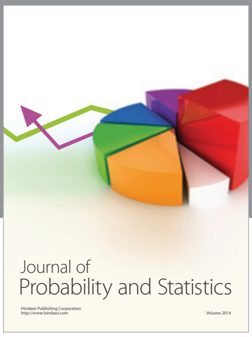
[11] J. K. Lenstra and A. H. G. Rinnooy Kan, "Complexity of vehicle routing and scheduling problems," *Networks*, vol. 11, no. 2, pp. 221–227, 1981.


Advances in
Operations Research

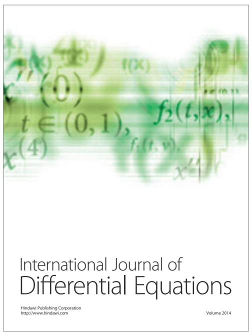

Advances in
Decision Sciences


Journal of
Applied Mathematics

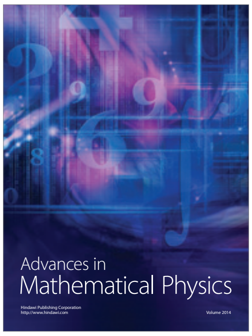

Algebra

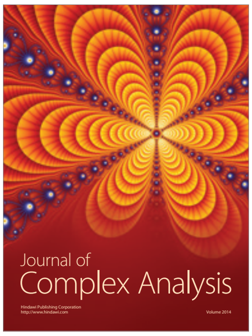

Journal of
Probability and Statistics

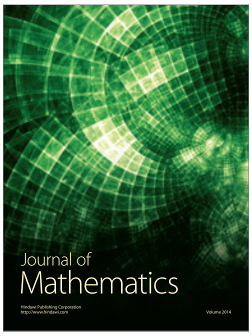
The Scientific
World Journal

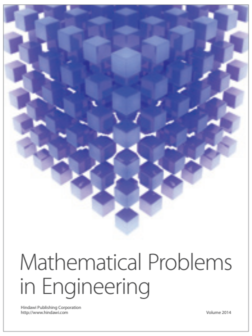

International Journal of
Differential Equations

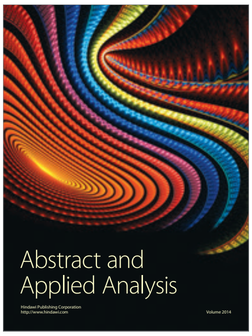
International Journal of
Combinatorics

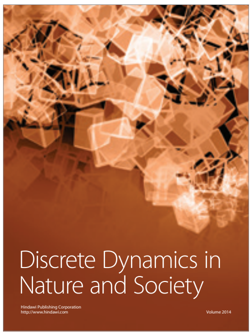

Advances in
Mathematical Physics

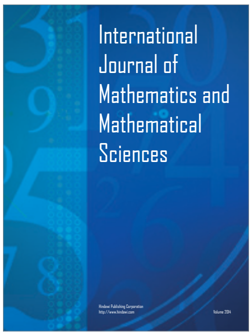

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

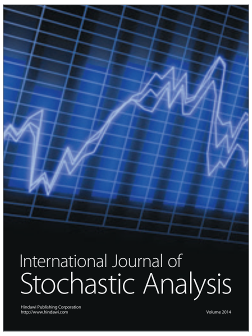

Abstract and
Applied Analysis


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