

## Research Article

# The Global Optimal Algorithm of Reliable Path Finding Problem Based on Backtracking Method

**Liang Shen, Hu Shao, Long Zhang, and Jian Zhao**

*School of Mathematics, China University of Mining and Technology, Xuzhou, Jiangsu 221116, China*

Correspondence should be addressed to Hu Shao; [shaohu@cumt.edu.cn](mailto:shaohu@cumt.edu.cn)

Received 18 April 2017; Accepted 29 August 2017; Published 18 October 2017

Academic Editor: Alessandro Arsie

Copyright © 2017 Liang Shen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

There is a growing interest in finding a global optimal path in transportation networks particularly when the network suffers from unexpected disturbance. This paper studies the problem of finding a global optimal path to guarantee a given probability of arriving on time in a network with uncertainty, in which the travel time is stochastic instead of deterministic. Traditional path finding methods based on least expected travel time cannot capture the network user's risk-taking behaviors in path finding. To overcome such limitation, the reliable path finding algorithms have been proposed but the convergence of global optimum is seldom addressed in the literature. This paper integrates the  $K$ -shortest path algorithm into Backtracking method to propose a new path finding algorithm under uncertainty. The global optimum of the proposed method can be guaranteed. Numerical examples are conducted to demonstrate the correctness and efficiency of the proposed algorithm.

## 1. Introduction

Finding a global optimal path in networks is a practical problem among communication and transportation. It has attracted much attention from researchers in numerous fields owing to its broad applications. The traditional path finding algorithms usually used the shortest average travel time as selection criteria. But in real networks, traffic condition is affected by many uncertain factors and the travel time is stochastic. Under such circumstances, the network users may face the risk of being late. For example, when scheduling significant issues (e.g., taking a flight), a network user needs to budget an extra travel time to guarantee a higher reliability of arriving on time. Then, it is very important for researchers to take travel time reliability into consideration for path finding. Thus, the travel time reliability-based path finding problem has attracted much attention by some recently proposed algorithms. However, few of them can guarantee the global optimum of the solution. In view of this, the problem of finding a global optimal path based on time reliability is studied in this paper.

## 2. Literature Review

The problem of finding optimal path in a transportation network has been investigated for a long time in the field of algorithm design and operation research. Most of these path finding algorithms were usually investigated under an ideal traffic condition; that is, the travel times are deterministic. For example, Andreatta and Romeo [1] considered Stochastic Shortest Path (SSP) problems in probabilistic networks. Polychronopoulos and Tsitsiklis [2] considered shortest path problems defined on graphs with random arc costs. And it assumed that information on arc cost values is accumulated as the graph is being traversed. Miller-Hooks and Mahmassani [3] addressed the problem of determining least expected time paths in stochastic, time-varying networks. Gao and Chabini [4] studied optimal routing policy problems in stochastic time-dependent networks, where link travel times are modeled as random variables with time-dependent distributions. Minh and Pumwa [5] addressed the problem of finding feasible path planning algorithms for nonholonomic vehicles. Most of all these existing path finding algorithms have been

investigated under various conditions, but the reliability of on-time arrival or the risk of being late in path finding problems has not received much attention.

The unreliability (or uncertainty) issue has been considered in some recently developed algorithms (Andreatta and Romeo [1], Polychronopoulos and Tsitsiklis [2], Fu and Rilett [6], and Zhang et al. [7]). When scheduling important issues (e.g., attending a conference), a network user considers a shorter travel time, as well as a higher reliability of on-time arrival. Based on the above discussion, the definition of travel time reliability is defined as the probability that a network user can arrive at the destination within a given travel time threshold. To account for this issue, Frank [8] developed an efficient method of hypothesis testing to solve the problem of finding shortest path probability distributions in graphs. Shao et al. [9] proposed a heuristic algorithm to search for the highest reliability path. Chen and Ji [10] developed a simulation-based genetic algorithm procedure to solve these path finding models under uncertainties. Ying [11] presented a reliable routing algorithm with real time and historical information. Nie and Wu [12] studied the problem of finding a prior shortest path to guarantee a given likelihood of arriving on time in a stochastic network. Chen et al. [13] proposed an efficient algorithm for finding a risk-averse path for users in autonomous vehicle navigation systems. By making use of information computed at the start of the trip, the authors presented a responsive version of the constrained  $A^*$  search, which reduced the on-demand response time. Khani and Boyles [14] studied the reliable path problem in the form of minimizing the sum of mean and standard deviation of path travel time. Chen et al. [15] investigated the problem of finding the  $K$ -reliable shortest paths (KRSP) in stochastic networks under travel time uncertainty. In such study, a deviation path algorithm was proposed to exactly solve the KRSP problem in large-scale networks. However, all these studies cannot guarantee the global optimum of the solution for path finding.

The main difficulty of finding the optimum of the reliable path is due to the nonadditive property of the travel time reliability. The path travel time reliability cannot be expressed as the sum of some terms corresponding to the links. Thus, the reliable path finding problem has been regarded as one of the nonadditive path cost problems. There is no simple way of converting the nonadditive route cost to equivalent link costs (Chen et al. [16]). Therefore, to search the reliable path needs to enumerate all the alternative paths between each O-D pair. This is of great difficulty for large-scale transportation networks. In this paper, the  $K$ -shortest finding algorithm is adopted so as to find the global optimum of the reliable path.

Some methods have been proposed to solve the  $K$ -shortest finding problem during the past two decades. Eppstein [17] gave algorithms for finding the  $K$ -shortest paths connecting a pair of vertices in a digraph, and cycles of repeated vertices are allowed. Hershberger et al. [18] described a new algorithm to enumerate the  $K$ -shortest simple (loopless) paths in a directed graph and reported on its implementation. It was based on a replacement paths algorithm. Carlyle and Wood [19] proposed a new algorithm

for solving the problem of enumerating all near-shortest simple (loopless) paths in a graph with nonnegative edge lengths. Liu and Ramakrishnan [20] presented a new algorithm,  $A^*$  Prune, to list the first  $K$  Multi-Constrained-Shortest Path (KMCSPP) between a given pair of nodes in a digraph. Although there exist a lot of  $K$ -shortest algorithms, few of them have been used for solving the reliability-based path finding problem.

In this paper, the  $K$ -shortest algorithm is integrated into the Backtracking algorithm for proposing a new reliability-based path finding algorithm. The Backtracking algorithm is widely used for finding solutions in some problem with high computational cost. It can find the optimum of the concerned problem. The efficiency of the Backtracking algorithm is based on the pruning function. An efficient pruning function is capable of saving the computational time for the implementation of Backtracking algorithm. In this paper, the results of the  $K$ -shortest algorithm are used as the threshold for the pruning function to improve the computational efficiency of the Backtracking algorithm. The main contributions of the proposed algorithm are given as follows.

(1) The new reliability-based path finding algorithm under uncertainty is presented based on the Backtracking algorithm. In this new algorithm, the global optimum solution of the proposed problems can be guaranteed.

(2) An efficient pruning function is capable of saving the computational time. Moreover, the  $K$ -shortest path finding algorithm is employed to further reduce the computational time of the proposed algorithm.

In the next section, the definition of reliable path will be introduced first, and then the detailed steps of the algorithm are presented. In Section 4, numerical examples are carried out to demonstrate the applications of the proposed method. Finally, Section 5 concludes the paper with possible directions for future research.

### 3. Global Optimal Path Finding Algorithm under Uncertainty

The optimal path is defined to find the path with minimum travel time budget (or effective travel time) in this paper (Chen et al. [21–25]). As aforementioned, it is difficult to find the path with minimum effective travel time unless enumerating all the paths. In order to avoid path enumeration while finding the reliable path, Wang et al. [26] proposed a heuristic solution algorithm using the  $K$ -shortest paths algorithm. However, the proposed algorithm in Wang et al. [26] cannot find out the global optimal solution. Thus, the Backtracking method is adopted in order to find the reliability-based global optimal path in this section.

To facilitate the presentation of the essential ideas without loss of generality, the following basic assumptions are made in this paper.

(A1) Due to network uncertainties, the travel time on each link is assumed to follow the normal distribution. The travel time on path between each OD pair is the summation of the link travel times on that path. For simplicity, it is assumed that the path travel time also follows the Normal distribution (Chang et al. [27]).

(A2) The link travel times are nonnegative and all link travel times are statistically independent.

(A3) Following the assumptions in Chen et al. [21–25], the reliable path is defined as the path with minimum effective travel time in this paper.

(A4) The travel time budget (or effective travel time) is defined as the sum of mean travel time and safe margin.

**3.1. The Definition of Reliable Path.** According to assumption (A1), it is assumed that expected travel time (or mean travel time) on path  $k$  between OD pair  $w$  is expressed as follows:

$$t_k^w = \sum_{a \in A} \delta_{k,a}^w t_a, \quad \forall k \in K^w, \quad w \in R^{\text{OD}}, \quad (1)$$

where  $K^w$  is the set of all paths between OD pair  $w$  and OD is the set of all OD pairs. It follows from assumption (A2) that the travel time variance on path  $k$  between OD pair  $w$  can be written as

$$(\sigma_k^w)^2 = \sum_{a \in A} \delta_{k,a}^w (\sigma_a)^2, \quad \forall k \in K^w, \quad w \in R^{\text{OD}}. \quad (2)$$

Following assumption (A1), the travel time on path  $k$  between OD pair  $w$  follows Normal distribution as follows:

$$T_k^w \sim N(t_k^w, \sigma_k^w), \quad \forall k \in K^w, \quad w \in R^{\text{OD}}, \quad (3)$$

where  $t_a$  and  $\sigma_a$  are the mean and standard deviation of the travel time on link  $a$ , respectively;  $\delta_{k,a}^w$  is the path-link incidence parameter; and  $\delta_{k,a}^w = 1$  if path  $k$  uses link  $a$  and  $\delta_{k,a}^w = 0$  otherwise.

According to Lo et al. [28], the travel time budget (or effective travel time) associated with path  $k$  can be written as

$$\hat{t}_k^w = t_k^w + \lambda \sigma_k^w, \quad \forall k \in K^w, \quad w \in R^{\text{OD}}, \quad (4)$$

where  $\lambda$  is a parameter related to the risk-taking behaviors of the travelers. According to assumption (A4), in order to avoid late arrival penalty, an additional travel time budget named safety margin, that is,  $\lambda \sigma_k^w$ , is proposed to ensure a more certain on-time arrival probability by the travelers. Mathematically, the additional travel time budget  $\lambda \sigma_k^w$  can be determined by the following chance constraint model:

$$\begin{aligned} \min_{\lambda} \quad & \hat{t}_k^w = t_k^w + \lambda \sigma_k^w \\ \text{s.t.} \quad & \Pr[T_k^w \leq \hat{t}_k^w] \geq \rho, \end{aligned} \quad (5)$$

where  $\rho$  is a given confidence level. If the network user wants the probability of arriving at the destination to be not less than 95%, then set the probability  $\rho = 95\%$ . By a simple manipulation,  $\lambda$  can be determined as the total travel time budget in minimization (5); then the parameter can be expressed as follows:

$$\lambda = \Phi^{-1}(\rho), \quad (6)$$

where  $\Phi(\cdot)$  is the cumulative distribution function (CDF) of standard normal distribution.

The definition of travel time budget ensures the on-time arrival reliability to be greater than or equal to  $\rho$ , while minimizing the travel time budget. Then, according to assumption (A3), the reliable path can be defined as follows: among all the paths between OD pair  $w$ , the reliable path between OD pair  $w$  is the path with minimum travel time budget. Mathematically, path  $k^*$  is the reliable path between OD pair  $w$  if and only if

$$k_{w,(p)}^* = \arg \min_k \hat{t}_k^w. \quad (7)$$

Then, the candidate path set  $M^{(p)}$  consists of the two path sets; that is,

$$M^{w,(p)} = M_{mt}^{w,(p)} \cup M_{sm}^{w,(p)}, \quad (8)$$

where  $M_{mt}^{w,(p)}$  includes all  $p$ -shortest paths in terms of minimum mean travel time and  $M_{sm}^{w,(p)}$  includes all  $p$ -shortest paths in terms of minimum variance. Due to the additive property of the mean travel time and the variance, the conventional  $K$ -shortest methods can be utilized to generate  $M_{mt}^{w,(p)}$  and  $M_{sm}^{w,(p)}$  without generating all the paths between OD pair  $w$ . It is known that if  $p$  is large enough, then  $M^{w,(p)}$  covers all the paths between OD pair  $w$ . Then, the reliable path can be found by calculating the travel time budgets of all the paths in set  $M^{w,(p)}$ . Therefore, the reliable path can be obtained in set  $M^{w,(p)}$  and it can be expressed as follows:

$$k_{w,(p)}^* = \arg \min_{k \in M^{w,(p)}} \hat{t}_k^w. \quad (9)$$

**3.2. The Algorithm Description.** Backtracking method adapted herein is a widely used algorithm for integer programming. If an appropriate/effective pruning function is defined in Backtracking method, the algorithm may quickly converge to the global optimum. The pruning function is adopted to avoid searching the feasible solutions which are not optimal solutions so as to save the computational time. The advantage of this algorithm is that it can guarantee finding out the global optimal solution. The proposed heuristic algorithm is shown as follows.

Firstly, we need to define two kinds of data structures: Stack and List. The Stack has the property of LIFO (Last in First out) and it is used to store the nodes which have not been visited. The List is used to store the nodes which have been visited before.

*Step 1.* Put initial node  $s$  into Stack and carry out the local optimal searching algorithm within the set  $M^{(p)}$  using heuristic algorithm in Carlyle and Wood [19]. Then, the effective travel time of the local optimal reliable path is defined as UB (upper bound). The upper bound of the effective travel time is selected as initial pruning function. Denote the corresponding reliable path as  $P$ .

*Step 2.* If the Stack is empty, then stop. Otherwise, go to Step 3.

*Step 3.* Select the first node  $V_i$  from the Stack and move the node to the List which is connected to node  $V_i$ .

TABLE 1: Comparisons of the proposed method with enumeration method.

| OD pairs | The proposed method |                                 | Results of enumeration method                  |                                 |
|----------|---------------------|---------------------------------|--|---------------------------------|
|          | The path sequence   | The effective travel time (min) | The reliable path sequence between OD pair 1-9 | The effective travel time (min) |
| (1, 9)   | [1 4 5 6 9]         | 31.41                           | [1 2 5 6 9]                                    | 32.47                           |
| (1, 8)   | [1 4 5 8]           | 22.06                           | [1 4 5 8 9]                                    | 31.53                           |
| (1, 7)   | [1 4 7]             | 16.03                           | [1 4 5 6 9]                                    | 31.41                           |
| (1, 6)   | [1 4 5 6]           | 21.53                           | [1 4 7 8 9]                                    | 33.63                           |
| (1, 5)   | [1 4 5]             | 14.21                           | [1 2 5 8 9]                                    | 32.80                           |
| (2, 9)   | [2 5 6 9]           | 23.64                           | [1 2 3 6 9]                                    | 33.51                           |

*Step 4.* Denote the current path as  $P_i$ . If node  $V_i$  is the destination node, then calculate the effective travel time of path  $P_i$  and denote it as ET. If  $ET > UB$ , then go to Step 2. Otherwise, update  $UB = ET$  and  $P = P_i$ ; then go to Step 2. If node  $V_i$  is not the destination node, then go to Step 5.

*Step 5.* Extend the subsequent nodes of  $V_i$  and put all the subsequent nodes into Stack.

*Step 6.* Select the first node  $V_j$  from the Stack.

- (i) If  $V_j \in P_i$ , then move the node  $V_j$  from the Stack to the List and go to Step 6. Otherwise, calculate the effective travel time of the reliable path  $P_j = [P_i, V_j]$  and denote it as ET.
- (ii) If  $V_j$  is the destination node and  $ET > UB$ , then move  $V_j$  from the Stack to the List and go to Step 6. If  $ET \leq UB$ , update  $UB = ET$  and  $P = P_i$  and then move node  $V_j$  from the Stack to the List and go to Step 6.
- (iii) If  $V_j$  is not the destination node and  $ET > UB$ , then move node  $V_j$  from the Stack to the List and go to Step 6. If  $ET \leq UB$ , then move node  $V_j$  from the Stack to the List and go to Step 2.

Based on the algorithm steps, the flow chart of the algorithm is shown in Figure 1.

## 4. Numerical Examples

Numerical examples are carried out to demonstrate the applications of the proposed algorithm using a small-sized network and a medium-sized network.

*4.1. Validation of the Proposed Algorithm.* A small-sized network is adopted to verify the proposed algorithm. The network used in this example is shown in Figure 2. The mean and standard deviation of link travel time are shown near each link in this figure with the format “mean/standard deviation” in minute. In this example, the confidence level is assumed to  $\rho = 95\%$  and  $K = 3$ .

The results of the proposed algorithm are given in Table 1. It can be seen that the reliable path from node 1 to node 9 is [1 4 5 6 9] and the effective travel time is 31.41 minutes. The enumeration method is adopted to validate this solution

and the corresponding results of the proposed algorithm are also shown in Table 1. It is clear that the solution of the heuristic solution algorithm is the same as the enumeration result and this demonstrates that the proposed heuristic algorithm in this paper is correct.

*4.2. Comparisons with Existing Reliability-Based Path Finding Algorithms.* The network used in this example is shown in Figure 3. The mean and standard deviations of link travel time are shown in Table 2. In this example, the confidence level is assumed to  $\rho = 95\%$  and  $K = 3$ .

For comparison purpose, a heuristic solution algorithm for travel time reliability-based (Wang et al. [26]) has been employed. The results of such algorithm and the proposed algorithm are shown in Table 3. It can be seen that the proposed method outperformed the algorithm of Wang et al. [26]. For example, for OD pair (1, 24), the proposed method can find the global reliable path (with effective travel time of 45.3438 minutes) but the algorithm of Wang et al. [26] can only find the local reliable path (with effective travel time of 53.0493 minutes). It can be seen from this example that the proposed algorithm can find more reliable path than the algorithm of Wang et al. [26].

## 5. Conclusion and Future Studies

This paper proposed a global optimal algorithm for travel time reliability-based path finding problem using Backtracking method. The traditional path finding algorithms usually used the shortest average travel time as selection criteria. But traffic condition is realistically affected by many uncertain factors and the travel time is stochastic in reality. Under such circumstances, the travelers suffer from the risk of being late. In view of this, researchers take travel time reliability into consideration while making path choice decisions. Due to the nonadditive property of the travel time reliability, finding the reliability-based path needs to enumerate all the paths between the OD pair, which is very time consuming for large-scale network. To address this issue, a new path finding algorithm was proposed in this paper by integrating the  $K$ -shortest paths algorithm into Backtracking method. This proposed method can guarantee the global optimum of the solution. Numerical studies illustrated that the proposed

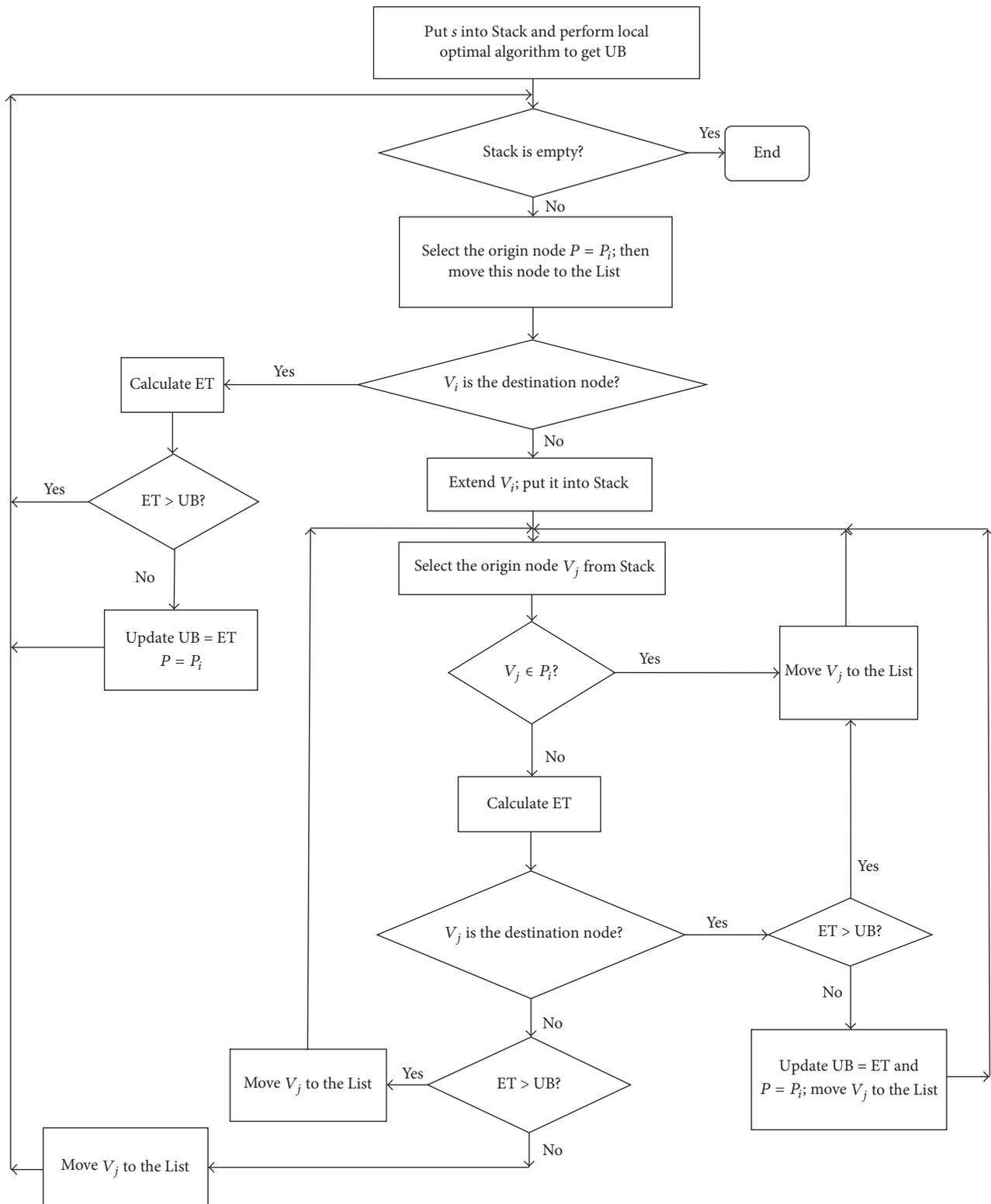


FIGURE 1: The flow chart of the global algorithm of reliable path finding.

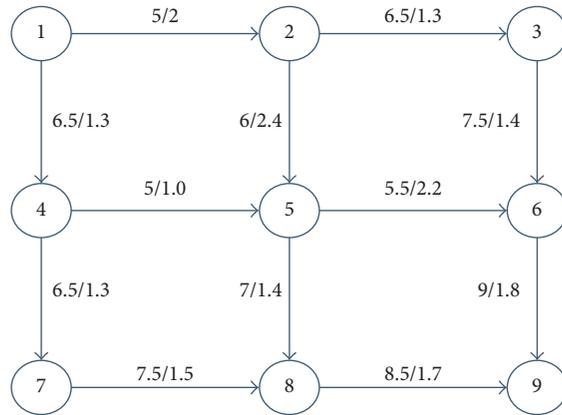


FIGURE 2: An example network for reliable path problem.

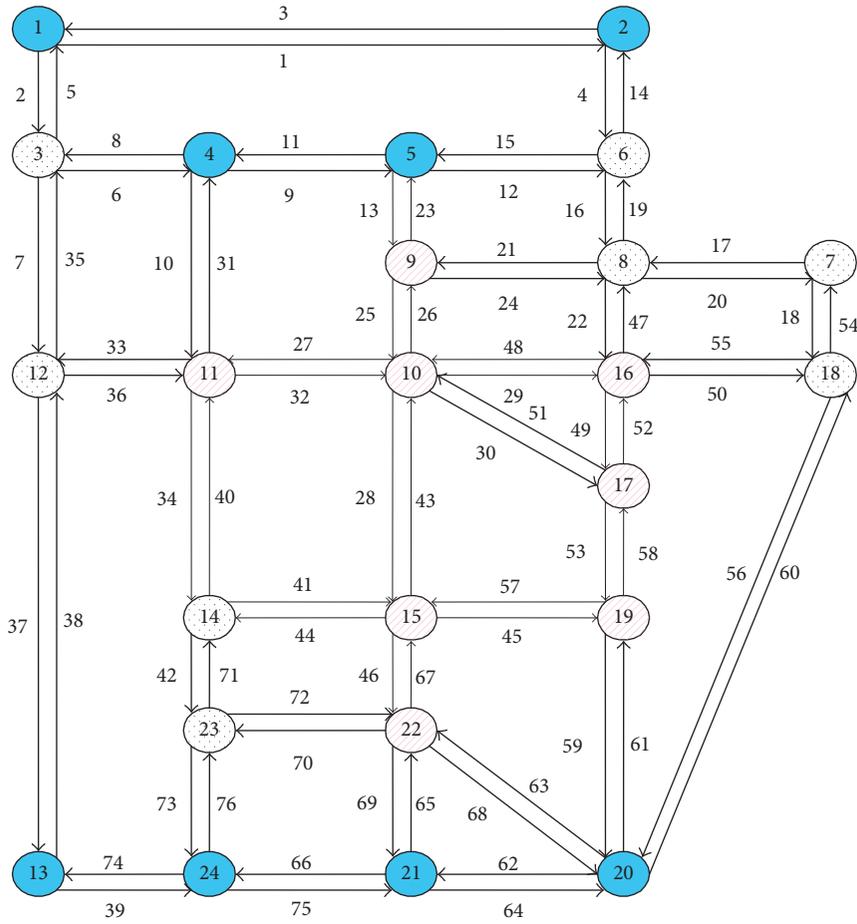


FIGURE 3: Sioux Falls network.

TABLE 2: The mean and standard deviations of each link travel time.

| Link number | Mean/variance |
|-------------|---------------|
| 1           | 3.6/3.7       |
| 2           | 2.4/11.7      |
| 3           | 3.6/17.6      |
| 4           | 3.0/3.7       |
| 5           | 2.4/12.2      |
| 6           | 2.4/9         |
| 7           | 2.4/12.2      |
| 8           | 2.4/4.2       |
| 9           | 1.2/7.7       |
| 10          | 3.6/9         |
| 11          | 1.2/11.7      |
| 12          | 2.4/4.2       |
| 13          | 3.0/8         |
| 14          | 3.0/7.7       |
| 15          | 2.4/8         |
| 16          | 1.2/9.5       |
| 17          | 1.8/4.2       |
| 18          | 1.2/9.5       |
| 19          | 1.2/7.4       |
| 20          | 1.8/4.3       |
| 21          | 2.0/4.2       |
| 22          | 3.0/3.1       |
| 23          | 3.0/2         |
| 24          | 2.0/7.4       |
| 25          | 1.8/9.4       |
| 26          | 1.8/16.3      |
| 27          | 3.0/3.1       |
| 28          | 3.6/9.4       |
| 29          | 3.0/20        |
| 30          | 4.2/10.7      |
| 31          | 3.6/9         |
| 32          | 2.4/4.3       |
| 33          | 3.6/12.3      |
| 34          | 2.4/13.8      |
| 35          | 2.4/9         |
| 36          | 3.6/12.2      |
| 37          | 1.8/13.8      |
| 38          | 1.8/17.6      |
| 39          | 2.4/3         |
| 40          | 2.4/3         |
| 41          | 3.0/13.7      |
| 42          | 2.4/4         |
| 43          | 3.6/13.8      |
| 44          | 3.0/13.7      |
| 45          | 2.4/12.2      |
| 46          | 2.4/7.2       |
| 47          | 3.0/16.3      |
| 48          | 3.0/20        |
| 49          | 1.2/13.8      |
| 50          | 1.8/12.2      |
| 51          | 4.2/5.7       |
| 52          | 1.2/5.7       |
| 53          | 1.2/15        |
| 54          | 1.2/9.6       |

TABLE 2: Continued.

| Link number | Mean/variance |
|-------------|---------------|
| 55          | 1.8/10.7      |
| 56          | 2.4/15        |
| 57          | 2.4/5.5       |
| 58          | 1.2/14.8      |
| 59          | 2.4/2         |
| 60          | 2.4/5.5       |
| 61          | 2.4/14.8      |
| 62          | 3.6/10        |
| 63          | 3.0/6.5       |
| 64          | 3.6/9.6       |
| 65          | 1.2/14.6      |
| 66          | 1.8/2.3       |
| 67          | 2.4/7.2       |
| 68          | 3.0/2.3       |
| 69          | 1.2/4.2       |
| 70          | 1.4/4         |
| 71          | 2.4/4.2       |
| 72          | 2.4/4         |
| 73          | 1.2/6.5       |
| 74          | 2.4/6         |
| 75          | 1.8/4         |
| 76          | 2.0/6         |

algorithm is capable of finding the globally reliable path and outperforming the existing algorithm.

On the basis of the solution algorithm proposed in this paper, some further extensions can be envisaged as follows:

- (i) Some new techniques, such as the multicriteria A\* algorithm (Chen et al. [21]), the multicriteria label-setting algorithm (Chen et al. [23]), and the Lagrangian substitution algorithm (Xing and Zhou [29]), could be employed in the proposed algorithm to provide a more efficient pruning function so as to further accelerate the convergence of the algorithm.
- (ii) How to incorporate the proposed algorithm into the reliability-based traffic assignment problem reveals significant investigations in further studies.
- (iii) Different travelers may have different attitudes toward risk of being late. Such kind of multiple risk-taking behaviors could be formulated as multiclass path choice criteria. How to modify the proposed algorithm to account for multiclass path choice criteria deserves extensions.
- (iv) The proposed algorithm is designed for static path finding problem and as such this method is different from the algorithm of the dynamic path finding problem. How to extend such method to the dynamic reliable path finding problems (e.g., Li et al. [30]) is worthy of further investigation.
- (v) The proposed algorithm can be combined with some stochastic travel time estimation methods (Tam and Lam [31, 32]; Chang et al. [33]; Shao et al. [34, 35]) so as to provide a reliable route guidance system in reality.

TABLE 3: Comparisons of the proposed method with existing method.

| OD pairs | The proposed method   |                                 | Heuristic method in Wang et al. (2010) |                                 |
|----------|-----------------------|---------------------------------|--|---------------------------------|
|          | The path sequence     | The effective travel time (min) | The reliable path sequence             | The effective travel time (min) |
| (1,24)   | [1 3 12 13 24]        | <b>45.3438</b>                  | [1 2 6 8 9 10 15 22 21 24]             | <b>53.0439</b>                  |
| (1,23)   | [1 3 12 13 24 23]     | 48.6681                         | [1 2 6 8 9 10 15 22 23]                | 51.1572                         |
| (1,20)   | [1 2 6 8 7 18 20]     | 48.2550                         | [1 2 6 8 7 18 20]                      | 48.2550                         |
| (1,15)   | [1 2 6 8 9 10 15]     | 44.3443                         | [1 2 6 8 9 10 15]                      | 44.3443                         |
| (2,23)   | [2 6 8 9 10 15 22 23] | 46.9724                         | [2 6 8 9 10 15 22 23]                  | 46.9724                         |
| (2,20)   | [2 6 8 7 18 20]       | 44.1193                         | [2 6 8 7 18 20]                        | 44.1193                         |
| (2,15)   | [2 6 8 9 10 15]       | 40.0977                         | [2 6 8 9 10 15]                        | 40.0977                         |
| (5,20)   | [5 9 10 15 22 20]     | 42.2762                         | [5 9 10 15 22 20]                      | 42.2762                         |
| (24,1)   | [24 23 14 11 4 3 1]   | 44.2569                         | [24 23 14 11 4 3 1]                    | 44.2569                         |
| (23,2)   | [23 14 11 4 5 6 2]    | 40.7664                         | [23 14 11 4 5 6 2]                     | 40.7664                         |

## Conflicts of Interest

The authors declare that there are no conflicts of interest of the received funding, mentioned in Acknowledgments, regarding the publication of this paper.

## Acknowledgments

The work described in this paper was jointly supported by grants from National Natural Science Foundation of China (Project nos. 71671184 and 71271205) and a discipline construction project from China University of Mining and Technology (Reconstruction of the Disciplinary Echelon for Teachers in Statistics, Project no. 04170116/10250301).

## References

- [1] G. Andreatta and L. Romeo, "Stochastic shortest paths with recourse," *Networks. An International Journal*, vol. 18, no. 3, pp. 193–204, 1988.
- [2] G. H. Polychronopoulos and J. N. Tsitsiklis, "Stochastic shortest path problems with recourse," *Networks*, vol. 27, no. 2, pp. 133–143, 1996.
- [3] E. D. Miller-Hooks and H. S. Mahmassani, "Least expected time paths in stochastic, time-varying transportation networks," *Transportation Science*, vol. 34, no. 2, pp. 198–215, 2000.
- [4] S. Gao and I. Chabini, "Optimal routing policy problems in stochastic time-dependent networks," *Transportation Research Part B: Methodological*, vol. 40, no. 2, pp. 93–122, 2006.
- [5] V. T. Minh and J. Pumwa, "Feasible path planning for autonomous vehicles," *Mathematical Problems in Engineering*, vol. 2014, Article ID 317494, 2014.
- [6] L. Fu and L. R. Rilett, "Expected shortest paths in dynamic and stochastic traffic networks," *Transportation Research Part B: Methodological*, vol. 32, no. 7, pp. 499–516, 1998.
- [7] X. Zhang, H. Li, J. Peng, and W. Liu, "A Cooperative Q-Learning Path Planning Algorithm for Origin-Destination Pairs in Urban Road Networks," *Mathematical Problems in Engineering*, vol. 2015, Article ID 146070, 2015.
- [8] H. Frank, "Shortest paths in probabilistic graphs," *Operations Research*, vol. 17, pp. 583–599, 1969.
- [9] H. Shao, W. H. K. Lam, and K. S. Chan, "The problem of searching the reliable path for transportation networks with uncertainty," in *Proceedings of the 9th Conference of Hong Kong Society for Transportation Studies*, pp. 226–234, China, 2004.
- [10] A. Chen and Z. Ji, "Path finding under uncertainty," *Journal of Advanced Transportation*, vol. 39, no. 1, pp. 19–37, 2005.
- [11] J. Q. Ying, "A reliable routing algorithm with real time and historical information," *Journal of Transportation Systems Engineering and Information Technology*, vol. 5, pp. 113–119, 2005.
- [12] Y. Nie and X. Wu, "Shortest path problem considering on-time arrival probability," *Transportation Research Part B: Methodological*, vol. 43, no. 6, pp. 597–613, 2009.
- [13] Y. Chen, M. G. H. Bell, and K. Bogenberger, "Risk-averse autonomous route guidance by a constrained A\* search," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 14, no. 3, pp. 188–196, 2010.
- [14] A. Khani and S. D. Boyles, "An exact algorithm for the mean-standard deviation shortest path problem," *Transportation Research Part B: Methodological*, vol. 81, part 1, pp. 252–266, 2015.
- [15] B. Y. Chen, Q. Li, and W. H. K. Lam, "Finding the  $k$  reliable shortest paths under travel time uncertainty," *Transportation Research Part B: Methodological*, vol. 94, pp. 189–203, 2016.
- [16] A. Chen, Z. Ji, and W. Recker, "Travel time reliability with risk-sensitive travelers," *Transportation Research Record*, vol. 1783, pp. 27–33, 2002.
- [17] D. Eppstein, "Finding the  $k$  shortest paths," *SIAM Journal on Computing*, vol. 28, no. 2, pp. 652–673, 1999.
- [18] J. Hershberger, M. Maxel, and S. Suri, "Finding the  $k$  shortest simple paths: a new algorithm and its implementation," *ACM Transactions on Algorithms (TALG)*, vol. 3, no. 4, Article ID 1290682, Art. 45, 19 pages, 2007.
- [19] W. M. Carlyle and R. K. Wood, "Near-shortest and  $K$ -shortest simple paths," *Networks. An International Journal*, vol. 46, no. 2, pp. 98–109, 2005.
- [20] G. Liu and K. G. Ramakrishnan, "A\* Prune: an algorithm for finding  $K$  shortest paths subject to multiple constraints," in *Proceeding of the 20th Annual Joint Conference IEEE Computer Society*, vol. 2, pp. 743–749, Anchorage, Alaska, USA, 2001.
- [21] B. Y. Chen, W. H. K. Lam, A. Sumalee, and Z.-L. Li, "Reliable shortest path finding in stochastic networks with spatial correlated link travel times," *International Journal of Geographical Information Science*, vol. 26, no. 2, pp. 365–386, 2012.
- [22] B. Y. Chen, W. H. K. Lam, Q. Li, A. Sumalee, and K. Yan, "Shortest path finding problem in stochastic time-dependent road networks with stochastic first-in-first-out property," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 4, pp. 1907–1917, 2013.
- [23] B. Y. Chen, W. H. Lam, A. Sumalee, Q. Li, H. Shao, and Z. Fang, "Finding reliable shortest paths in road networks under uncertainty," *Networks and Spatial Economics*, vol. 13, no. 2, pp. 123–148, 2013.
- [24] B. Y. Chen, W. H. K. Lam, A. Sumalee, Q. Li, and M. L. Tam, "Reliable shortest path problems in stochastic time-dependent networks," *Journal of Intelligent Transportation Systems: Technology, Planning, and Operations*, vol. 18, no. 2, pp. 177–189, 2014.
- [25] B. Y. Chen, C. Shi, J. Zhang, W. H. K. Lam, Q. Li, and S. Xiang, "Most reliable path-finding algorithm for maximizing on-time arrival probability," *Transportmetrica B*, vol. 5, no. 3, pp. 253–269, 2017.
- [26] S. Wang, H. Shao, L. Tao, and Q. Ni, "Travel time reliability-based optimal path finding," in *Proceedings of the 3rd International Joint Conference on Computational Sciences and Optimization, CSO 2010: Theoretical Development and Engineering Practice*, pp. 531–534, chn, May 2010.
- [27] T.-S. Chang, L. K. Nozick, and M. A. Turnquist, "Multiobjective path finding in stochastic dynamic networks, with application to routing hazardous materials shipments," *Transportation Science*, vol. 39, no. 3, pp. 383–399, 2005.
- [28] H. K. Lo, X. W. Luo, and B. W. Y. Siu, "Degradable transport network: travel time budget of travelers with heterogeneous risk aversion," *Transportation Research Part B: Methodological*, vol. 40, no. 9, pp. 792–806, 2006.
- [29] T. Xing and X. Zhou, "Reformulation and solution algorithms for absolute and percentile robust shortest path problems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 2, pp. 943–954, 2013.
- [30] F. Li, Z. Gao, K. Li, and D. Z. W. Wang, "Train routing model and algorithm combined with train scheduling," *Journal of Transportation Engineering*, vol. 139, no. 1, pp. 81–91, 2013.

- [31] M. L. Tam and W. H. K. Lam, "Using automatic vehicle identification data for travel time estimation in Hong Kong," *Transportmetrica*, vol. 4, no. 3, pp. 179–194, 2008.
- [32] M. L. Tam and W. H. K. Lam, "Application of automatic vehicle identification technology for real-time journey time estimation," *Information Fusion*, vol. 12, no. 1, pp. 11–19, 2011.
- [33] X. Chang, B. Y. Chen, Q. Li, X. Cui, L. Tang, and C. Liu, "Estimating real-time traffic carbon dioxide emissions based on intelligent transportation system technologies," *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, no. 1, pp. 469–479, 2013.
- [34] H. Shao, W. H. K. Lam, A. Sumalee, and A. Chen, "Journey time estimator for assessment of road network performance under demand uncertainty," *Transportation Research Part C: Emerging Technologies*, vol. 35, pp. 244–262, 2013.
- [35] H. Shao, W. H. Lam, A. Sumalee, and A. Chen, "Network-wide on-line travel time estimation with inconsistent data from multiple sensor systems under network uncertainty," *Transportmetrica A: Transport Science*, pp. 1–20, 2017.



# Hindawi

Submit your manuscripts at  
<https://www.hindawi.com>

