Applied Soft Computing 12 (2012) 1231–1237

Contents lists available at SciVerse ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Short communication

Fuzzy Dijkstra algorithm for shortest path problem under uncertain environment

Yong Deng[a,b,c,*], Yuxin Chen[b], Yajuan Zhang[a], Sankaran Mahadevan[c]

[a] School of Computer and Information Sciences, Southwest University, Chongqing 400715, China
[b] School of Electronics and Information Technology, Shanghai Jiao Tong University, Shanghai 200240, China
[c] School of Engineering, Vanderbilt University, Nashville, TN 37235, USA

ARTICLE INFO

Article history:
Received 7 June 2011
Received in revised form 18 October 2011
Accepted 1 November 2011
Available online 20 November 2011

Keywords:
Shortest path problem
Dijkstra algorithm
Fuzzy numbers
Graded mean representation

ABSTRACT

A common algorithm to solve the shortest path problem (SPP) is the Dijkstra algorithm. In this paper, a generalized Dijkstra algorithm is proposed to handle SPP in an uncertain environment. Two key issues need to be addressed in SPP with fuzzy parameters. One is how to determine the addition of two edges. The other is how to compare the distance between two different paths with their edge lengths represented by fuzzy numbers. To solve these problems, the graded mean integration representation of fuzzy numbers is adopted to improve the classical Dijkstra algorithm. A numerical example of a transportation network is used to illustrate the efficiency of the proposed method.

1. Introduction

In a transportation management system, it is necessary to provide the shortest path from a specified origin node to other destination nodes [1,2]. In a network, the lengths of the arcs are assumed to represent transportation time or cost, rather than the geographical distances. Consider an acyclic directed network $G(N, A)$, consisting of a set of nodes $N = \{1, 2, \ldots, n\}$, and m directed arcs $A \subseteq N \times N$. Each arc is denoted by an ordered pair (i, j), where $i, j \in N$. It is assumed that there is only one directed arc (i, j) from i to j. Although in classical graph theory, the weights of the edges in a shortest path finding are represented as real numbers, most practical applications, however, have parameters that are not precise (e.g. costs, capacities, demands, time, etc.) [3]. Fuzzy set theory provide the ability to handle vague information and is widely used in many fields such as environmental assessment [4,5], pattern recognition [6–8], decision making [9–12].

In finding the shortest path under uncertain environment, an appropriate modelling approach is to make use of fuzzy numbers. As a result, many researchers have paid attention to the fuzzy shortest path problem (FSPP) [13–22].

One of the most used methods to solve the shortest path problem is the Dijkstra algorithm. In the case of crisp number to model arc lengths, the Dijkstra algorithm can be easily to implemented. However, due to the reason that many optimization methods for crisp numbers cannot be applied directly to fuzzy numbers, some modifications are needed before using the classical methods. For example, one straight forward approach is to transform the fuzzy number into a crisp one. A typical work [23] transforms the trapezoidal number into crisp number by the defuzzification function, also called Yager's ranking index [24]. Generally speaking, there are two main issues that need to be solved when applying the Dijkstra algorithm in a fuzzy environment. One is the summing operation of fuzzy numbers, the other is the ranking and comparison of fuzzy numbers, which is still an open issue in fuzzy set theory research fields.

The canonical representation of operations on triangular fuzzy numbers that are based on the graded mean integration representation method leads to the result that multiplication and addition of two fuzzy numbers can be represented as a crisp number [25]. This method is widely used in many applications such as multi-criteria decision making [26–36], risk evaluation [37], portfolio selection [38], evaluation of enterprise knowledge management capabilities [39], product adoption [40], evaluation of airline service quality [41] and efficient network selection in heterogeneous wireless networks [42].

In this paper, the classical Dijkstra algorithm is generalized based on the canonical representation of operations on triangular fuzzy numbers to handle the fuzzy shortest path problem. Compared with existing methods, the proposed method is more efficient due to the fact that the summing operation and the ranking of fuzzy numbers can be done is a easy and straight manner. The paper is organized as follows. Section 2 gives a brief introduction to

* Corresponding author at: School of Computer and Information Sciences, Southwest University, Chongqing 400715, China. Fax: +86 023 68254555.
E-mail addresses: ydeng@swu.edu.cn, professordeng@hotmail.com (Y. Deng).

1568-4946/$ – see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.asoc.2011.11.011

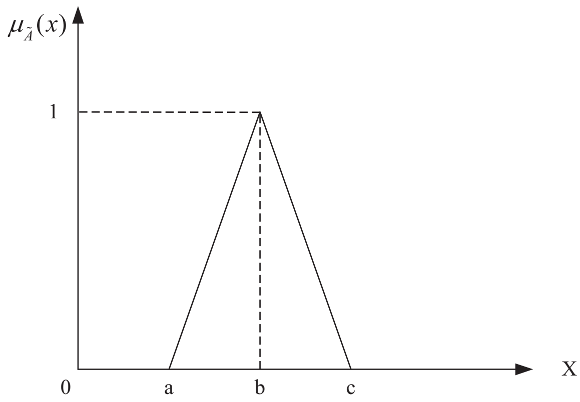

Fig. 1. A triangular fuzzy number.

the basic theory used in our proposed method, including fuzzy set theory and the Dijkstra algorithm. Section 3 develops the proposed method in detail. In Section 4, a numerical shortest path example under fuzzy environment is used to illustrate the efficiency of the proposed method. Section 5 concludes the paper.

2. Preliminaries

In this section, some basic concepts are briefly introduced including fuzzy sets theory and dynamic programming.

2.1. Fuzzy numbers

Definition 2.1. "Fuzzy set": Let X be a universe of discourse. Where $\tilde{A}$ is a fuzzy subset of X; and for all $x \in X$, there is a number $\mu_{\tilde{A}}(x) \in [0, 1]$ which is assigned to represent the membership degree of x in $\tilde{A}$, and is called the membership function of $\tilde{A}$ [13].

Definition 2.2. "Fuzzy number": A fuzzy number $\tilde{A}$ is a normal and convex fuzzy subset of X [13].

Here, "normality" implies that:

$$\exists x \in \mathbb{R}, \quad \bigvee_{x} \mu_{\tilde{A}}(x) = 1 \tag{1}$$

and "convex" means that:

$$\begin{aligned} &\forall x_1 \in X, \quad x_2 \in X, \quad \forall \alpha \in [0, 1], \mu_{\tilde{A}}(\alpha x_1 + (1-\alpha)x_2) \\ &\geq \min(\mu_{\tilde{A}}(x_1), \mu_{\tilde{A}}(x_2)) \end{aligned} \tag{2}$$

Definition 2.3. A triangular fuzzy number $\tilde{A}$ can be defined by a triplet (a, b, c), where the membership can be determined as follows.

A triangular fuzzy number $\tilde{A} = (a, b, c)$ can be shown in Fig. 1.

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, & x < a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ \dfrac{c-x}{c-b}, & b \le x \le c \\ 0, & x > c \end{cases} \tag{3}$$

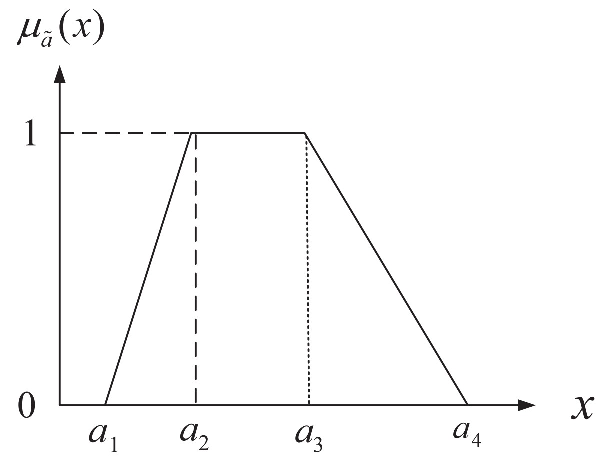

Fig. 2. A trapezoidal fuzzy number $\tilde{a}$.

Definition 2.4. A trapezoidal fuzzy number $\tilde{A}$ can be defined as $\tilde{A} = (a_1, a_2, a_3, a_4)$, where the membership can be determined as follows and shown in Fig. 2:

$$\mu_{\tilde{A}} = \begin{cases} 0 & x \le a_1 \\ \dfrac{x-a_1}{a_2-a_1} & a_1 \le x \le a_2 \\ 1 & a_2 \le x \le a_3 \\ \dfrac{a_4-x}{a_4-a_3} & a_3 \le x \le a_4 \\ 0 & a_4 \le x \end{cases} \tag{4}$$

2.2. Canonical representation of operations on fuzzy numbers

In this paper, the canonical representation of operations on triangular fuzzy numbers which are based on the graded mean integration representation method [25], is used to obtain a simple fuzzy shortest path algorithm.

Definition 2.5. Given a triangular fuzzy number $\tilde{A} = (a_1, a_2, a_3)$, the graded mean integration representation of triangular fuzzy number $\tilde{A}$ is defined as:

$$P(\tilde{A}) = \frac{1}{6}(a_1 + 4 \times a_2 + a_3) \tag{5}$$

Let $\tilde{A} = (a_1, a_2, a_3)$ and $\tilde{B} = (b_1, b_2, b_3)$ be two triangular fuzzy numbers. By applying Eq. (5), the graded mean integration representation of triangular fuzzy numbers $\tilde{A}$ and $\tilde{B}$ can be obtained, respectively, as follows:

$$P(\tilde{A}) = \frac{1}{6}(a_1 + 4 \times a_2 + a_3)$$
$$P(\tilde{B}) = \frac{1}{6}(b_1 + 4 \times b_2 + b_3)$$

The representation of the addition operation $\oplus$ on triangular fuzzy numbers $\tilde{A}$ and $\tilde{B}$ can be defined as:

$$P(\tilde{A} \oplus \tilde{B}) = P(\tilde{A}) + P(\tilde{B}) = \frac{1}{6}(a_1 + 4 \times a_2 + a_3) + \frac{1}{6}(b_1 + 4 \times b_2 + b_3) \tag{6}$$

The canonical representation of the multiplication operation on triangular fuzzy numbers $\tilde{A}$ and $\tilde{B}$ is defined as:

$$P(\tilde{A} \otimes \tilde{B}) = P(\tilde{A}) \times P(\tilde{B}) = \frac{1}{6}(a_1 + 4 \times a_2 + a_3) \times \frac{1}{6}(b_1 + 4 \times b_2 + b_3) \tag{7}$$

Table 1
The arc lengths of the network.

Arc	Membership function	Arc	Membership function	Arc	Membership function
(1,2)	(12,13,15,17)	(7,10)	(9,10,12,13)	(15,18)	(8,9,11,13)
(1,3)	(9,11,13,15)	(7,11)	(6,7,8,9)	(15,19)	(5,7,10,12)
(1,4)	(8,10,12,13)	(8,12)	(5,8,9,10)	(16,20)	(9,12,14,16)
(1,5)	(7,8,9,10)	(8,13)	(3,5,8,10)	(17,20)	(7,10,11,12)
(2,6)	(5,10,15,16)	(9,16)	(6,7,9,10)	(17,21)	(6,7,8,10)
(2,7)	(6,11,11,13)	(10,16)	(12,13,16,17)	(18,21)	(15,17,18,19)
(3,8)	(10,11,16,17)	(10,17)	(15,19,20,21)	(18,22)	(3,5,7,9)
(4,7)	(17,20,22,24)	(11,14)	(8,9,11,13)	(18,23)	(5,7,9,11)
(4,11)	(6,10,13,14)	(11,17)	(6,9,11,13)	(19,22)	(15,16,17,19)
(5,8)	(6,9,11,13)	(12,14)	(13,14,16,18)	(20,23)	(13,14,16,17)
(5,11)	(7,10,13,14)	(12,15)	(12,14,15,16)	(21,23)	(12,15,17,18)
(5,12)	(10,13,15,17)	(13,15)	(10,12,14,15)	(22,23)	(4,5,6,8)
(6,9)	(6,8,10,11)	(13,19)	(17,18,19,20)		
(6,10)	(10,11,14,15)	(14,21)	(11,12,13,14)		

Definition 2.6. Given a trapezoidal fuzzy number $\tilde{A} = (a_1, a_2, a_3, a_4)$, the graded mean integration representation of triangular fuzzy number $\tilde{A}$ is defined as:

$$P(\tilde{A}) = \frac{1}{6}(a_1 + 2 \times a_2 + 2 \times a_3 + a_4) \tag{8}$$

Similar to the triangular fuzzy numbers, the graded mean integration representation of operations on trapezoidal fuzzy numbers can also obtained. Let $\tilde{A} = (a_1, a_2, a_3, a_4)$ and $\tilde{B} = (b_1, b_2, b_3, b_4)$ be two trapezoidal fuzzy numbers. By applying Eq. (8), the graded mean integration representation of the addition operation of trapezoidal fuzzy numbers $\tilde{A}$ and $\tilde{B}$ can be defined as

$$P(\tilde{A} \oplus \tilde{B}) = \frac{1}{6}(a_1 + 2 \times a_2 + 2 \times a_3 + a_4) + \frac{1}{6}(b_1 + 2 \times b_2 + 2 \times b_3 + b_4) \tag{9}$$

The multiplication operation on trapezoidal fuzzy numbers $\tilde{A}$ and $\tilde{B}$ is defined as

$$P(\tilde{A} \otimes \tilde{B}) = \frac{1}{6}(a_1 + 2 \times a_2 + 2 \times a_3 + a_4) \times \frac{1}{6}(b_1 + 2 \times b_2 + 2 \times b_3 + b_4) \tag{10}$$

```
function fuzzy  Dijkstra(Graph, source):
    for each vertex v in Graph:                  // Initializations
        dist[v]: = infinity;                     // Unknown distance function from source to v
        previous[v]: = undefined;                // Previous node in optimal path from source
    end for;
    dist[source]: = 0;                           // Distance from source to source
    Q := the set of all nodes in Graph;          // All nodes in the graph are unoptimized - thus are in Q
    while Q is not empty:                        // The main loop
        u := vertex in Q with smallest dist[];
        if dist[u] = infinity:
            break;                               // all remaining vertices are inaccessible from source
        end if;
        remove u from Q;
        for each neighbor v of u:                // where v has not yet been removed from Q.
            alt := dist[u] + dist_between(u, v);
          // the graded mean integration representation of addition operation of fuzzy numbers is used
            if alt < dist[v]:                    // Relax (u,v,a)
                dist[v] := alt;
                previous[v] := u;
                decrease-key v in Q;             // Reorder v in the Queue
            end if;
        end for;
    end while;
    return dist[];
end Dijkstra.
```

Fig. 3. Pseudocode of the proposed fuzzy Dijkstra algorithm.

For example, see Fig. 5 and Table 1. From node 17, there are two routes to node 23. One is $17 \to 20 \to 23$ and the other is $17 \to 21 \to 23$. The use of the canonical representation of the addition operation on fuzzy numbers in shortest path finding problem can be illustrated as follows. For the first route $17 \to 20 \to 23$, the length can be obtained as

$$
\begin{aligned}
& Arc_1(17, 23) \\
& = Arc(17, 20) \oplus Arc(20, 23) \\
& = (7, 10, 11, 12) \oplus (13, 14, 16, 17) \\
& = \frac{1}{6}(7 + 2 \times 10 + 2 \times 11 + 12) + \frac{1}{6}(13 + 2 \times 14 + 2 \times 16 + 17) \\
& = \frac{151}{6}
\end{aligned}
$$

For the second route, the length can be obtained as

$$
\begin{aligned}
& Arc_2(17, 23) \\
& = Arc(17, 21) \oplus Arc(21, 23) \\
& = (6, 7, 8, 10) \oplus (12, 15, 17, 18) \\
& = \frac{1}{6}(6 + 2 \times 7 + 2 \times 8 + 10) + \frac{1}{6}(12 + 2 \times 15 + 2 \times 17 + 18) \\
& = \frac{140}{6}
\end{aligned}
$$

The result shows that the former is worse than the latter since $\frac{151}{6} > \frac{140}{6}$. As can be seen from the example above, one merit of the canonical representation of the addition operation is that its result is a crisp number. The decision making can be easily obtained without the process of ranking fuzzy numbers, commonly used in many other fuzzy shortest path problems. This is very advantageous in the Dijkstra algorithm under fuzzy environment.

2.3. Dijkstra algorithm

Dijkstra's algorithm, was conceived by the Dutch computer scientist Edsger Dijkstra in 1956 and published in 1959 [43].

```
S := empty sequence
u := target
while previous[u] is defined:
      insert u at the beginning of S
      u := previous[u]
```

Fig. 4. Pseudocode for shortest path from source to target.

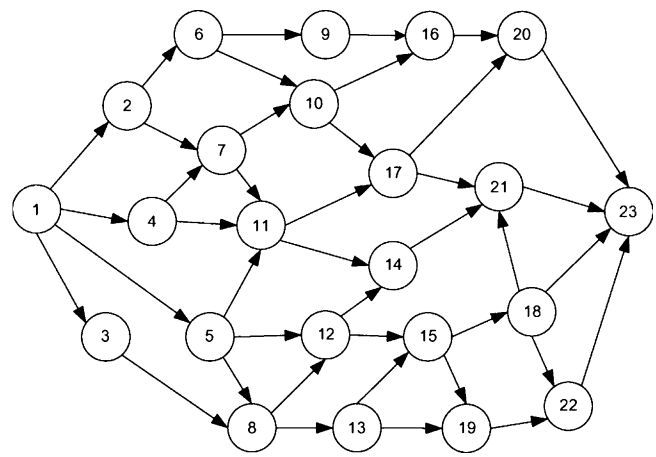

Fig. 5. A transportation network.

For a given source vertex (node) in the graph, the algorithm finds the path with lowest cost (i.e. the shortest path) between that vertex and every other vertex. It can also be used for finding costs of shortest paths from a single vertex to a single destination vertex by stopping the algorithm once the shortest path to the destination vertex has been determined. For example, if the vertices of the

Step	Set *S* (shortest paths)	Set *U* (unsettled)
1	***S***={(1)} (1)→(1)=0 Seek from (1)	***U***={(2),(3), ∞ ,(23)} (1)→(2)=14.1667 (1)→(3)=12.0000 (1)→(4)=10.8333 (1)→(5)=8.5000 (1)→others=∞ Find out the shortest path: **(1)→(5)=8.5000**
2	***S***={(1),(5)} (1)→(1)=0 (1)→(5)=8.5000 Seek from (1)→(5)	***U***={(2),(3),(4),(6), ∞ ,(23)} (1)→(5)→(8) =18.3333 (1)→(5)→(11)=19.6667 (1)→(5)→(12)=22.3333 (1)→(5)→others=∞ Find out the shortest path: **(1)→(4)=10.8333**
3	***S***={(1),(5),(4)} (1)→(1)=0 (1)→(5)=8.5000 (1)→(4)=10.8333 Seek from (1)→(4)	***U***={(2),(3),(6), ∞ ,(23)} (1)→(4)→(7) =31.6666 (1)→(4)→(11)=21.8333 (abandon) >(1)→(5)→(11)=19.6667 (1)→(4)→others=∞ Find out the shortest path: **(1)→(3)=12.0000**

Fig. 6. The first three steps of fuzzy Dijkstra algorithm in numerical example.

Step	Set *S* (shortest paths)	Set *U* (unsettled)
22	***S***={(1),(5),(4),(3),(2),(8),(11),(12),(7),(13),(6),(17),(14),(9),(10),(15),(21), (20), (16),(19),(18),(23)} (1)→(1)=0 (1)→(5)=8.5000 (1)→(4)=10.8333 (1)→(3)=12.0000 (1)→(2)=14.1667 (1)→(5)→(8) =18.3333 (1)→(5)→(11)=19.6667 (1)→(5)→(12)=22.3333 (1)→(5)→(8)→(13)=24.8333 (1)→(2)→(6)=26 (1)→(5)→(11)→(17)=29.1667 (1)→(5)→(11)→(14)=29.8334 (1)→(2)→(6)→(9)=34.8333 (1)→(2)→(7)→(10)=35.6667 (1)→(5)→(12)→(15)=36.6666 (1)→(5)→(11)→(17)→(20)=39.3334 (1)→(2)→(6)→(9)→(16)=42.8333 (1)→(5)→(8)→(13)→(19)=43.3333 (1)→(5)→(12)→(15)→(18)=46.8333 (1)→(5)→(11)→(17)→(21)→(23)= 52.5001 Seek from (1)→(5)→(11)→(17)→(21)→(23)	***U***={(22) } (1)→(5)→(11)→(17)→(21)→(23)→others=∞ Find out the shortest path: **(1)→(5)→(12)→(15)→(18)→(22)= 52.8333**

Fig. 7. Step 22 of the fuzzy Dijkstra algorithm in the numerical example.

graph represent cities and edge path costs represent driving distances between pairs of cities connected by a direct road, Dijkstra's algorithm can be used to find the shortest route between one city and any other city.

Let the node at the beginning of the path be called the **origin node**. Let the **distance of node** *Y* be the distance from the **origin node** to *Y*. Dijkstra's algorithm will assign some initial distance values and will try to improve them step by step.

1. Assign to every node a distance value: set it to zero for the origin node and to infinity for all other nodes.
2. Mark all nodes as unvisited. Set initial node as current.
3. For current node, consider all its unvisited neighbors and calculate their tentative distance. For example, if current node *A* has distance of 6, and an edge connecting it with another node *B* has length 2, the distance to *B* through *A* will be 6 + 2 = 8. If this distance is less than the previously recorded distance, overwrite the distance.
4. Considering all neighbors of the current node, mark it as visited. A visited node will not be checked again; its distance recorded is final and minimal.
5. If all nodes have been visited, stop. Otherwise, set the unvisited node with the smallest distance (from the initial node, considering all nodes in the graph) as the next "current node" and continue from step 3.

The details information of Dijkstra's algorithm can be found in [44].

3. Proposed method

In a fuzzy environment, two issues, namely the addition of fuzzy numbers and ranking of numbers should be solved. Based on the canonic representation [25], the classical Dijkstra algorithm can be easily generalized to a fuzzy Dijkstra algorithm as follows.

In the proposed method, it is necessary to clarify some basic notation. *Q* is the set of all unvisited vertex (nodes) that are to be removed. In set *Q*, *u* represents the vertex with the smallest distance from the source node. Let *v* be one variant (cf. Fig. 3), $dist[v]$ denote the current shortest distance between *v* and the source node, $dist\_between(u, v)$ the distance between two vertices *u*, *v*, $previous[v]$ the nearest vertex to the current *v* along the shortest path from the source node. The sign 'alt' is an alternative variant of distance used for comparison. At the initial state, for each vertex *v* in graph (excluding the source node), $dist[v]$=infinity and $previous[v]$=0 and *Q* is the set of all nodes in graph. For the source node, $dist[source]$=0. Then the following three steps are repeated till completion.

1. Check if *Q* is empty. While *Q* is not empty, seek *u* in *Q*. If $dist[u]$ is not infinity, then from *Q*, *u* is removed (this means that all remaining vertices are inaccessible from source and the analysis is completed).
2. After finding *v*, each neighbor node of *u*, and calculate $alt = dist[u] + dist_{between}(u, v)$ by using the canonical representation operation introduced in Section 2. If there exists $alt < dist[v]$, then replace $dist[v]$ with *alt* and record the previous $[v] = u$ (if

Step	Set *S* (shortest paths)	Set *U* (unsettled)
23	***S***={(1),(5),(4),(3),(2),(8),(11),(12),(7),(13),(6),(17),(14),(9),(10),(15),(21), (20), (16),(19),(18),(23),(22)} (1)→(1)=0 (1)→(5)=8.5000 (1)→(4)=10.8333 (1)→(3)=12.0000 (1)→(2)=14.1667 (1)→(5)→(8) =18.3333 (1)→(5)→(11)=19.6667 (1)→(5)→(12)=22.3333 (1)→(5)→(8)→(13)=24.8333 (1)→(2)→(6)=26 (1)→(5)→(11)→(17)=29.1667 (1)→(5)→(11)→(14)=29.8334 (1)→(2)→(6)→(9)=34.8333 (1)→(2)→(7)→(10)=35.6667 (1)→(5)→(12)→(15)=36.6666 (1)→(5)→(11)→(17)→(20)=39.3334 (1)→(2)→(6)→(9)→(16)=42.8333 (1)→(5)→(8)→(13)→(19)=43.3333 (1)→(5)→(12)→(15)→(18)=46.8333 (1)→(5)→(11)→(17)→(21)→(23)= 52.5001 (1)→(5)→(12)→(15)→(18)→(22)= 52.8333	***U***={ } *DONE*

Fig. 8. The final step of fuzzy Dijkstra algorithm in numerical example.

not, the previous record of each neighbor vertex v remains the same).

3. Choose the one among each v with shortest $dist[v]$, and replace it with u and go to Step 1. The pseudocode of the proposed fuzzy Dijkstra algorithm is shown in Fig. 3.

If only the shortest path between a single source and target pair is of interest, the search can be terminated at line 13 if $u = target$. Now the shortest path from origin to target can be found, as shown in Fig. 4. Sequence S is the list of vertices constituting one of the shortest paths from source to target, or the empty sequence if no path exists.

4. Transportation network application

In this section, a numerical example of the fuzzy shortest path problem, illustrated in [17,18], is used to show the efficiency of the proposed method. Consider the transportation network shown in Fig. 5, with 23 nodes and 40 arcs. It is assumed that all arc lengths are trapezoidal fuzzy numbers with membership functions as shown in Table 1.

The first three steps are detailed in Fig. 6. Let S be the set of visited nodes representing the shortest path and U the set of unvisited nodes which are to be settled.

In Step 1, node (1), which is the source node, is moved from U to S, and the shortest distance from (1) to (1) is zero, while the distance from source node to accessible neighbor nodes are calculated. Among them, the shortest one (1)→(5) is picked out along with distance value 8.5.

In Step 2, we move (5) from U to S, this time the seeking of shortest path should be begun from (5). The shortest distance from (1) → (5) to its neighbors are calculated. None of them is shorter than the path (1)→(4) by comparing all the records in the column Set U.

So in Step 3, (4) is moved and all the data starts with (4) and (1) (3) with distance 12.000 is the result.

Due to the limitation of space, the other steps are not illustrated. Only Steps 22 and 23, the final two steps in this numerical example, are shown in Figs. 7 and 8, respectively. At the end of Step 21, the shortest path is (1)→(5)→(11)→(17)→(21)→(23). However, there are no neighbor nodes accessible from (23), and only (1)→(5)→(12)→(15)→(18)→(22) is found from the previous record of column Set U. As (22) is drawn to S, U becomes an empty set and the search is complete. The shortest path, whose length is 52.83 based on the proposed fuzzy Dijkstra algorithm, is (1)→(5)→(11)→(17)→(21)→(23), which is the same in Refs. [17,18].

5. Conclusion

This paper extended the Dijkstra algorithm to solve the shortest path problem with fuzzy arc lengths. Two key issues are addressed. One is how to determine the addition of two edges. The other is how to compare the distance between two different paths when their edges length are represented by fuzzy numbers. The proposed method to find the shortest path under fuzzy arc lengths is based on the graded mean integration representation of fuzzy numbers. A numerical example was used to illustrate the efficiency of the proposed method. The proposed method can be applied to real applications in transportation systems, logistics management, and many other network optimization problem that can be formulated as shortest path problem.

It should be pointed out that the uncertainty in the shortest path problem is not limited to the geometric distance. For example, due to the weather and other unexpected factors, the travel time from one city to another city may be represented as a fuzzy variable, even if the geometric distance is fixed.

Acknowledgements

The authors thank the anonymous reviews for their valuable suggestions. The Ph.D. students of the first author in Shanghai Jiao Tong University, X.Y. Su and P.D. Xu, provided valuable discussions. The work is partially supported by National Natural Science Foundation of China, Grant Nos. 60874105, 61174022, Program for New Century Excellent Talents in University, Grant No. NCET-08-0345, Chongqing Natural Science Foundation, Grant No. CSCT, 2010BA2003, Aviation Science Foundation, Grant No. 20090557004, the Fundamental Research Funds for the Central Universities, Grant No. XDJK2010C030, the Fundamental Research Funds for the Central Universities, Grant No. XDJK2011D002, the Southwest University Scientific & Technological Innovation Fund for Graduates, Grant No. ky2011011, and Doctor Funding of Southwest University, Grant No. SWU110021.

References

[1] M. Xu, Y. Liu, Q. Huang, Y. Zhang, G. Luan, An improved Dijkstra shortest path algorithm for sparse network, Applied Mathematics and Computation 185 (2007) 247–254.
[2] X. Lu, M. Camitz, Finding the shortest paths by node combination, Applied Mathematics and Computation 217 (2011) 6401–6408.
[3] T.N. Chuang, J.Y. Kung, The fuzzy shortest path length and the corresponding shortest path in a network, Computers & Operations Research 32 (2005) 1409–1428.
[4] R. Sadiq, S. Tesfamariam, Developing environmental using fuzzy numbers ordered weighted averaging (FN-OWA) operators, Stochastic Environmental Research and Risk Assessment 22 (2008) 495–505.
[5] Y. Deng, W. Jiang, R. Sadiq, Modeling contaminant intrusion in water distribution networks: a new similarity-based DST method, Expert Systems with Applications 38 (2011) 571–578.
[6] Y. Deng, W.K. Shi, F. Du, Q. Liu, A new similarity measure of generalized fuzzy numbers and its application to pattern recognition, Pattern Recognition Letters 25 (2004) 875–883.
[7] H.W. Liu, New similarity measures between intuitionistic fuzzy sets and between elements, Mathematical and Computer Modelling 42 (2005) 61–70.
[8] J. Ye, Cosine similarity measures for intuitionistic fuzzy sets and their applications, Mathematical and Computer Modelling 53 (2011) 91–97.
[9] Y. Deng, Plant location selection based on fuzzy topsis, International Journal of Advanced Manufacturing Technology 28 (2006) 839–844.
[10] Y. Deng, F.T.S. Chan, A new fuzzy Dempster MCDM method and its application in supplier selection, Expert Systems with Applications 38 (2011) 6985–6993.
[11] Y. Deng, F.T.S. Chan, Y. Wu, D. Wang, A new linguistic MCDM method based on multiple-criterion data fusion, Expert Systems with Applications 38 (2011) 9854–9861.
[12] Y. Deng, R. Sadiq, W. Jiang, S. Tesfamariam, Risk analysis in a linguistic environment: a fuzzy evidential reasoning-based approach, Expert Systems with Applications (2011), doi:10.1016/j.eswa.2011.06.018.
[13] D. Dubois, H. Prade, Fuzzy Sets and Systems: Theory and Applications, Academic Press, New York, 1980.
[14] C. Lin, M.S. Chern, The fuzzy shortest path problem and its most vital arcs, Fuzzy Sets and Systems 58 (1993) 343–353.
[15] A. Boulmakoul, Generalized path-finding algorithms on semirings and the fuzzy shortest path problem, Journal of Computational and Applied Mathematics 162 (2004) 263–272.
[16] T.N. Chuang, J.Y. Kung, A new algorithm for the discrete fuzzy shortest path problem in a network, Applied Mathematics and Computation 174 (2006) 1660–1668.
[17] X. Ji, K. Iwamura, Z. Shao, New models for shortest path problem with fuzzy arc lengths, Applied Mathematical Modelling 31 (2007) 259–269.
[18] I. Mahdavi, R. Nourifar, A. Heidarzade, N.M. Amiri, A dynamic programming approach for finding shortest chains in a fuzzy network, Applied Soft Computing 9 (2009) 503–511.
[19] M. Ghatee, S.M. Hashemi, M. Zarepisheh, E. Khorram, Preemptive priority-based algorithms for fuzzy minimal cost flow problem: an application in hazardous materials transportation, Computers & Industrial Engineering 57 (2009) 341–354.
[20] M. Ghatee, S.M. Hashemi, Application of fuzzy minimum cost flow problems to network design under uncertainty, Fuzzy Sets and Systems 160 (2009) 3263–3289.
[21] E. Keshavarz, E. Khorram, A fuzzy shortest path with the highest reliability, Journal of Computational and Applied Mathematics 230 (2009) 204–212.
[22] A. Tajdin, I. Mahdavi, N.M. Amiri, B. Sadeghpour-Gildeh, Computing a fuzzy shortest path in a network with mixed fuzzy arc lengths using a-cuts, Computers and Mathematics with Applications 60 (2010) 989–1002.
[23] S.T. Liu, C. Kao, Network flow problems with fuzzy arc lengths, IEEE Transaction on Systems, Man, and Cybernetics. Part B: Cybernetics 34 (2004) 765–769.
[24] R.R. Yager, A procedure for ordering fuzzy subsets of the unit interval, Information Sciences 24 (1981) 143–151.
[25] C.C. Chou, The canonical representation of multiplication operation on triangular fuzzy numbers, Computers and Mathematics with Applications 45 (2003) 1601–1610.
[26] C.C. Chou, P.C. Chang, A fuzzy multiple criteria decision making model for selecting the distribution center location in china: a Taiwanese manufacturer's perspective, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 5618 (2009) 140–148.
[27] C.C. Chou, Application of FMCDM model to selecting the hub location in the marine transportation: A case study in southeastern Asia, Mathematical and Computer Modelling 51 (2010) 791–801.
[28] C.C. Chou, R.H. Gou, C.L. Tsai, M.C. Tsou, C.P. Wong, H.L. Yu, Application of a mixed fuzzy decision making and optimization programming model to the empty container allocation, Applied Soft Computing Journal 10 (2010) 1071–1079.
[29] C.C. Chou, F.T. Kuo, R.H. Gou, C.L. Tsai, C.P. Wong, M.C. Tsou, Application of a combined fuzzy multiple criteria decision making and optimization programming model to the container transportation demand split, Applied Soft Computing Journal 10 (2010) 1080–1086.
[30] C.C. Chou, An integrated quantitative and qualitative fmcdm model for location choices, Soft Computing 14 (2010) 757–771.
[31] C.C. Chou, A fuzzy backorder inventory model and application to determining the optimal empty-container quantity at a port, International Journal of Innovative Computing, Information and Control 5 (2009) 4825–4834.
[32] I. Chamodrakas, I. Leftheriotis, D. Martakos, In-depth analysis and simulation study of an innovative fuzzy approach for ranking alternatives in multiple attribute decision making problems based on topsis, Applied Soft Computing Journal 11 (2011) 900–907.
[33] M. Tseng, A. Chiu, Evaluating firm's green supply chain management in linguistic preferences, Journal of Cleaner Production (2010), doi:10.1016/j.jclepro.2010.08.007.
[34] M. Tseng, Green supply chain management with linguistic preferences and incomplete information, Applied Soft Computing. 11 (2011) 4894–4903.
[35] M. Tseng, Using a hybrid mcdm model to evaluate firm environmental knowledge management in uncertainty, Applied Soft Computing 11 (2011) 1340–1352.
[36] M. Tseng, Using linguistic preferences and grey relational analysis to evaluate the environmental knowledge management capacity, Expert systems with applications 37 (2010) 70–81.
[37] X. Pan, H. Wang, W. Chang, A fuzzy synthetic evaluation method for failure risk of aviation product R&D project, in: 5th IEEE International Conference on Management of Innovation and Technology, ICMIT, 2010, pp. 1106–1111.
[38] W. Chen, S. Tan, Fuzzy portfolio selection problem under uncertain exit time, IEEE International Conference on Fuzzy Systems 5 (2009) 550–554.
[39] R.G. Qi, S.G. Liu, Research on comprehensive evaluation of enterprises knowledge management capabilities, in: 2010 International Conference on Management Science and Engineering, ICMSE, 2010, pp. 1036–1301.
[40] S. Kim, K. Lee, J.K. Cho, C.O. Kim, Agent-based diffusion model for an automobile market with fuzzy topsis-based product adoption process, Expert Systems with Applications 38 (2011) 7270–7276.
[41] C.C. Chou, L.J. Liu, S.F. Huang, J.M. Yih, T.C. Han, An evaluation of airline service quality using the fuzzy weighted SERVQUAL method, Applied Soft Computing Journal 11 (2011) 2117–2128.
[42] I. Chamodrakas, D. Martakos, A utility-based fuzzy topsis method for energy efficient network selection in heterogeneous wireless networks, Applied Soft Computing Journal 11 (2011) 3734–3743.
[43] E.W. Dijkstra, A note on two problems in connexion with graphs, Numerische Mathematik 1 (1959) 269–271.
[44] T.H. Cormen, C.E. Leiserson, R.L. Rivest, C. Stein, Introduction to Algorithms, MIT Press and McGraw-Hill, 2001.