



An algorithmic approach for finding the fuzzy constrained shortest paths in a fuzzy graph

Xiaoqun Liao¹ · JiaYi Wang² · Li Ma²

Received: 8 February 2020 / Accepted: 12 April 2020 / Published online: 27 June 2020
© The Author(s) 2020

Abstract

Shortest path problem (SPP) is a fundamental and well-known combinatorial optimization problem in the area of graph theory. In real-life scenarios, the arc weights in a shortest path of a network/graph have the several parameters which are very hard to define exactly (i.e., capacity, cost, demand, traffic frequency, time, etc.). We can incorporate the fuzziness into a graph to handle this type of uncertain situation. In this manuscript, we propose the idea of constrained SPP (CSPP) in fuzzy environment. CSPP has an useful real-life application in online cab booking system. The main motivation of this study is to determine a path with minimal cost where traveling time within two locations does not more than predetermined time. We can not predicate the exact time and cost of the path due to uncertain traffic scenarios and another unexpected reasons; still, the geometrical distance between the locations is fixed. Here, we use trapezoidal fuzzy number to describe the edge weight of a fuzzy network/graph for CSPP. We define this CSPP as fuzzy CSPP (FCSPP). The utility of FCSPP is described in several real-life scenarios. We propose a mathematical formulation for the FCSPP and an algorithm is proposed for solving the FCSPP. We describe an application of our proposed algorithm on an online cab booking system.

Keywords Fuzzy graph · SPP · CSPP · Fuzzy shortest path problem · Online cab booking system

Introduction

The SPP among the specified starting vertex and a target vertex is a popular combinatorial optimization problem in operation research and graph theory. We emerge graph/network as a mathematical model of many optimization problems, e.g., traffic scheduling, transportation system, telecommunication system, computer wireless network, texture mapping, supply demand management, computer wireless network, etc. Many real-life problems can be remodeled as a path finding problem between one node and an another node in a network/graph. The optimal path is calculated based on total number of pre-

determined measure. We evaluate the optimality measure in terms of arc lengths which are real numbers, and each and every arc is associated with those real values of the network. In any real-life network, an arc/link is related with a specific direction and with an evaluate of impedance, computing the several parameters, such as traveling cost, traveling time, traffic condition, capacity, demand, etc., along the graph. In most of real-world applications, such as transportation, communication, vehicle green scheduling, routing, etc., which are very much associated with several environmental matters, the edge costs may be changeable. Since it depends on the variation with traffic issues or weather issues, however, the length between two location is always constant. Therefore, ticketing the accurate path cost in such real-world networks/graphs always be very hard. We can use the idea of fuzziness into a graph to handle this uncertain environment.

In any metro city, there are many route and different paths between two cities. It is very useful information to determine minimum cost and time of transportation, the least traffic issues, the reliable path, etc. This type of problem can be modeled as network/graph and the shortest path is needed to make a decision of this problem. For e.g., in the ambulance service stations, the main object is to visit the ambulance to

✉ Xiaoqun Liao
lxq@xust.edu.cn

JiaYi Wang
wangjiayi041128@163.com

Li Ma
mary@xust.edu.cn

¹ Information and Network Center, Xi'an University of Science and Technology, Xi'an 710054, China

² Department of Telecommunication and Information, Xi'an University of Science and Technology, Xi'an 710054, China

patient home within the specific network in the minimum possible time. For any disaster management term, the purpose is to search for the least destructed route for evacuation. For any online cab booking system, it also needs to determine a route in the city with the minimum time/lowest cost. The SPP is said to CSPP [1–4]. If and only if the time limit is considered into account. For example, the main objective of a online cab booking system is to decide a path with the lowest traveling cost, so that cars could arrive to destination locations in the minimum possible time. These types of transportation companies are new establish throughout the world; however, nowadays, these companies are very much popular in India. Online cab booking system was first established in 2010 in India. Uber Cab service is an online based cab company in India. It has introduced itself easily and cheapest available online cab service. This company has begun with their own website, and now, this company has established their own app. This cab company has many cab drivers in China, India, and many other major metro cities. The arrival time of the online cab into the transportation network minimizes the transportation cost of passenger. Uber Cab service has their own technique to find and prepare a fair amount of the cab. From the data of the starting and destination locations of each and every passenger in different clock time, weather condition and traffic scenarios are gradually gathered by their algorithms. Their algorithms provide the cab drivers to have a host of fare amount at various specific clock times that increase their avidity to do in this system. Routing of cab service is not efficient based on past traffic information. This is the main reason that there would sometimes be varied in the traffic condition due to route repairs or accidents. That is why, the application of uncertain data such as fuzzy numbers, as the parameters of the CSPP, is much more useful than crisp numbers.

The SPP with real values as edge weights has been researched by some great researchers, and they have introduced several efficient algorithms to solve the SPP. These algorithmic methods are considered as precise shortest path algorithms. However, in real life, there are frequently several types of uncertainty encountered in arc lengths of a network due to incomplete information, imperfect information, failure, or some unexpected reasons. In such cases, algorithms of determining the shortest paths of network are considered as approximate shortest algorithms. In uncertain environments, it is more naturalistic and authentic to use SPP with fuzzy arc lengths. While introducing an algorithmic method for working FSPP, we need ranking and addition methods between two fuzzy number. SPP with different types fuzzy number as arc lengths has received attention of several researchers over the last 2 decades. This type of problem in fuzzy environment is called fuzzy SPP (FSPP). There are several studies in FSPP [5–27]. An algorithmic technique for FSPP was developed for the first time by Dubois and Prade [5]. They have extended

the standard Ford–Moore–Bellman algorithm to solve the FSPP. They have described the concept of criticality of any path. Let $D_{x,y}$ be a fuzzy number, which describes the length of shortest path between the vertex x and vertex y , and the criticality value of any path possible p is described the possibility value that the path will be the shortest path. They have assumed the fuzzy membership value of p of the shortest paths between the vertex x and vertex y . $D_{x,y}$ is used as a possibility of being the shortest path among the vertex x and vertex y . However, the main disadvantage of the algorithmic technique is unable to determine the length of the shortest path. Chanas and Kamburowski [9] described a fuzzy relationship to chose a specific path and applied it in their introduced algorithm for solving FSPP. The main drawback of their algorithm is that they have not considered the fuzzy membership function as discussed in [5]. Yager [14] introduced a method for solving FSPP using the concept of possibility measurement. He considered a possibility of visiting among two nodes and determined the total possibility of any path between a starting vertex and destination vertex utilizing the T-norm. Their technique determines the path with highest possibility. They presented a searching method to eliminate the combinatorial increasing of paths which reduces the required time of the introduced algorithm. In [15], the authors described each edge with a significant value in the graph. The most significant arc eliminates from the graph enhances the length of the shortest path maximum. The edge weight of their fuzzy graph is presented using fuzzy number. The authors have introduced a fuzzy membership degree of the FSPP by modeling this problem as a linear programming model. Using this idea, they have developed an algorithmic approach for finding the one most significant edge in a fuzzy graph. Okada and Soper [16] used triangular fuzzy number to represent edge weight of the fuzzy graph. The authors developed an order relationship between the different fuzzy numbers using the idea of “fuzzy min” and “fuzzy max” which used to generate all nondominated paths of the graph. Their proposed method determines each and every nondominated paths within the starting vertex and all other vertices. They have used the multiple labeling [28] approach for figuring out the multicriteria SPP. The main disadvantage of their proposed method is that all the possible paths within starting vertex and goal vertex cannot be determined by the expert. For any big graph, the total number of paths can be very high. In such case, they suggested the concept of h -nondominated paths. The lower number of h -nondominated path always has maximum the possibility value of h . In [11], the author has also considered fuzzy numbers to represent the edge weight of a directed fuzzy graph. They have used the fuzzy addition operation to determine cost of the path. Every path within starting node and another node is assumed to be dependent, because the two different paths may be shared some common edges. Each and every edge has a specific possibility grade

to present in a shortest path. They have considered α -cut of triangular fuzzy number the multiple labeling approach in their algorithm. In [12], Takahashi et al. presented the FSPP and extended the method introduced by Okada [11]. They have presented a genetic algorithm to find the solution of FSPP for large graph. Nayeem and Pal [17] denoted the edge weights of a fuzzy graph by imprecise numbers. Those imprecise numbers are represented by triangular fuzzy number and interval number. Their proposed algorithm determines the shortest path and their algorithm can work with two different types of uncertain numbers. They have proposed the concept of index of acceptability for describing a grading method between the different fuzzy numbers. The degree of satisfaction of the expert is calculated based on value of acceptability index. However, the drawback of their proposed method is that it can provide several different shortest paths for a specific index value of acceptability [29,30]. Moazeni [18] used a positive fuzzy quantity to represent the each edge weight of the fuzzy graph. They described a lexicographic-order relationship between the edges. Using the extension principle of fuzzy set, they introduced a method for computing all non-dominated paths between a source vertices and every other vertices on a fuzzy network. In their method, they used the idea of Dijkstra's algorithm [31] and Hansen's labeling algorithm [28]. In [19], the authors presented a generic method for solving SPP in fuzzy environment. They have assumed the triangular fuzzy numbers to represent the edge weights. They have applied a generic ranking method in their algorithm for comparing the fuzzy edges. This algorithmic method is modification of Bellman Ford technique. In [13], Mahdavi et al. developed an algorithm based on dynamic programming to find the shortest path in fuzzy environment. Their proposed ranking technique can able to eliminate to generate the Pareto shortest paths, because there may exist high number of possible Pareto shortest paths in a large fuzzy network. In such scenarios, it will be difficult for an expert to chose a specific path. Deng et al. [20] have modified the simple Dijkstra's method to find the solution of shortest path problem with fuzzy edge weight. They have used the idea of graded mean method of triangular fuzzy numbers in their proposed algorithm to determine the fuzzy shortest path length and fuzzy shortest path among the all possible paths. Hassanzadeh et al. [7] presented a method to determine the fuzzy shortest path in a fuzzy graph. They introduced an addition of fuzzy number applying α cut to determine the length of path. They have developed a least square method to determine an approximation membership grade for any addition operation in their proposed addition operation. A genetic algorithm is developed to solve the FSPP. This algorithm can deal with FSPP for a large fuzzy graph.

The CSPP determines the shortest path between a specific starting vertex and a specific goal vertex with in a directed graph under some predetermined constraint, e.g., time, cost,

distance, etc. The value of the constraint must not be greater than upper limit. CSPP is NP-hard problem and it is studied as a special type of shortest path problem. Several real-world problems can be modeled as CSPP, e.g., aircraft scheduling [32–34], communications networks [35,36], etc.

Generally, methods of determining the solutions of CSPP by optimization algorithms are classified into three categories: path ranking algorithms, lagrangian relaxation algorithms, and dynamic programming approach. The lagrangian relaxation algorithm for CSPP first determines some specific Lagrangian dual of CSPP and different techniques are used to minimize the lagrangian duality gap. In [1], the authors proposed a Lagrangian relaxation method to find the solutions of the Lagrangian dual problem. They have used the shortest path algorithm to minimize the duality gap, stopping with the starting path meeting the constraint. An algorithm is introduced which looked for in this way. Their algorithm is applied to find a shortest path between the starting vertex and goal vertex in a graph, subject to constraint. Carlyle et al. [3] introduced an algorithmic technique for CSPP.

The main motivation of this manuscript is to introduce a special type of fuzzy shortest path problem which will be very efficient for real-life problems. As far as this author's knowledge, most of the studies made on FSPP focused on fuzzy graphs with several types of fuzzy numbers as the edge weights, and there exists no study on the CSPP with fuzzy numbers as edge weights in the literature. We need to formulate the time and cost for solving the CSPP problem. It is impossible to prediction the exact traveling time and traveling cost of a path due to traffic jam and weather scenarios. In this paper, we present the constrained shortest path problem with edge weights as trapezoidal fuzzy numbers. We call this problem as fuzzy constrained shortest path problem. Here, the edge weights are used to represent the cost and time. In this study, we present a mathematical formulation for a CSPP with fuzzy arc lengths. LINGO software is applied to find the solution the FCSPP. An algorithm is also introduced to solve this CSPP in fuzzy environment. We have also modified the Disktra's algorithm and a genetic algorithm to find the solution of this problem. Finally, we present an application of the proposed model on online-based cab service.

Preliminaries

In this section, some significant and basic concepts of fuzzy set, fuzzy graph, triangular fuzzy number, and ranking and addition operation on fuzzy numbers are described.

Definition 1 There are many types in which a network can be fuzzy network. Below we have described the different ways of fuzziness possible in networks.

Type 1: A fuzzy graph/network makes up with a fuzzy collection of crisp networks.

Type 2: A fuzzy graph/network makes up with a collection of crisp nodes and fuzzy arcs.

Type 3: A fuzzy graph/network makes up with a collection of crisp nodes and fuzzy arcs with connectivity.

Type 4: A fuzzy graph/network makes up with a collection of fuzzy nodes and crisp arcs.

Type 5: A fuzzy graph/network makes up with a crisp graph with fuzzy arc lengths.

Definition 2 A fuzzy number is a special type of fuzzy set which satisfies the following four conditions.

- (a) This fuzzy set must be convex.
 - (b) This fuzzy set must be normalized.
 - (c) The membership grade of this fuzzy set must be exactly continuous.
 - (d) The membership grade must be in the real number.
- There are various different types of fuzzy number in the literature, e.g., bell shape, triangular, trapezoidal, and L–R.

Definition 3 Let $\tilde{N} = \{\underline{n}, n_0, \bar{n}\}$ be a triangular fuzzy number (TFN). The fuzzy number \tilde{N} is described by a membership function $\mu_{\tilde{N}}(y)$ and it is defined by the following equations:

$$\mu_{\tilde{N}}(y) = \begin{cases} \frac{y-\underline{n}}{n_0-\underline{n}} & \text{if } n_0 \geq y \geq \underline{n} \\ \frac{\bar{n}-y}{\bar{n}-n_0} & \text{if } \bar{n} \geq y \geq n_0 \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

Here, \underline{n} is the lowest possible value, n_0 is the primary possible value, and \bar{n} is the maximum possible value. Figure 1 is used to represent the graph of a TFN.

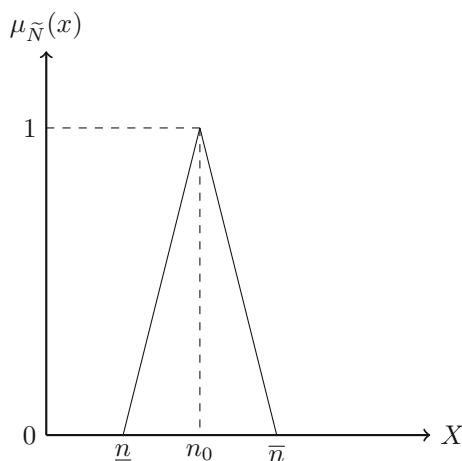


Fig. 1 A TFN

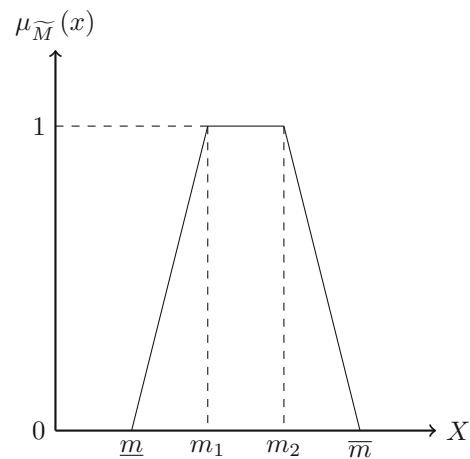


Fig. 2 A TrFN

Definition 4 Let $\tilde{M} = \{\underline{m}, m_1, m_2, \bar{m}\}$ be a trapezoidal fuzzy number (TrFN). \tilde{M} always has a trapezoidal shaped membership function. The membership function $\mu_{\tilde{M}}$ satisfies the following equations:

$$\mu_{\tilde{M}}(y) = \begin{cases} \frac{y-m_1}{m_1-\underline{m}} & \text{if } m_1 \geq y \geq \underline{m} \\ 1 & \text{if } m_1 \geq y \geq m_2 \\ \frac{\bar{m}-y}{\bar{m}-m_2} & \text{if } \bar{m} \geq y \geq m_2 \\ 0 & \text{if } y < \underline{m} \text{ or } y > \bar{m} \end{cases} \quad (2)$$

Here, the values of m_1 and m_2 are two main possible values, \underline{m} is the least possible value, and \bar{m} is the maximum possible value. We have shown a TrFN in Fig. 2.

Ranking of operations on fuzzy numbers

Ranking is the most significant part in any decision-making in fuzzy optimization problem. The ranking operation of fuzzy number is done by defuzzification technique which transfers the fuzzy set into a real number. This real value is used to find the rank of n number of fuzzy numbers. In the literature, there exists several ranking methods, but the most well-known and efficient defuzzification techniques are center of area, first of maximal, last of maximal, and middle of maximal. In this manuscript, we use the canonical form of TrFNs [37] in our proposed algorithm.

Definition 5 Let $\tilde{N} = \{\underline{n}, n_0, \bar{n}\}$ is a TFN; the graded mean of \tilde{N} is calculated as:

$$P(\tilde{N}) = \frac{1}{6}(\underline{n} + 4n_0 + \bar{n}). \quad (3)$$

Definition 6 Let $\tilde{N} = \{\underline{n}, n_0, \bar{n}\}$ and $\tilde{M} = \{\underline{m}, m_0, \bar{m}\}$ be two TFNs. The graded mean values of two TFNs can be calculated by the following equations:

$$P(\tilde{N}) = \frac{1}{6}(\underline{n} + 4n_0 + \bar{n}) \tag{4}$$

$$P(\tilde{M}) = \frac{1}{6}(\underline{m} + 4m_0 + \bar{m}). \tag{5}$$

The comparison between two number \tilde{N} and \tilde{M} can be described as follows:

- (a) The value of \tilde{N} is equal to the value of \tilde{M} if $P(\tilde{N}) = P(\tilde{M})$.
- (b) The value of \tilde{N} is greater than the value of \tilde{M} if $P(\tilde{N}) > P(\tilde{M})$.
- (c) The value of \tilde{N} is less than the value of \tilde{M} if $P(\tilde{N}) < P(\tilde{M})$.

Definition 7 Let $\tilde{N} = \{\underline{n}, n_0, \bar{n}\}$ and $\tilde{M} = \{\underline{m}, m_0, \bar{m}\}$ be two TFNs. The formula of addition method is described as:

$$P(\tilde{N} \oplus \tilde{M}) = \frac{1}{6}(\underline{n} + 4n_0 + \bar{n}) + \frac{1}{6}(\underline{m} + 4m_0 + \bar{m}). \tag{6}$$

Definition 8 Let $\tilde{N} = \{\underline{n}, n_1, n_2, \bar{n}\}$ be a TrFN. The graded mean value of \tilde{N} can be determined as:

$$P(\tilde{N}) = \frac{1}{6}(\underline{n} + 2n_1 + 2n_2 + \bar{n}). \tag{7}$$

Definition 9 Let $\tilde{N} = \{\underline{n}, n_1, n_2, \bar{n}\}$ and $\tilde{M} = \{\underline{m}, m_1, m_2, \bar{m}\}$ be two TrFNs. The addition \oplus between \tilde{N} and \tilde{M} can be calculated by:

$$P(\tilde{N} \oplus \tilde{M}) = \frac{1}{6}(\underline{n} + 2n_1 + 2n_2 + \bar{n}) + \frac{1}{6}(\underline{m} + 2m_1 + 2m_2 + \bar{m}). \tag{8}$$

Mathematical formulation for FCSP

In this section, we describe the FCSP and a fuzzy mathematical programming method is also introduced to model the FCSP.

Let $G = (X, \tilde{Y})$ be an acyclic discrete directed fuzzy network consisting of n number of vertices symbolized as $X = \{1, 2, \dots, n\}$ and of m number of directed edges $\tilde{Y} \subseteq X \times X$. Each edge is represented by a simple ordered pair (a, b) ; here, $a \neq b$ and $a, b \in X$. We have considered in this network that there exists only one directed edge (a, b) among the two vertices a and b . A path is a consequence of finite of alternating vertices and edges between the two nodes in which the starting vertex of each edge is same as the ending vertex of preceding edge in the sequence. The source vertex of this fuzzy network is s and ending vertex is t . Here, we consider two non negative trapezoidal fuzzy numbers $\tilde{t}_{a,b}$ and $\tilde{c}_{a,b}$ which are associated with the edge (a, b) . Here, $\tilde{T}_{a,b}$ is

used to represent the fuzzy traversal time between the nodes u and v . The another fuzzy number $\tilde{c}_{a,b}$ is used to represent the fuzzy cost between the nodes a and b .

In this FCSP, we have to select a shortest path within the source vertex and other vertices with a specific constraint that the fuzzy traveling time of the selected path does not more than a fuzzy predetermined waiting time \tilde{T}_{delay} . Here, \tilde{T}_{delay} represents the highest possible travel time to transverse a specific path. In another sentence, we can say that the additional constraint demonstrates the a maximum time limit on the sum of fuzzy travel time of edges in the path.

The fuzzy mathematical programming of FCSP is described as follows:

$$\text{Minimize } \sum_{a=1}^n \sum_{b=1}^n \tilde{c}_{ab}x_{ab}$$

subject to:

$$\sum_{a=1}^n x_{ab} - \sum_{a=1}^n x_{ba} = \begin{cases} 1 & \text{if } a = s, \\ 0 & \text{if } a \neq s, \\ -1 & \text{if } a = t, \end{cases} \quad d(a = 1, 2, \dots, n)$$

$$\sum_{a=1}^n \sum_{b=1}^n \tilde{t}_{ab}x_{ab} \leq \tilde{T}_{\text{delay}}$$

$$x_{ij} \in \{0, 1\}, \forall (u, v) \in \tilde{Y}. \tag{9}$$

Here, x_{ab} is variable associated with each edge (a, b) . If edge (a, b) belongs to the set of edges admitted in the fuzzy constrained shortest path, then $x_{ab} = 1$, otherwise $x_{ab} = 0$.

The collection of all possible paths between the vertex s to vertex d is represented by P_{sd} . Let $T(p) = \sum_{u,v \in p} \tilde{t}_{u,v}$ be fuzzy travel time of the path p and $L(p) = \sum_{u,v \in p} \tilde{c}_{u,v}$ be fuzzy traveling cost of the path p . Given $\tilde{T} > 0$ and $P_{sd}(\tilde{T}) = \{p_{sd} : T(p_{sd}) \leq \tilde{T}\}$ is the set of all possible paths between two nodes s and d which satisfies the condition $T(p_{sd}) \neq \tilde{T}$. Each and every path belongs to the set $P_{sd}(\tilde{T})$ is said to be a fuzzy feasible path. In this FCSP, our main objective to select a fuzzy feasible path with minimum fuzzy cost.

Definition 10 Let p_{sd}^* be a fuzzy feasible path. p_{sd}^* is called as fuzzy optimal constrain path in a FCSP if and only if $P(\tilde{L}(p_{sd}^*)) \leq P(\tilde{L}(p_{sd}))$ for any fuzzy feasible path p_{sd} . Here, $\tilde{L}(p_{sd})$ is trapezoidal fuzzy number and the value of $P(\tilde{L}(p_{sd}))$ is calculated using (7).

Let $\tilde{d}_j(\tilde{\omega})$ denotes the fuzzy shortest path length between the source point/node s and destination point/node j subject to a specific criteria that the traversing time of the path does

not more than $\tilde{\omega}$. Let $\tilde{p}_j(\tilde{\omega})$ is the last edge of this path. We consider that $P(\tilde{d}_j(\tilde{\omega})) = \infty$ for $\tilde{\omega} < \tilde{0}$. In this case, the dynamic formulation is given as follows:

$$\begin{aligned}\tilde{d}_j(\tilde{0}) &= \tilde{0} \\ \tilde{d}_j(\tilde{\omega}) &= \min \left\{ \tilde{d}_j(\tilde{\omega} \ominus \tilde{1}), \min_k \left\{ \tilde{d}_k(\tilde{\omega} \ominus \tilde{t}_{kj}) \oplus \tilde{c}_{kj} \right\} \right\}.\end{aligned}$$

Proposed algorithm for FCSP

In Sect. 4, we present an algorithmic approach to determine the fuzzy constrained optimal paths within a source node s and every other nodes in the fuzzy network. The proposed algorithm is presented in Algorithm 1.

Case study

In this section, we present an application of the proposed model on a location-based online cab service.

People visit from one city to another cities due to several purposes, e.g., jobs, business, few for holidays and vacations, education, etc. It is not to take their own a car in each and every cities or countries which they visit. Travelers always choose other alternatives like cabs, trains, public transports, rent cars, buses, and bikes. It is very much popular to take a car on rent due to the traveling cost of a car is very much cheaper than taking a car/cab/taxi for a total day. To solve this transportation problem, some entrepreneurs have developed an app-based car rental service and users can book the car using online car rental system. The affordable mobile app-based cab services have provided the senior citizens,

Algorithm 1 Fuzzy Constrained Shortest Path Algorithm

Input: A fuzzy graph with trapezoidal fuzzy number as the edge weights and the source vertex s .

Output: The fuzzy constrained optimal paths from a source node s to every other vertices.

```

1: Begin
2:  $P(\tilde{d}_j(\tilde{\omega})) = \infty, \forall \tilde{\omega} < \tilde{0}, \forall j$ 
3:  $P(\tilde{d}_s(\tilde{t})) = 0, P(\tilde{d}_j(\tilde{t})) = \infty, \forall 0 \leq P(\tilde{d}_j(\tilde{t})) \leq P(\tilde{d}_j(\tilde{T}), 0)$ 
4:  $S := \{s\}$ 
5:  $U := S$ 
6: while  $S \neq \{\}$  do
7:    $S := \{\}$ 
8:   for  $(u, v) \in E; \exists u \in U$  do
9:     for  $P(\tilde{d}_j(\tilde{t})) \leq P(\tilde{d}_j(\tilde{T} \ominus \tilde{t}_{uv}))$  that  $P(\tilde{d}_u(\tilde{t}) \oplus \tilde{c}_{uv}) <$ 
        $P(\tilde{d}_v(\tilde{t} \ominus \tilde{t}_{uv}))$  do
10:        $\tilde{d}_v(\tilde{t} \ominus \tilde{t}_{uv}) := \tilde{d}_u(\tilde{t}) \oplus \tilde{c}_{uv}$ 
11:        $P(\tilde{d}_v(\tilde{t} \ominus \tilde{t}_{uv})) = P(\tilde{d}_u(\tilde{t}) \oplus \tilde{c}_{uv})$ 
12:        $P_v(\tilde{t} \ominus \tilde{t}_{uv}) = (u, v)$ 
13:        $S := S \cup \{v\}$ 
14:     end for
15:   end for
16: end while
17:  $P = \{\}$ 
18:  $v = \text{Target node}, \tilde{t} = \tilde{T}$ 
19: while  $v \neq s$  do
20:    $(u, v) = P_v(\tilde{t})$ 
21:    $P := P \cup (u, v)$ 
22:    $\tilde{t} := \tilde{t} \ominus \tilde{t}_{uv}$ 
23:    $v := u$ 
24: end while

```

working women, students, and the party goes to visit alone to the place at any time of the day. For this reason, recently, it becomes an efficient tool when a person wants to hire a cab. These mobile app-based cab services were first time established in the UK, but it has now very much popular all over the world. It becomes very much comfortable to find a cab in the minimum possible waiting time rather of waiting to find a cab on a busy road. Each app-based cab service has always their own advantages and they provide several services to the user. Each user needs a mobile app to determine the proper location of the traveler.

Uber is the most popular and largest mobile app-based cab service in India. It is very simple for citizenry to arrive to their possible destination place easily and safely with Uber cab wherever users are in a specific city or even send their parcels. To find a cab for ride via Uber, the user has to choose the source location and the destination location. Then, the passenger watches the traveling cost beforehand, and finally, he finds the traveling time in the mobile app. Their mobile app system connects with the specific passenger to the closest Uber driver and it sends the a message of passenger’s location to the driver. The Uber driver can accept or even refuse the request of the passenger. If he accepts the request of the passenger, the driver will must pick up that passenger.

In this manuscript, we have examined our proposed model in the Uber cab service system in China. Here, we have used the Fig. 3 to demonstrate the proposed algorithmic technique for Uber. The main objective of this case study is to find the FCSP for the online cabs. This graph describes a network among several locations of China. This graph consists of 23 vertices and lines (arcs). Table 1 shows the fuzzy cost value and the fuzzy travel time from each line. This table includes the triangular fuzzy numbers which have been figured out based on the average daily traffic.

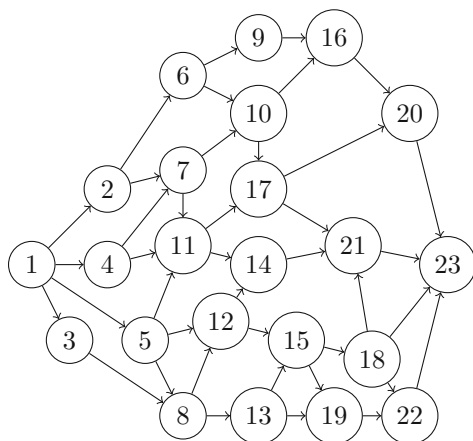


Fig. 3 A network

Table 1 Fuzzy traveling costs and fuzzy traveling time of the network

Arc	Fuzzy cost	Fuzzy time
(1,2)	(200, 220, 230, 280)	(11,13,15,18)
(1,3)	(130, 150, 160, 170)	(10,11,13,14)
(1,4)	(300, 350, 360, 390)	(7,10,12,14)
(1,5)	(270, 290, 300, 330)	(6,8,9,11)
(2,6)	(300, 360, 370, 380)	(4,10,15,16)
(2,7)	(190, 200, 220, 250)	(6,11,11,13)
(3,8)	(100, 130, 150, 160)	(10,11,16,17)
(4,7)	(150, 170, 180, 190)	(17,20,22,24)
(4,11)	(230, 250, 260, 270)	(6,11,12,14)
(5,8)	(310, 320, 330, 350)	(7,9,11,14)
(5,11)	(320, 340, 360, 380)	(5,10,13,16)
(5,12)	(90, 100, 105, 130)	(9,13,15,18)
(6,9)	(120, 130, 150, 170)	(5,8,10,12)
(6,10)	(230, 260, 300, 320)	(9,10,15,16)
(7,10)	(100, 105, 110, 120)	(7,10,12,15)
(7,11)	(70, 80, 90, 100)	(4,7,8,11)
(8,12)	(50, 70, 80, 90)	(4,8,9,11)
(8,13)	(80, 90, 100, 120)	(4,5,8,9)
(9,16)	(160, 170, 180, 190)	(5,7,9,11)
(10,16)	(100, 110, 130, 140)	(6,9,10,13)
(10,17)	(200, 205, 215, 220)	(13,19,20,23)
(11,14)	(290, 300, 310, 330)	(6,9,11,15)
(11,17)	(250, 270, 280, 300)	(5,9,11,14)
(12,14)	(300, 310, 320, 330)	(10,14,16,21)
(12,15)	(320, 350, 360, 370)	(10,14,15,18)
(13,15)	(200, 230, 250, 270)	(8,12,14,17)
(13,19)	(300, 350, 360, 370)	(15,18,19,22)
(14,21)	(235, 245, 260, 275)	(10,12,13,15)
(15,18)	(100, 120, 160, 180)	(6,9,11,15)
(15,19)	(120, 150, 160, 170)	(6,7,10,11)
(16,20)	(200, 230, 240, 250)	(4, 7, 9, 11)
(17,20)	(330, 340, 350, 360)	(5,10,11,16)
(17,21)	(200, 240, 250, 260)	(5,7,8,11)
(18,21)	(150, 160, 180, 190)	(13,17,18,21)
(18,22)	(170, 180, 190, 200)	(4,5,7,8)
(18,23)	(230, 250, 260, 280)	(4,7,9,12)
(19,22)	(200, 220, 230, 250)	(14,16,17,20)
(20,23)	(210, 250, 260, 270)	(11,14,16,19)
(21,23)	(220, 230, 260, 280)	(11,15,17,19)
(22,23)	(230, 250, 260, 290)	(3,5,6,9)

Algorithm 1 is used to determine the fuzzy constrained shortest path from source vertex 1 to several destination vertices. In Table 2, we have shown the values of fuzzy traveling cost and the fuzzy traveling time of the FCSP from the source vertex 1 to several target vertices. The shortest path between the source vertex 1 and the target vertex 23 path is $1 \rightarrow 5 \rightarrow 11 \rightarrow 17 \rightarrow 21 \rightarrow 23$, as long as the maximum waiting time is [33, 52, 60, 70]. The fuzzy traveling cost of this path is [1260, 1370, 1450, 1610]. If the maximum waiting time allowed is decreased to [32, 49, 56, 74], then there exists no path between vertex 1 and vertex 23 meeting this additional criterion of the FCSP. The fuzzy shortest path from the vertex 1 to vertex 18 is $1 \rightarrow 5 \rightarrow 12 \rightarrow 15 \rightarrow 18$ if the maximum waiting time allowed is [34, 45, 55, 67] and the fuzzy traveling cost of this specific path is [840, 860, 925, 1070]. In this path, if the maximum waiting time permitted is to increase to [40, 50, 65, 70], the fuzzy shortest path will be $1 \rightarrow 3 \rightarrow 8 \rightarrow 13 \rightarrow 15 \rightarrow 18$ with fuzzy cost [510, 720, 820, 900]. There exists no possible path from the vertex 1 to vertex 18 if the maximum waiting time is reduced to [31, 42, 48, 64]. We have shown the FCSP from the starting vertex 1 to several other destination vertices.

In this manuscript, we have also solved this FCSP using fuzzy linear programming approach. The fuzzy mathematical model, the FCSP, is described in (9). The LINGO software is used to solve the fuzzy mathematical model, the FCSP. We have shown the results of our proposed method and fuzzy

linear programming approach in Table 3. In Table 3, the calculated shortest paths between the source vertex and all the other target vertices using proposed algorithm are exactly the same to the results of the fuzzy linear programming approach. However, our proposed algorithm needs less time than fuzzy Dijkstra’s algorithm and fuzzy linear programming approach. It is shown in Table 3.

Conclusion

The FSPP is one of most popular combinatorial optimization problems in the area of fuzzy operation research. There exists many different types of FSPPs. CSPP is a special type of SPP which determines the shortest paths between a predetermined starting point and all other target points in a directed graph under an additional constraint of being not more than the sum of traveling time for the shortest path. Two import transportation parameters (traveling time and traveling cost) are related to each and every arc in a CSPP. It is not always possible to predicate the exact traveling cost and traveling time of a path due to the fluctuation of weather conditions and traffic frequency of the path. The fuzzy set is an useful method to deal those uncertainties in the information. In this manuscript, we investigate the constrained shortest path problem, where edge weights are presented by trapezoidal fuzzy numbers. We define this type of FSPP as FCSP. The significance of

Table 2 Solutions of our proposed algorithm

Source to destination	Optimal path	Max delay time	Fuzzy traveling time	Fuzzy traveling cost
1-23	$1 \rightarrow 5 \rightarrow 11 \rightarrow 17 \rightarrow 21 \rightarrow 23$	[33, 52, 60, 70]	[33, 49, 58, 71]	[1260, 1370, 1450, 1610]
1-23	No possible path	[31, 48, 55, 76]		
1-18	$1 \rightarrow 3 \rightarrow 8 \rightarrow 13 \rightarrow 15 \rightarrow 18$	[40, 50, 65, 75]	[38, 48, 62, 72]	[510, 720, 820, 900]
1-18	$1 \rightarrow 5 \rightarrow 12 \rightarrow 15 \rightarrow 18$	[34, 45, 55, 67]	[34, 45, 55, 67]	[840, 860 925, 1070]
1-18	No possible path	[31, 48, 55, 76]		
1-20	$1 \rightarrow 4 \rightarrow 7 \rightarrow 10 \rightarrow 16 \rightarrow 20$	[39, 58, 68, 78]	[41, 56, 65, 77]	[850, 975, 1032, 1100]
1-20	$1 \rightarrow 5 \rightarrow 11 \rightarrow 17 \rightarrow 20$	[20, 39, 48, 55]	[21, 37, 44, 57]	[990, 105, 1100, 1170]
1-20	No possible path	[21, 30, 40, 50]		
1-17	$1 \rightarrow 4 \rightarrow 11 \rightarrow 17$	[20, 33, 37, 44]	[18, 30, 35, 42]	[780, 870, 900, 960]
1-17	$1 \rightarrow 5 \rightarrow 11 \rightarrow 17$	[18, 26, 34, 42]	[16, 27, 33, 41]	[840, 900, 940, 1010]
1-17	No possible path	[15, 24, 30, 36]		
1-13	$1 \rightarrow 3 \rightarrow 8 \rightarrow 13$	[25, 30, 40, 42]	[24, 27, 37, 40]	[320, 370, 410, 450]
1-13	$1 \rightarrow 5 \rightarrow 8 \rightarrow 13$	[18, 24, 29, 35]	[17, 22, 28, 34]	[660, 700, 730, 800]
1-13	No possible path	[16, 20, 25, 36]		

Table 3 Comparison of our proposed algorithm with fuzzy linear programming

Source to destination	Solution of our proposed algorithm	Solution of LINGO	Max possible delay	Total fuzzy traveling time (path)	Total cost (path)	Run time	Run time LINGO
1-23	1 → 5 → 11 → 17 → 21 → 23	$x_{1,5} = 1,$ $x_{5,11} = 1, x_{1,5} = 1,$ $x_{5,11} = 1$	[33, 52, 60, 70]	[33, 49, 58, 71]	[1260, 1370, 1450, 1610]	0.170	0.37
1-18	1 → 3 → 8 → 13 → 15 → 18	$x_{1,3} = 1, x_{3,8} = 1$ $x_{13,15} = 1,$ $x_{15,18} = 1$	[40, 50, 65, 75]	[38, 48, 62, 72]	[510, 720, 820, 900]	0.19	0.57
1-18	1 → 5 → 12 → 15 → 18	$x_{1,5} = 1, x_{12,15} = 1$ $x_{15,18} = 1,$	[34, 45, 55, 67]	[34, 45, 55, 67]	[840, 860 925, 1070]	0.28	0.37
1-20	1 → 4 → 7 → 10 → 16 → 20	$x_{1,4} = 1, x_{4,7} = 1$ $x_{10,16} = 1,$ $x_{16,20} = 1$	[39, 58, 68, 78]	[41, 56, 65, 77]	[850, 975, 1032, 1100]	0.39	0.59
1-20	1 → 5 → 11 → 17 → 20	$x_{1,5} = 1, x_{5,11} = 1$ $x_{17,20} = 1$	[20, 39, 48, 55]	[21, 37, 44, 57]	[990, 105, 1100, 1170]	0.19	0.27
1-17	1 → 4 → 11 → 17	$x_{1,4} = 1, x_{11,17} = 1$	[20, 33, 37, 44]	[18, 30, 35, 42]	[780, 870, 900, 960]	0.19	0.34
1-13	1 → 3 → 8 → 13	$x_{1,3} = 1, x_{8,13} = 1$	[25, 30, 40, 42]	[24, 27, 37, 40]	[320, 370, 410, 450]	0.29	0.54

using trapezoidal fuzzy numbers as edge weights in FCSPSP is described in this study. This paper introduces a mathematical model for a constrained shortest path problem with fuzzy edge weights. The LINGO software is used to solve this problem. Here, we present an algorithmic technique for solving this problem. In our proposed method, we incorporate the uncertainties in dynamic programming algorithm for FCSPSP using trapezoidal fuzzy number as edge weight. Finally, we describe a real-life application of our proposed model on an app-based taxi service in China. In future, the proposed algorithm can be utilized to real-life problems in supply chain management, transportation, and other relevant fields.

Acknowledgements Project supported by the National Natural Science Foundation of China, Research on Precision Evaluation Model of Goaf Pressure Relief Gas Drainage Based on LSTM Regression Number: 51804248, Science and Technology Project of State Grid Zizang Electric Power Co., Ltd (SGXZJY00JHJS2000008), and Research Technology Service of Multi Energy Complementary Demonstration Application

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Handler GY, Zang I (1980) A dual algorithm for the constrained shortest path problem. *Networks* 10(4):293–309
2. Beasley JE, Christofides N (1989) An algorithm for the resource constrained shortest path problem. *Networks* 19(4):379–394
3. Matthew Carlyle W, Royset JO, Kevin Wood R (2008) Lagrangian relaxation and enumeration for solving constrained shortest-path problems. *Networks* 52(4):256–270
4. Santos L, Coutinho-Rodrigues J, Current JR (2007) An improved solution algorithm for the constrained shortest path problem. *Transp Res Part B Methodol* 41(7):756–771
5. Dubois D, Parde H (1980) *Fuzzy sets and systems: theory and applications*, vol 144. Academic, New York
6. Klein CM (1991) Fuzzy shortest paths. *Fuzzy Sets Syst* 39(1):27–41
7. Hassanzadeh R, Mahdavi I, Mahdavi-Amiri N, Tajdin A (2013) A genetic algorithm for solving fuzzy shortest path problems with mixed fuzzy arc lengths. *Math Comput Model* 57(1):84–99
8. Dey A, Pradhan R, Pal A, Pal T (2015) The fuzzy robust graph coloring problem. In: *Proceedings of the 3rd international conference on frontiers of intelligent computing: theory and applications (FICTA) 2014*. Springer, pp 805–813
9. Chanas S, Kamburowski J (1983) The fuzzy shortest route problem. In: *Interval and fuzzy mathematics, Proc. Polish Symp, Technical University of Poznan, Poznan*, pp 35–41
10. Blue M, Bush B, Puckett J (2002) Unified approach to fuzzy graph problems. *Fuzzy Sets Syst* 125(3):355–368
11. Okada S (2004) Fuzzy shortest path problems incorporating inter-activity among paths. *Fuzzy Sets Syst* 142(3):335–357
12. Takahashi MT, Yamakami A (2005) On fuzzy shortest path problems with fuzzy parameters: an algorithmic approach. In: *Fuzzy information processing society, 2005. NAFIPS 2005. Annual meeting of the North American. IEEE*, pp 654–657
13. Mahdavi I, Nourifar R, Heidarzade A, Amiri NM (2009) A dynamic programming approach for finding shortest chains in a fuzzy network. *Appl Soft Comput* 9(2):503–511
14. Yager RR (1986) Paths of least resistance in possibilistic production systems. *Fuzzy Sets Syst* 19(2):121–132
15. Lin K-C, Chern M-S (1993) The fuzzy shortest path problem and its most vital arcs. *Fuzzy Sets Syst* 58(3):343–353
16. Okada S, Soper T (2000) A shortest path problem on a network with fuzzy arc lengths. *Fuzzy Sets Syst* 109(1):129–140
17. Md Abu Nayeem S, Pal M (2005) Shortest path problem on a network with imprecise edge weight. *Fuzzy Optim Decis Mak* 4(4):293–312
18. Moazeni S (2006) Fuzzy shortest path problem with finite fuzzy quantities. *Appl Math Comput* 183(1):160–169
19. Hernandez F, Lamata MT, Verdegaj JL, Yamakami A (2007) The shortest path problem on networks with fuzzy parameters. *Fuzzy Sets Syst* 158(14):1561–1570
20. Deng Y, Chen Y, Zhang Y, Mahadevan S (2012) Fuzzy dijkstra algorithm for shortest path problem under uncertain environment. *Appl Soft Comput* 12(3):1231–1237
21. Sheng G, Yuliang S, Wang W (2019) A new fractal approach for describing induced-fracture porosity/permeability/compressibility in stimulated unconventional reservoirs. *J Petrol Sci Eng* 179:855–66
22. Dey A, Pal A, Pal T (2016) Interval type 2 fuzzy set in fuzzy shortest path problem. *Mathematics* 4(4):62
23. Dey A, Pradhan R, Pal A, Pal T (2018) A genetic algorithm for solving fuzzy shortest path problems with interval type-2 fuzzy arc lengths. *Malays J Comput Sci* 31(4):255–270
24. Dey A, Pal A (2019) Computing the shortest path with words. *Int J Adv Intell Paradig* 12(3–4):355–369
25. Yang Y, Junhua H, Liu Y, Chen X (2019) Alternative selection of end-of-life vehicle management in China: a group decision-making approach based on picture hesitant fuzzy measurements. *J Clean Prod* 206:631–645
26. Dey A, Pal A, Long HV et al (2019) Fuzzy minimum spanning tree with interval type 2 fuzzy arc length: formulation and a new genetic algorithm. *Soft Comput* 1–12
27. Wang R, Wang J, Gao H, Wei G (2019) Methods for madm with picture fuzzy muirhead mean operators and their application for evaluating the financial investment risk. *Symmetry* 11(1):6
28. Hansen P (1980) Bicriterion path problems. In: *Multiple criteria decision making theory and application*. Springer, pp 109–127
29. Sengupta A, Pal TK (2000) On comparing interval numbers. *Eur J Oper Res* 127(1):28–43
30. Zhao H, Lingfei X, Guo Z, Liu W, Zhang Q, Ning X, Li G, Shi L (2019) A new and fast waterflooding optimization workflow based on insim-derived injection efficiency with a field application. *J Petrol Sci Eng* 179:1186–1200
31. Dijkstra EW (1959) A note on two problems in connexion with graphs. *Numer Math* 1(1):269–271
32. Grönkvist M (2006) Accelerating column generation for aircraft scheduling using constraint propagation. *Comput Oper Res* 33(10):2918–2934
33. Weide O, Ryan D, Ehgott M (2010) An iterative approach to robust and integrated aircraft routing and crew scheduling. *Comput Oper Res* 37(5):833–844

34. Almasi MH, Mounes SM, Koting S, Karim MR (2014) Analysis of feeder bus network design and scheduling problems. *Sci World J* 2014
35. Xue G (2003) Minimum-cost qos multicast and unicast routing in communication networks. *IEEE Trans Commun* 51(5):817–824
36. Sivakumar B, Bhalaji N, Sivakumar D (2014) A survey on investigating the need for intelligent power-aware load balanced routing protocols for handling critical links in manets. *Sci World J* 2014
37. Chou C-C (2003) The canonical representation of multiplication operation on triangular fuzzy numbers. *Comput Math Appl* 45(10):1601–1610

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.