

Routing Planning As An Application Of Graph Theory

Prof Boominathan P, Kanchan Arora

ABSTRACT:- This paper presents a routing algorithm that uses fuzzy logic technique to find the shortest routing path. The basic idea behind path finding is searching a graph, starting at one point, and exploring adjacent nodes from there until the destination node is reached. Generally, the goal is of course to obtain the shortest route to the destination. The proposed Fuzzy Routing Algorithm (FRA) modifies the well-known Dijkstra's Single-source shortest path algorithm by using fuzzy-logic membership functions in the path-cost update process. The main objective of FRA is to reduce path-request blocking and increase overall utilization. The fuzzy weighted graphs, along with generalizations of algorithms for finding optimal paths within them, have emerged as an adequate modeling tool for prohibitively complex and/or inherently imprecise systems. These algorithms are reviewed and formalized with uncertainty which comes from weights on edges according to actual situation on the road such as weather conditions, and road capacity at the specified time. The two key issues need to be addressed in SPP (Shortest Path Algorithm) with fuzzy parameters are to determine the addition of two edges and to compare the distance between two different paths with their edge lengths represented by fuzzy numbers.

Keywords: FuzzyLogic, PathFinding, Weighted Graph, Path-RequestBlocking, Fuzzy RoutingAlgorithm, NearestNeighbor Algorithm, DijkstraAlgorithm.

2. INTRODUCTION:-

In many applications such as transportation, routing, communications, economical, and so on, graphs emerge naturally as a mathematical model of the observed real world system. Fuzzy logic is a form of many-valued logic or probabilistic logic; it deals with reasoning that is approximate rather than fixed and exact. Compared to traditional binary sets (where variables may take on true or false values) fuzzy logic variables may have a truth value that ranges in degree between 0 and 1. Fuzzy logic has been extended to handle the concept of partial truth, where the truth value may range between completely true and completely false. Dijkstra's algorithm (named after its discover, E.W. Dijkstra) solves the problem of finding the shortest path from a point in a graph (the source) to a destination. It turns out that one can find the shortest paths from a given source to all points in a graph in the same time, hence this problem is sometimes called the single-source shortest paths problem. Dijkstra's algorithm keeps two sets of vertices:

S:--The set of vertices whose shortest paths from the source have already been determined and

V:--S the remaining vertices.

The basic mode of operation is: Initialise d and p_i , Set S to empty, While there are still vertices in $V-S$, Sort the vertices in $V-S$ according to the current best estimate of their distance from the source, Add u , the closest vertex in $V-S$, to S , Relax all the vertices still in $V-S$ connected to u . 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], [14], [15], [16], [17], [18], [19], [20], [21] and [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 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 transforms the trapezoidal

number into crisp number by the defuzzification function, also called Yager's ranking index. 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. This method is widely used in many applications such as multi-criteria decision making, risk evaluation, portfolio selection, evaluation of enterprise knowledge management capabilities, product adoption, evaluation of airline service quality and efficient network selection in heterogeneous wireless networks. 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 in a easy and straight manner. The paper is organized as follows. Section 2 gives a brief introduction to 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.

3. IMPLEMENTATION

3.1 Path finding algorithms

A path finding algorithm for transit network is proposed to handle the special characteristics of transit networks such as city emergency handling and drive guiding system, in where the optimal paths have to be found. As the traffic condition among a city changes from time to time and there are usually a huge amounts of requests occur at any moment, it needs to quickly find the best path. Therefore, the efficiency of the algorithm is very important. The algorithm takes into account the overall level of services and service schedule on a route to determine the shortest

path and transfer points. There are several methods for pathfinding: In Dijkstra's algorithm the input of the algorithm consists of a weighted directed graph G and a source vertex in Graph. Let's denote the set of all vertices in the graph G as V . Each edge of the graph is an ordered pair of vertices (u, v) representing a connection from vertex u to vertex v . The set of all edges is denoted E . Weights of edges are given by a weight function $w: E \rightarrow [0, \infty]$; therefore $w(u, v)$ is the non-negative cost of moving from vertex u to vertex v . The cost of an edge can be thought of as the distance between those two vertices. The cost of a path between two vertices is the sum of costs of the edges in that path. For a given pair of vertices s and t in V , the algorithm finds the path from s to t with lowest cost (i.e. the shortest path). It can also be used for finding costs of shortest paths from a single vertex s to all other vertices in the graph.

3.2 The Proposed Method

A connected network with given arcs and nodes in which s is the source node and e is the end (sink) node is considered. The problem is to find the shortest path between s and e with respect to the cost (or time or distance etc.) parameter related to each arc. This parameter is considered to be in terms of IFNs $c_{ij} = (\mu_{ij}, \nu_{ij})$ where μ_{ij} represents the degree of acceptance that the arc $i - j$ will be included in the shortest path with respect to the cost for traveling along the arc $i - j$. Similarly ν_{ij} represents the degree of rejection that the arc $i - j$ will be included in the shortest path with respect to the cost for traveling along the arc $i - j$. From the IFNs, the IF value for each arc of the network is considered in the form of $[tA(x), 1 - fA(x)]$ described earlier. The weight vector $\omega = \omega_{12}, \omega_{13}, \dots, \omega_{23}, \omega_{24}, \dots, \omega_{ij}, \dots, \omega_{nT}$, where each ω_{ij} related to arc $i - j$ is considered as the opinion of the expert regarding the IF cost c_{ij} required for traveling along the arc $i - j$, $0 \leq \omega_j \leq 1$. The Modified Intuitionistic Fuzzy Dijkstra's Algorithm (MIFDA)

Input: Let $G = (V, E)$ be a simple weighted network with intuitionistic fuzzy parameters for the arcs. ' s ' is the starting point and ' e ' is the terminal point.

Output: (a) Weighted aggregated IFV of the minimum-cost path or the shortest path w.r.t. the total intuitionistic fuzzy cost for traveling through the shortest path.

(b) the shortest path.

Let $L(x)$ denote the label of the node ' x ' which represents the weighted aggregated IFV for the path from the node ' s ' to the node ' x '. The weight vector

$\omega = \omega_{12}, \omega_{13}, \dots, \omega_{23}, \omega_{24}, \dots, \omega_{ij}, \dots, \omega_{nT}$, where each ω_{ij} related to arc $i - j$ is

considered as the opinion of the expert regarding the IF cost

c_{ij} required for traveling along the arc $i - j$, $0 \leq \omega_j \leq 1$.

Step1: Let $P = \emptyset$, where P is the set of those nodes which have permanent labels and $T = \{\text{all nodes of the network}$

$G\}$. At first, the permanent label to ' s ' has been assigned as $L(s) = (1, 0)$, (initially), ' s ' is the starting node, so definitely it will be present in the shortest path. This is represented by the IFN $(1, 0)$ where 1 represents the degree of acceptance and 0 represents the degree of rejection of the fact that node ' s ' is in the shortest path. Also, $L(x) = (0, 1) \forall x \in T$ and $x \neq s$

Step2: That node ' v ' in T is selected which has the highest score value of its label, called the permanent label of ' v ' (i.e., $L(v)$). Then $P = P \cup \{v\}$ and $T = T - (v)$. Again the node in T with highest score value of its label is selected. The new label of a node ' x ' in T is given by $L(x) = \max \{\text{old } L(x), \text{IFHG}(L(v), c(v, x))\}$, (11) where $c(v, x)$ is the IF cost for traveling along the arc $v - x$. The associated weights $w_j > 0$ of the IFHG operator are evaluated using Eqs. 8 and 9 where

$$n_j = 1$$

$$w_j = 1.$$

Then using the IFHG operator of the Eq. 10 of Section 2.5 described in this paper, the aggregated value of $L(v)$ and $c(v, x)$ are derived which are also in terms of intuitionistic fuzzy value. The max function is used for evaluating the maximum of the two IFNs using the Eqs. 3, 5 and 6. If there is no difference between two scores, then the accuracy degrees are calculated by using Eq. 4. It has been assumed that $c(v, x) = (0, 1)$, if there is no edge joining the node ' v ' directly to the node ' x '.

Step3: STOP. The process of finding the nodes with permanent label is repeated until ' e ' gets a permanent label. The above steps do not actually list the shortest path from the starting node to the terminal node; it only gives the weighted aggregated IFV of the IF cost of traveling the shortest path giving its degree of acceptance and the degree of rejection for the shortest path.

Step4: The shortest path can be easily constructed by working backwards from the terminal node such that one moves to that predecessor from whom the current node has got its permanent label.

Step5: End In Step 2 for calculating the IFHG $(L(v), c(v, x))$, one has to proceed as follows: ω_j = the weight to be considered as the opinion of the expert regarding the IF cost c_{ij} required for traveling along the arc $i - j$. The normalized weight vectors for evaluating IFHG $(L(v), c(v, x))$ are calculated whenever required.

Considering the IF values

$$\begin{aligned} _ap &= _t \\ _ap, 1 - f & \\ _ap_ &\text{ of the } p = 1, 2, \dots, q \text{ arcs, the} \\ &\text{weighted IF values} \end{aligned}$$

$$_a.p, p = 1, 2, \dots, q \text{ for the } q \text{ arcs are calculated using Eq.7 as } _ap = _tqx_ _wjp_ _ap, \quad 1 - f_apqx_ _wjp_ \quad (12)$$

Now, by using the ranking method (utilizing the scores S_a

p
by using

Eq. 3) of the IFVs described in Section 2.3, the p th largest of the weighted IFVs

$\sigma(p)$. (If there is no difference between two scores, then it is required to calculate the accuracy degrees $H_{\sigma(p)}$ by using Eq. 4. After that the alternatives $\sigma(p)$ are ranked in accordance with the accuracy degrees.) Then using the IFHG operator the weighted aggregated IFV for $(L(v), c(v, x))$ are derived in the form of $[t_j, 1 - f_j]$ and hence the corresponding weighted aggregated IFN $r_j = [t_j, f_j]$ for IFHG $(L(v), c(v, x))$ are obtained which can be used for evaluating $L(x)$.

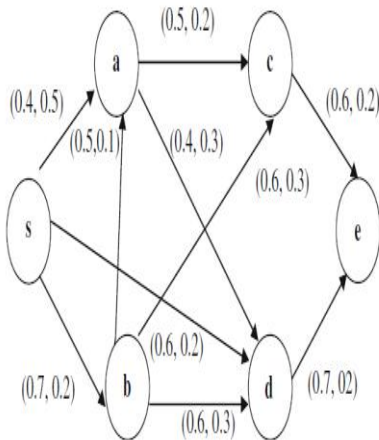


Fig. 1 Network with IF costs

Table 1 Data for IF Costs for traveling along the respective arcs of the given network

Arcs (i - j)	Intuitionistic Fuzzy Costs along these arcs $c(i, j)$	Weights (w_j)
sa	(0.4, 0.5)	0.3
sb	(0.7, 0.2)	0.6
bd	(0.6, 0.3)	0.6
de	(0.7, 0.2)	0.8
ac	(0.5, 0.2)	0.5
ce	(0.6, 0.2)	0.6
sd	(0.6, 0.2)	0.6
ba	(0.5, 0.1)	0.4
ad	(0.4, 0.3)	0.4
bc	(0.6, 0.3)	0.5

4. APPLICATION

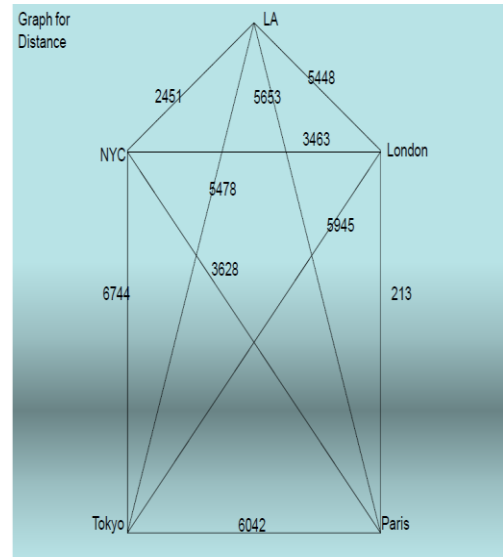
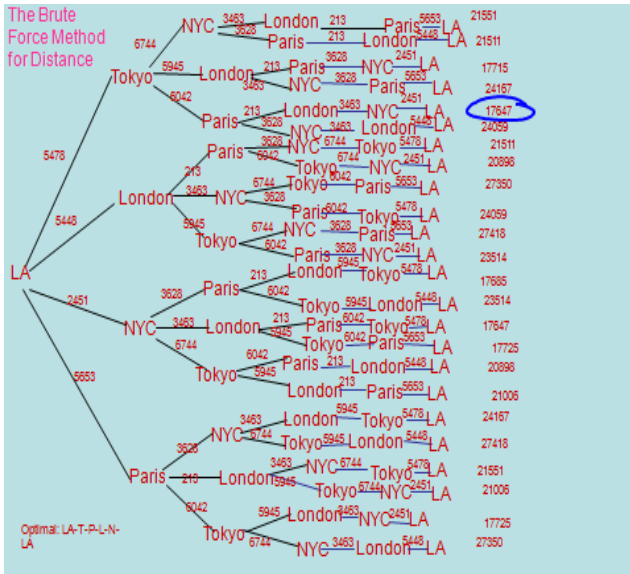
Practical application of routing through graph theory is route planning. Following is the research example of route planning: The problem is how efficiently we can use Graph Theory in route planning for a convenient trip. The question is of two types; whether we want shortest route for the trip or we want to have cheapest route (that will take minimum cost) to travel to five famous cities around the world: Paris, London, New York City, Tokyo and Los Angeles.

Method:

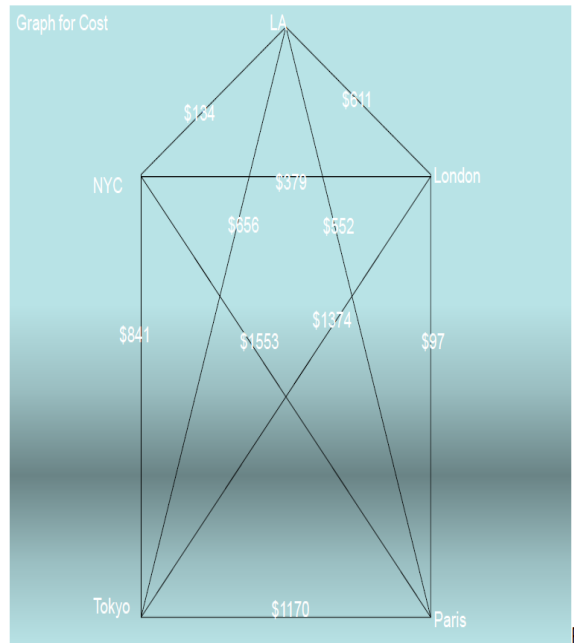
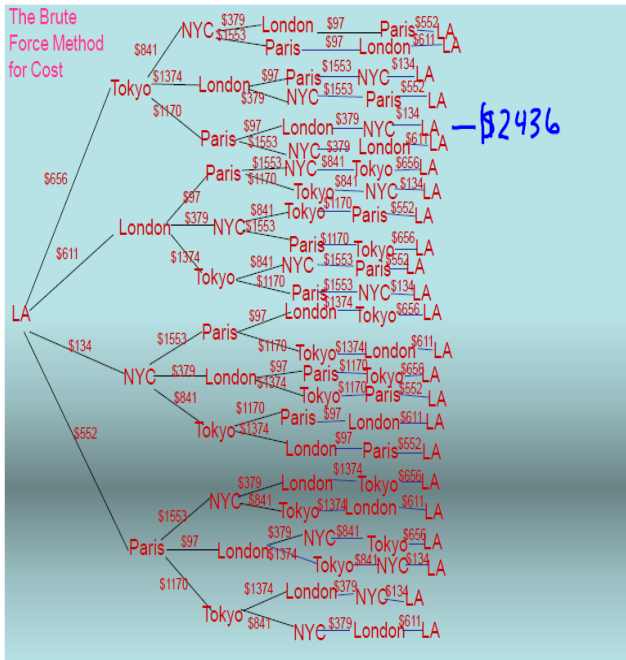
1. Find out the distances and cost to travel for all the cities with respect to each other. For this we can refer travel agencies or internet search engines. Here we are taking a research example where the distance and cost was given in a research.
2. Draw the graphs for distance and price between each city using the algorithms: Nearest neighbour algorithm.
3. The results can be compared by lowest price or least distance, whichever is required.

5. Procedure:

The price is taken from the website of a travel agency and the distance is taken from the internet. The information was maintained into a table. For Demonstration purposes, this set of information was applied to the Nearest Neighbor Algorithm. Since the Brute Force Method organizes the set of data in a way that is applicable for the purpose of route planning (in the case of this project), we chose the Brute Force Method as our primary procedure. The right side of the graph containing the outer “branches” show the five cities listed in every combination possible for the purposes of establishing a comparison between the results of each individual combination.

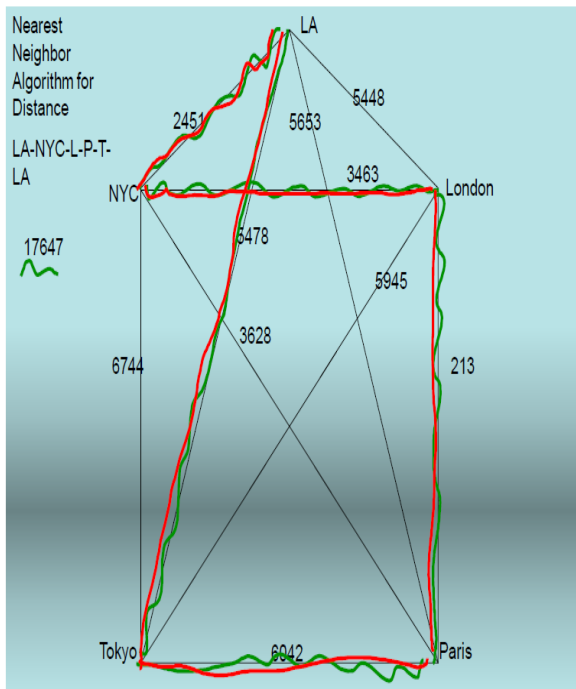


Route with the greatest distance is indicated in blue.

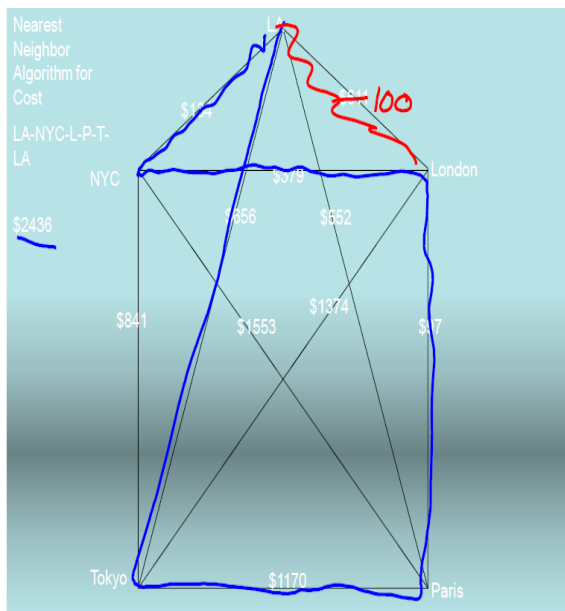


Implementing Nearest Neighbour Algorithm for minimum distance:

We next organized the data in the branched format as specified by the Brute Force Algorithm. In the case of this project, the combination with the lowest figure is the optimal solution.



Implementing Nearest Neighbour Algorithm for minimum cost:



7.Result:

After completing the four types of graph, I found that the route (LA-Tokyo-Paris-London-NYC-LA) was the route representing the shortest distance of 17647 miles and the route (LA-Tokyo-Paris-London-NYC-LA) had the lowest price of \$2,436.

6.Conclusion

This paper solved the problem of shortest path problem with Fuzzy arc length resulting in extending the Dijkstra algorithm. Two important issues are addressed by the paper. Firstly it tells how to determine the addition of two edges. Secondly it gives us the way to compare the

distance between two different paths when fuzzy numbers represent their edges length. The method which is proposed under fuzzy arc lengths to find the shortest path is based on the graded mean integration representation of fuzzy numbers. A numerical example was used to illustrate the efficiency of the proposed method. As far as real applications are concerned in the field of transportation systems, logistics management, and many other network optimization problem that can be formulated as shortest path problem this method can be applied. Through an example this Paper also tried to simplify the application of routing planning in our general life which basically gave the shortest path between some important cities of the world. At the same time, the uncertainty in the shortest path is not limited to the geometric distance this is also shown by this paper. For example, even if the geometric distance is fixed due to the weather and other unexpected factors, the travel time from one city to another city may be represented as a fuzzy variable.

8.References

- [1]. 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.
- [2]. H.W. Liu, New similarity measures between intuitionistic fuzzy sets and between elements, *Mathematical and Computer Modelling* 42 (2005) 61–70.
- [3]. J. Ye, Cosine similarity measures for intuitionistic fuzzy sets and their applications, *Mathematical and Computer Modelling* 53 (2011) 91–97.
- [4]. Y. Deng, Plant location selection based on fuzzy topsis, *International Journal of Advanced Manufacturing Technology* 28 (2006) 839–844.
- [5]. 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.
- [6]. 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.
- [7]. X. Lu, M. Camitz, Finding the shortest paths by node combination, *Applied Mathematics and Computation* 217 (2011) 6401–6408.
- [8]. 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.
- [9]. 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.

- [10]. 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.
- [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]. C. Lin, M.S. Chern, The fuzzy shortest path problem and its most vital arcs, *Fuzzy Sets and Systems* 58 (1993) 343–353.