Fuzzy Shortest Route Algorithm for Telephone Line Connection

Dr. G. Nirmala\(^1\) & K. Uma\(^2\)

\(^1\) Department, Institute Name Associate Professor in Mathematics, K.N.G.A. College (Women), Thanjavur, India
\(^2\) Assistant Professor in Mathematics, A.V.V.M. Sri Pushpam College, Poondi, Thanjavur Dist., India

Abstract- In computer science, there are many algorithms that find a minimum spanning tree for a connected weighted undirected fuzzy graph. This means it finds a subset of the edge-set that forms a fuzzy spanning tree that includes every vertex, where the total weight of all the edges in the fuzzy tree is minimized. Generally, algorithms have many applications, such as landline (intercom) phone connection to a particular college in order to minimize the shortest distance, which applies Nearest Neighbor Algorithm, Sorted Edges Algorithm, Kruskal’s Algorithm to a randomly weighted fuzzy graph. In this paper, we find the solution for the problem, that A.V.V.M. Sri Pushpam College, (Autonomous), Poondi, Thanjavur District, South India need to connect updated intercom lines connecting all the departments using fuzzy shortest route algorithm. The problem is to minimize the amount of new line using Kruskal’s Algorithm with fuzzy graph. These algorithms continuously increase the size of a fuzzy tree, one edge at a time, starting with a fuzzy tree consisting of a single vertex, until it spans all vertices of the fuzzy graph.

Index Terms- Fuzzy spanning tree –Nearest Neighbor Algorithm- Minimum weight link Algorithm-Kruskal’s Algorithm.

I. INTRODUCTION

In many transportation, routing, communications, economical, and other applications, fuzzy graphs emerge naturally as a mathematical model of the observed real world system. Indeed, many problems can be reformulated as a quest for a path (between two nodes in a fuzzy graph) which is optimal in the sense of a number of preset criteria. Very often, these optimality criteria are evaluated in terms of weights, that is, vectors of real numbers, associated with the links of the fuzzy graph. Numerous algorithms have been developed to ease this and related quests. More recently, 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. We review and formalize these algorithms, paying special attention to the ranking methods used for path comparison. We discussed which criteria must be met for algorithm correctness and present an efficient method, based on defuzzification of fuzzy weights, for finding optimal paths.

The network is a standard application pattern for phone lines to connect them up with each other; with minimum distance and company with different charges for different connection using set of lines that connects with a minimum total cost. It should be a fuzzy spanning tree, since if a network isn’t a fuzzy tree.

A less obvious application is that the minimum fuzzy spanning tree can be used to approximately solve the traveling salesman problem. A convenient formal way of defining this problem is to find the shortest path that visits each point exactly once. This problem can be solved by many different algorithms, which are recently developed in research topics. There are several "best" algorithms, depending on the following assumptions:

- A randomized algorithm can solve it in linear expected time.
- It can be solved in worst case time if the weights are large integers.
- Select the minimum weightage unused edge in the fuzzy graph; highlight it.

II. DEFINITION

Definition 2.1: A fuzzy graph with V as the underlying set is a pair G: (A, Γ) where A: V → [0,1] is a fuzzy subset, Γ: V x V -----> [0,1] is a fuzzy relation on the fuzzy subset A, such that Γ(u,v) ≤ A(u) ∩ A(v) for all u,v ∈ V.

Definition 2.2: A fuzzy Hamiltonian circuit is a circuit that visits every vertex in a fuzzy graph once with no repeats, being a fuzzy Hamiltonian circuits must start and end at the same vertex.

Definition 2.3: A fuzzy Hamiltonian path is a path that passes through each of the vertices in a fuzzy graph exactly once.

Definition 2.4: A fuzzy spanning tree is a fuzzy tree which covers all the vertices of a fuzzy graph.

NOTE 2.1:

Fuzzy trees has no circuits, and it is fine to have vertices with degree higher than two.

EXAMPLE 2.1:

A fuzzy spanning tree of a fuzzy graph is just a fuzzy subgraph that contains all the vertices and is a fuzzy tree. A fuzzy graph may have many fuzzy spanning trees; for instance the complete fuzzy graph on n vertices has n(n-1)/2 fuzzy spanning trees.
Forming the fuzzy spanning tree by finding a fuzzy subgraph - a new fuzzy graph formed using all the vertices and some of the edges from the original fuzzy graph.

**Brute Force Algorithm**

Step 1: List all possible Fuzzy Hamiltonian circuits.
Step 2: Find the length of each circuit by adding the edge weights.
Step 3: Select the circuit with minimal total weight.

**Example: 2.2**

<table>
<thead>
<tr>
<th>SL.NO</th>
<th>Fuzzy Hamiltonian circuits</th>
<th>Weights</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>x₁x₂x₃x₄x₅x₁</td>
<td>0.36</td>
</tr>
<tr>
<td>2.</td>
<td>x₁x₂x₃x₄x₅x₁</td>
<td>0.85</td>
</tr>
<tr>
<td>3.</td>
<td>x₁x₂x₃x₄x₅x₁</td>
<td>0.64</td>
</tr>
<tr>
<td>4.</td>
<td>x₁x₄x₃x₅x₃x₁</td>
<td>0.85</td>
</tr>
<tr>
<td>5.</td>
<td>x₁x₅x₃x₅x₄x₁</td>
<td>0.57</td>
</tr>
</tbody>
</table>

There are unique circuits on this fuzzy graph. All other circuits are the reverse of the listed ones or start at a different vertex, but result in the same weights. From this we can see that the first circuit, x₁x₂x₃x₄x₅x₁, is the fuzzy optimal circuit.

**Nearest Neighbor Algorithm (NNA)**

Step 1: Select a starting point.
Step 2: Move to the nearest unvisited vertex (the edge with smallest weight).
Step 3: Repeat until the circuit is complete.

Consider Fig 2.1, starting at vertex x₂, the nearest neighbor is vertex x₅ with a weight of 0.06. From x₅, the nearest neighbor is x₁, with a weight of 0.02. From x₁, the nearest neighbor is x₃, with a weight of 0.2. From x₃, our only option is to move to vertex x₄, the only unvisited vertex, with a weight of 0.12. From x₄ we return to x₂ with a weight 0.15. The resulting circuit x₂x₅x₁x₃x₄x₂ is with a total weight of 0.55.

**Sorted Edges Algorithm (Minimum Weight Link Algorithm)**

Step 1: Select the minimum weight unused edge in the fuzzy graph; highlight it.
Step 2: Repeat step 1, adding the cheapest unused edge to the fuzzy graph, unless:

a. Addition of edge would create a circuit that doesn’t contain all vertices, or
b. Addition of edge would give a vertex degree 3.

3) Repeat until a fuzzy Hamiltonian circuit containing all vertices is formed.

Consider example:2.2, we have five vertex fuzzy graph, the minimum weight edge is x₁x₅, with a weight of 0.02. The next fuzzy shortest edge is x₅x₄, with a weight of 0.05. The next fuzzy shortest edge is x₄x₃, with a weight of 0.12. The next fuzzy shortest edge is x₃x₂, with a weight of 0.07. The next fuzzy shortest edge is x₂x₄, but that edge would create a fuzzy Hamiltonian circuit x₁x₅x₄x₃x₂.

**EXAMPLE 2.3:** The telephone line connection to plan and connect an efficient fuzzy route for A.V.V.M. Sri Pushpam College (Autonomous), Poondi, Thanjavur District for intercom landline to reach it all the department exactly once and return to starting location (e.g. Office room). Consider each department as vertex such as x₁ – office room, x₂ – mathematics, x₃ – economics, x₄ – History, x₅ – Computer Science, x₆ – Library, x₇ – Physics, x₈ – Chemistry, x₉ – Botany and x₁₀ – Physical education. The distance between them are represented as fuzzy weights in the tabular column 2.1.
Using NNA starting at $x_8$ (Nearest Neighbor Algorithm) with a large number of departments, it is helpful to mark off the departments as they are connected to keep from accidently connecting them again. Looking in the row for $x_8$, the smallest distance is 0.03, to $x_9$. Following that idea, our circuit will be:

- $x_8$ to $x_9$ 0.03
- $x_9$ to $x_4$ 0.02
- $x_4$ to $x_6$ 0.03
- $x_6$ to $x_7$ 0.06
- $x_7$ to $x_{10}$ 0.11
- $x_{10}$ to $x_2$ 0.01
- $x_2$ to $x_3$ 0.25
- $x_3$ to $x_1$ 0.1
- $x_1$ to $x_5$ 0.07
- $x_5$ to $x_9$ 0.34

Therefore, total connecting length is 1 (approximately).

Using **SORTED EDGES ALGORITHM** it is helpful to draw an empty fuzzy graph, perhaps by drawing vertices in a circular pattern. Adding edges to the fuzzy graph and select them with help of any fuzzy circuits or vertices with degree 3.

- $x_{10}$ to $x_2$ 0.01
- $x_4$ to $x_9$ 0.02
- $x_8$ to $x_9$ 0.03
- $x_4$ to $x_6$ 0.03

The next shortest edge is from $x_4$ to $x_7$ at 0.04, but adding that edge would give $x_4$ degree 3. Continuing on, and skip over any edge pair that contains $x_9$ to $x_4$, since they both already have degree 2.

- $x_8$ to $x_{10}$ 0.05
- $x_6$ to $x_7$ 0.06
- $x_8$ to $x_3$ (reject – it forms fuzzy circuit)
- $x_1$ to $x_5$ 0.07
- $x_7$ to $x_2$ (reject – it forms fuzzy circuit)
- $x_7$ to $x_3$ 0.09
- $x_3$ to $x_1$ 0.1

The fuzzy graph after adding these edges is shown in the following figure 2.4. At this point, skip over any edge pair that contains $x_9$, $x_{10}$, $x_6$, $x_8$ or $x_4$ since they already have degree 2.

- $x_7$ to $x_2$ (reject – it forms fuzzy circuit)
- $x_7$ to $x_3$ 0.09
- $x_3$ to $x_1$ 0.1

The fuzzy graph after adding these edges is shown in the following figure 2.3.

At this point the only way to complete the circuit is to add:

- $x_5$ to $x_2$ 0.5

### Table 2.1

<table>
<thead>
<tr>
<th></th>
<th>$x_1$</th>
<th>$x_2$</th>
<th>$x_3$</th>
<th>$x_4$</th>
<th>$x_5$</th>
<th>$x_6$</th>
<th>$x_7$</th>
<th>$x_8$</th>
<th>$x_9$</th>
<th>$x_{10}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$x_1$</td>
<td>-</td>
<td>0.37</td>
<td>0.1</td>
<td>0.22</td>
<td>0.07</td>
<td>0.17</td>
<td>0.25</td>
<td>0.28</td>
<td>0.24</td>
<td>0.35</td>
</tr>
<tr>
<td>$x_2$</td>
<td>0.37</td>
<td>-</td>
<td>0.25</td>
<td>0.16</td>
<td>0.5</td>
<td>0.19</td>
<td>0.07</td>
<td>0.1</td>
<td>0.13</td>
<td>0.01</td>
</tr>
<tr>
<td>$x_3$</td>
<td>0.1</td>
<td>0.25</td>
<td>-</td>
<td>0.12</td>
<td>0.27</td>
<td>0.12</td>
<td>0.09</td>
<td>0.16</td>
<td>0.13</td>
<td>0.24</td>
</tr>
<tr>
<td>$x_4$</td>
<td>0.22</td>
<td>0.16</td>
<td>0.12</td>
<td>-</td>
<td>0.43</td>
<td>0.03</td>
<td>0.04</td>
<td>0.08</td>
<td>0.02</td>
<td>0.15</td>
</tr>
<tr>
<td>$x_5$</td>
<td>0.07</td>
<td>0.5</td>
<td>0.27</td>
<td>0.43</td>
<td>-</td>
<td>0.45</td>
<td>0.47</td>
<td>0.34</td>
<td>0.36</td>
<td>0.42</td>
</tr>
<tr>
<td>$x_6$</td>
<td>0.17</td>
<td>0.19</td>
<td>0.12</td>
<td>0.03</td>
<td>0.45</td>
<td>-</td>
<td>0.06</td>
<td>0.11</td>
<td>0.06</td>
<td>0.18</td>
</tr>
<tr>
<td>$x_7$</td>
<td>0.25</td>
<td>0.07</td>
<td>0.09</td>
<td>0.04</td>
<td>0.47</td>
<td>0.06</td>
<td>-</td>
<td>0.11</td>
<td>0.08</td>
<td>0.11</td>
</tr>
<tr>
<td>$x_8$</td>
<td>0.28</td>
<td>0.1</td>
<td>0.16</td>
<td>0.08</td>
<td>0.34</td>
<td>0.11</td>
<td>0.11</td>
<td>-</td>
<td>0.03</td>
<td>0.05</td>
</tr>
<tr>
<td>$x_9$</td>
<td>0.24</td>
<td>0.13</td>
<td>0.13</td>
<td>0.02</td>
<td>0.38</td>
<td>0.06</td>
<td>0.08</td>
<td>0.03</td>
<td>-</td>
<td>0.11</td>
</tr>
<tr>
<td>$x_{10}$</td>
<td>0.35</td>
<td>0.01</td>
<td>0.24</td>
<td>0.15</td>
<td>0.42</td>
<td>0.18</td>
<td>0.11</td>
<td>0.05</td>
<td>0.11</td>
<td>-</td>
</tr>
</tbody>
</table>
Therefore, total trip length is 0.97

Another fuzzy route can make the tour 
\( x_1x_3x_5x_9x_8x_2x_10x_7x_6x_5x_1 \) is 0.96.

### III. KRUSKAL’S ALGORITHM

Step 1: Select the minimum weight unused edge in the fuzzy graph; highlight it.
Step 2: Repeat step 1, addition of minimum weight unused edge to the fuzzy graph, unless:
Addition of edge would create a fuzzy circuit.
Step 3: Repeat until a fuzzy spanning tree is formed.

**Example: 2.4**

The telephone line connection to plan and connect an efficient fuzzy route for A.V.V.M Sri Pushpam College, (Autonomous), Poondi, Thanjavur District, Tamil Nadu, South India. Consider each department as vertex such as \( x_1 \) – office room, \( x_2 \) – mathematics, \( x_3 \) – economics, \( x_4 \) – History, \( x_5 \) – Computer Science, \( x_6 \) – Library, \( x_7 \) – Physics, \( x_8 \) – Chemistry, \( x_9 \) – Botany and \( x_{10} \) – Physical education. The distance between them are represented as fuzzy weights shown in the Tabular column 2.1.

We find the solution for the problem A.V.V.M Sri Pushpam College, (Autonomous), Poondi, Thanjavur District, South India needs to connect updated intercom lines connecting all the departments exactly once. The problem is to minimize the amount of new line using Kruskal’s Algorithm with fuzzy graph.

**Solution:** Selecting the cheapest unused edge in the fuzzy graph, we have

- \( x_{10} \) to \( x_2 \) 0.01
- \( x_4 \) to \( x_9 \) 0.02
- \( x_5 \) to \( x_6 \) 0.03
- \( x_4 \) to \( x_6 \) 0.03
- \( x_4 \) to \( x_7 \) 0.04
- \( x_9 \) to \( x_6 \) (reject – it forms fuzzy circuit)
- \( x_8 \) to \( x_{10} \) 0.05
- \( x_7 \) to \( x_9 \) (reject – it forms fuzzy circuit)
- \( x_4 \) to \( x_8 \) (reject – it forms fuzzy circuit)
- \( x_6 \) to \( x_7 \) (reject – it forms fuzzy circuit)
- \( x_8 \) to \( x_2 \) (reject – it forms fuzzy circuit)
- \( x_1 \) to \( x_5 \) 0.07
- \( x_6 \) to \( x_8 \) (reject – it forms fuzzy circuit)
- \( x_7 \) to \( x_8 \) (reject – it forms fuzzy circuit)
- \( x_7 \) to \( x_{10} \) (reject – it forms fuzzy circuit)
- \( x_9 \) to \( x_{10} \) (reject – it forms fuzzy circuit)
- \( x_3 \) to \( x_6 \) 0.12
- \( x_3 \) to \( x_9 \) (reject – it forms fuzzy circuit)
- \( x_2 \) to \( x_7 \) (reject – it forms fuzzy circuit)
- \( x_9 \) to \( x_2 \) (reject – it forms fuzzy circuit)
- \( x_4 \) to \( x_{10} \) (reject – it forms fuzzy circuit)
- \( x_8 \) to \( x_3 \) (reject – it forms fuzzy circuit)
- \( x_2 \) to \( x_4 \) (reject – it forms fuzzy circuit)
- \( x_6 \) to \( x_1 \) 0.17

Accepted path is connected and shown in the following figure 2.6.

This connects the fuzzy graph. The total length of cable to lay would be 0.54

### IV. CONCLUSION

The result on comparing with heuristic algorithm such as Nearest Neighbor Algorithm, Sorted Edges Algorithm and Kruskal’s Algorithm for the problem that connecting distances to plan and visit an efficient fuzzy route for A.V.V.M Sri Pushpam College, (Autonomous), Poondi, Thanjavur District, South India intercom landline(phone) to reach it all the department exactly once is Kruskal’s Algorithm. So can we conclude that Kruskal’s Algorithm is the best to adopt for these types of problems.

### REFERENCES


AUTHORS

First Author – Dr. G.Nirmala, Associate Professor in Mathematics, K.N.G.A. College (Women), Thanjavur.
Second Author – K.Uma, Assistant Professor in Mathematics, A.V.V.M. Sri Pushpam College, Poondi, Thanjavur Dist.