RIEM

REVISTA INQUIETUD EMPRESARIAL
ISSN 0121-1048, E-ISSN 2422-3230
2024 VOL. 24 N.º 2, 1–17
HTTPS://DOI.ORG/10.19053/UPTC.01211048.17896

OPTIMIZATION OF CHINESE POSTMAN PROBLEM USING FUZZY-BASED PRIORITY WEIGHTED GRAPH

JAY CHANDRA YADAV[1*], MOHAMMAD RIZWANULLAH[2*] ✉

Citation: Yadav, J. C. & Rizwanullah, M. (2024). Optimization of Chinese Postman Problem Using Fuzzy-Based Priority Weighted Graph. *Inquietud Empresarial*, *24*(2), 1-17. https://doi.org/10.19053/uptc.01211048.17896

Editor: Blanco-Mesa, Fabio

Received: 18/07/2024.
Accepted: 29/09/2024.
Published: 13/10/2024.

JEL codes: C44

Type of article: Methods or Methodologies

Abstract: In the early 1960s, Chinese mathematician Mei-Ko Kwan (M. Guan) introduced a method aimed at minimizing mail carriers' route lengths. This method led to the exploration of various formulations of the Chinese Postman Problem (CPP), resulting in at least eight distinct formulations. In response to a practical concern, a new problem formulation called the Priority Constrained Chinese Postman Problem (PCCPP) emerged. In PCCPP, a linear order is provided for a set of significant nodes, and the objective is to traverse all edges at least once while prioritizing the prompt visitation of higher-priority nodes. This paper presents a modified approach to Chinese Postman Problems (CPP) using priorities in a fuzzy environment ranking function applied to priority nodes in CPP to get the optimal result. Finally, this paper also illustrates and justifies the implementation of the modified approach with a numerical problem. The consideration of multi-factors in a multi-graph with priorities gives another scope of research.

keywords: Priority, Eulerian tour, fleury algorithm, Dijkstra's algorithms, Undirected Graph, Triangular Fuzzy Number.

[1] Department of Mathematics and Statistics; Manipal University Jaipur; Jaipur-Ajmer Express Highway, Dehmi Kalan, Near Gvk Toll Plaza, Jaipur, Rajasthan 303007, India; jay.211051026@muj.manipal.edu; https://orcid.org/0000-0002-1394-8680
[2] Department of Mathematics and Statistics; Manipal University Jaipur; Jaipur-Ajmer Express Highway, Dehmi Kalan, Near Gvk Toll Plaza, Jaipur, Rajasthan 303007, India; rizwansal@yahoo.co.in; https://orcid.org/0000-0002-6270-5562 *Corresponding author

Optimización del problema del cartero chino utilizando un gráfico ponderado con prioridad restringida en un entorno difuso

Resumen: A principios de los años 1960, el matemático chino Mei-Ko Kwan (M. Guan) introdujo un método destinado a minimizar las longitudes de las rutas de los carteros. Este método condujo a la exploración de varias formulaciones del Problema del Cartero Chino (CPP), lo que dio como resultado al menos ocho formulaciones distintas. En respuesta a una preocupación práctica, surgió una nueva formulación del problema denominada Problema del Cartero Chino con Prioridad Restringida (PCCPP). En PCCPP, se proporciona un orden lineal para un conjunto de nodos significativos y el objetivo es atravesar todos los bordes al menos una vez mientras se prioriza la visita rápida de los nodos de mayor prioridad. Este artículo presenta un enfoque modificado para los Problemas del Cartero Chino (CPP) utilizando prioridades en una función de clasificación de entorno difuso aplicada a los nodos prioritarios en CPP para obtener el resultado óptimo. Finalmente, este artículo también ilustra y justifica la implementación del enfoque modificado con un problema numérico. La consideración de múltiples factores en un multigrafo con prioridades brinda otro alcance de investigación.

Palabras clave: Prioridad, recorrido euleriano, algoritmo de Fleury, algoritmo de Dijkstra, grafo no dirigido, número difuso triangular.

1 INTRODUCTION

The famous Chinese Postman Problem is used to find the shortest path for which he could cover every route of that zone once. The Chinese Postman Problem is solved by using the Euler graph to find the smallest path that covers all the edges. In CPP, our goal is to cover each edge at the minimum distance. It has many applications where it is usable in real life, like mail deliveries, waste collection, salt gritting, and snow plowing. The Chinese Postman Problem (CPP), a renowned problem in graph theory and combinatorial optimization, presents an intriguing challenge of finding an optimal route for a postman to deliver mail while considering priorities assigned to each street or path (Sokmen, et al., 2019), September. Originally known as the Route Inspection Problem or the Postman Tour Problem, the CPP gains complexity and real-world relevance when priorities, in the form of weights or costs associated with edges in the graph, are introduced. This variant, known as the" Chinese Postman Problem based on priorities," calls for a sophisticated algorithmic approach to determine the most efficient route that both minimizes the total cost and ensures that every edge is visited at least once (Zhang & Peng, 2015). In this situation, we're going to explore the details of the problem, including how it pertains to real-world scenarios and the methods used to address its computational challenges.

There are two primary types of routing problems: arc and node routing, as mentioned in Figure 1. The goal of arc routing issues is to find the shortest route and pathways that pass through each other at least once on the way back to the starting node Lines along a line (Campbell et al., 2021) .Its origins can be traced back to the remarkable story of Meigu Guan (or Kwan Mei-ko), a Chinese mathematician who found himself immersed in the world of postal work during the tumultuous period of the Chinese Cultural Revolution. It was Guan who, while navigating the responsibilities of a mailman, formulated the fundamental challenge that is the CPP. In his words," A mailman must cover his assigned segment before returning to the post office. The problem is to find the shortest walking distance for the mailman" (Keskin & Triki, 2022).

The roots of the CPP can be traced further back to the famous problem of the Konigsberg bridges. This problem, as it was tackled by Leonhard Euler in 1736, was essentially an early incarnation of the CPP. It sought to answer a critical question: could one find a closed path that traversed each of the seven bridges in the city of Konigsberg exactly once? Euler, in his pioneering work, provided both a necessary and sufficient condition for the existence of such a path on any connected, undirected graph. Alas, for the case of Konigsberg, no such path was to be found. Euler's insights led to the classification of graphs that possess this unique property, now known as Eulerian graphs, and the closed path itself was aptly named the Eulerian tour, Eulerian circuit, or Eulerian cycle.

A Eulerian graph, as defined formally, signifies the existence of a closed path that visits each edge exactly once and each vertex at least once. It's important to note that this definition extends beyond the confines of connected, undirected graphs.

In this way, a method also comes from the stochastic network model, and it works on the Chinese postman problem in stochastic networks (Kramberger & Zerovnik, 2005). There are many methods to solve CPP, like length-constrained k-drone and time-dependent (Kramberger & Zerovnik, 2007; Nilofer & Rizwanullah, 2020). The relationship between a CPP solution and a Eulerian tour is illuminating. In cases where a graph qualifies as a Eulerian graph, the CPP solution aligns perfectly with the Eulerian tour. Conversely, when the graph fails to meet this criterion, some edges must be traversed multiple times to ensure complete coverage. In such scenarios, the CPP tour materializes as the most efficient path amongst these edge-covering routes.

Therefore, when a graph finds itself in a state of non-Eulerian disposition, it becomes necessary to traverse certain edges more than once. The quest then becomes one of locating the most optimal route for these edge repetitions. It's a testament to the elegant interplay between graph theory and practical problem-solving that, by addressing a subproblem of augmentation—the addition of a set of arcs with minimal cost—one can transform the original graph into a Eulerian one. This transformation elegantly equips us to find a solution to the CPP by identifying a Eulerian tour on the revised graph. Then a role of priorities is important to deliver important delivery, in which Given a set of important nodes that are arranged in a linear order, the task is to traverse every edge at least once in a way that ensures the highest-priority nodes are visited as early as possible (Sokmen et al., 2019; Tan et al., 2005; Xin et al., 2022).

Undirected Chinese Postman Problem (UCPP): The Undirected Chinese Postman Problem, often referred to as UCPP, deals with a specific variant of the Chinese Postman Problem (CPP) on undirected graphs, as mentioned in Figure-01 route map. In the realm of UCPP, the Eulerian property plays a pivotal role in its resolution. A Eulerian graph is a critical indicator, as it signifies the existence of a closed path within the graph that traverses each edge exactly once.

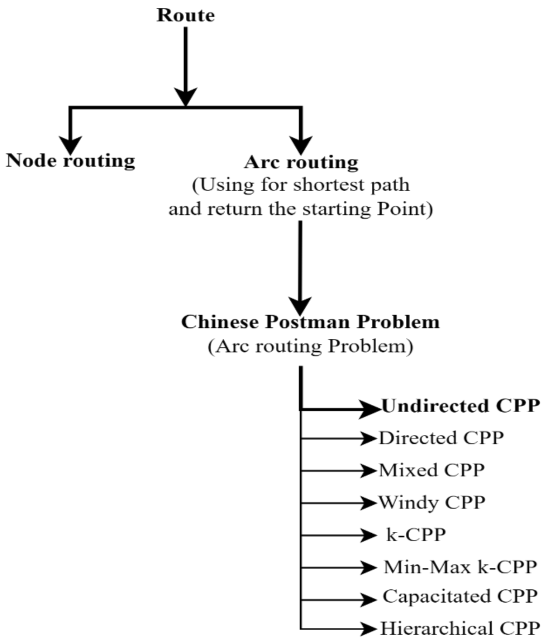

Figure 1. Route map of undirected CPP

In his pioneering work, Guan made an intriguing observation about UCPP. He noted that any graph, G, in this context will invariably have an even number of vertices with an odd degree. Moreover, he demonstrated that a Eulerian graph, denoted as G', can be derived from G by introducing additional edges to connect these vertices with odd degrees. Guan's profound insight led to an essential discovery: the necessity and sufficiency for the optimality of a Eulerian tour on G' hinges on the absence of redundancy. In essence, a Chinese Postman tour should never employ an edge more than twice. Consequently, the length of the Chinese Postman tour, often abbreviated as CP tour, will not surpass twice the length of the original graph, G.

The formulation of the UCPP typically begins with an integer programming model in the initial stage. The primary objective is to ascertain the minimal increase in the cost of transforming G into G' while ensuring that all vertices in G' possess an even degree. This elegant approach encapsulates the essence of the Undirected Chinese Postman Problem, offering a systematic and structured solution to a problem with significant real-world applications.

One way to solve this problem is to first find all the nodes with odd degrees and then pair them up in a way that the overall expense of connecting them with the shortest paths is minimized. This will create a new graph that has a Eulerian circuit, which is a closed walk that visits every edge exactly once. Then, among all the possible Eulerian circuits, we choose the one that visits the priority nodes in the given order as early as possible. But when we start to consider which paths or connections are more important using fuzzy logic, things get a bit more complicated. It’s like saying, “Well, this path might be kind of important, but it’s not totally certain.” So, instead of being very sure about the importance of a path, we become a bit unsure and make decisions based on probabilities and uncertainties (Sokmen et al., 2019; Tan et al., 2005).

In this context, a fuzzy environment is one in which the priorities or weights assigned to the edges are subject to imprecision, vagueness, or uncertainty (Zhang & Peng, 2015; Yadav & Rizwanullah, 2022). Fuzzy logic and fuzzy set theory provide a powerful framework for modeling and solving problems in such uncertain environments. This approach recognizes that real-world scenarios often involve imprecise information or subjective judgments regarding edge priorities, making it challenging to determine an exact, crisp solution.

This problem requires accommodating the inherent uncertainties in edge priorities, making it suitable for an approach that incorporates fuzzy logic and fuzzy optimization techniques. In Figure 2, fuzzy interface, we have shown how to assign priorities. Novel approaches that consider the graph’s shape and the imprecise nature of edge importance are required to solve the Weighted Chinese Postman Problem with fuzzy priorities. This introduction sets the stage for solving this challenging issue with fuzzy logic and optimization methods (Yu & Batta, 2010; Yılmaz, (2021). To deal with the uncertainty of edge priorities, we aim to balance path optimization with path minimization. “Fuzzy” is a term used to describe something that is not clear, well-defined, or precisely delineated. It often refers to situations or concepts where boundaries are not distinct and there is uncertainty or ambiguity involved. In various contexts, “fuzzy” can mean:

• *Fuzzy Logic:* In mathematics and computer science, fuzzy logic is a form of multi-valued logic that allows for degrees of truth instead of just true or false. It’s used to handle imprecise information or approximate reasoning.

• *Fuzzy Sets:* Fuzzy set theory is a mathematical framework that deals with sets whose elements have degrees of membership rather than being strictly in or out of the set.

• *Fuzzy Thinking:* This term is sometimes used to describe thinking that is not entirely clear or well-structured and may involve ambiguity or vagueness.

• *Fuzzy Concepts:* In everyday language, “fuzzy” can describe concepts or ideas that are not well-defined and may lack precision. For example, when describing someone’s appearance as having “fuzzy hair,” it suggests that the person’s hair is not perfectly smooth or clearly defined.

• *Fuzzy Boundaries:* This refers to situations where it’s challenging to determine precise boundaries or distinctions between categories or states. For example, the boundary between colors in a gradient may be fuzzy because there’s no distinct point where one-color ends and another begins.

• *Fuzzy Feeling:* In casual conversation, if someone has a “fuzzy feeling,” it typically means they have a vague, warm, or affectionate emotion without specific or clear reasons.

Fuzzy is used to express the idea of something being uncertain, unclear, or imprecise, where distinctions or definitions are not sharp or well-defined. It's often used in contexts where traditional binary (true or false) logic or clear-cut definitions are inadequate to describe the situation accurately. We used fuzzy to decide the priority based on it.

Fuzzy Membership: These fuzzy representations use triangular membership functions to capture the degree of membership of each data point in the Good, Fair, and Poor Cost categories. A membership value of 1 indicates full membership, while values between 0 and 1 represent partial membership, reflecting the uncertainty or imprecision in the data.

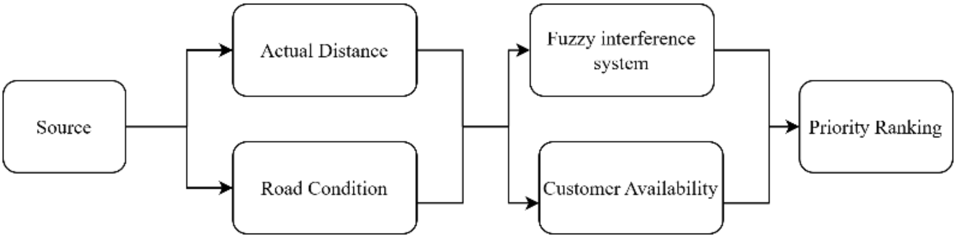

Figure 2. Fuzzy Interface

For Good ($\tilde{\mu}_G$): The function decreases linearly from 1 to 0 between points a and b. For $x \leq a$, the membership value is 1, and for $x > b$, it is 0.

$$\tilde{\mu}_G(x)=\begin{cases} 1 \ \ if \ x \leq a \\ \frac{b-x}{b-a} \ \ if \ a < x \leq b \\ 0 \ \ if \ x > b \end{cases} \tag{1}$$

For Fair ($\tilde{\mu}_F$): This is a more complex case with a triangular shape. The membership value increases linearly from 0 to 1 between points a and b and then decreases linearly from 1 to 0 between points b and c.

$$\tilde{\mu}_F(x)=\begin{cases} 0 \ \ if \ x \leq a \\ \frac{x-a}{b-a} \ \ if \ a < x \leq b \\ \frac{c-x}{c-b} \ \ if \ b < x \leq c \\ 0 \ \ if \ x > c \end{cases} \tag{2}$$

For Poor ($\tilde{\mu}_P$): The function increases linearly from 0 to 1 between points a and b. For $x \leq a$, the membership value is 0, and for $x > b$, it is 1.

$$\tilde{\mu}_P(x)=\begin{cases} 0 \ \ if \ x \leq a \\ \frac{x-a}{b-a} \ \ if \ a < x \leq b \\ 1 \ \ if \ x > b \end{cases} \tag{3}$$

2 THEORICAL FRAMEWORK

The problem of finding an Euler circle in a connected weighted graph is called the Chinese Postman problem. The weighted graph G consists of a non-empty set V whose elements are vertices, a set E of two element subsets of V and a set W of real numbers called weighted on each edge. Such problem is readily transformed into network optimization problem; like street can be transformed into arcs or edges and junction into nodes of a network and the distance traveled is considered as length associated with the arc.

Therefore, The Chinese Postman Problem (CPP), has number of applications like street cleaning, postal delivery, and other fields, aims to determine the shortest path that passes by each edge of a graph at least once. Optimization is more difficult in non-Eulerian graphs because additional edges must be created to balance vertices (Sokmen, et al., 2019).

Conventional CPP assumes fixed edge weights (e.g., cost, distance), however variances arise from real-world conditions such as weather and traffic. Furthermore, depending on their significance or urgency, various edges can have varied priorities, which would add complexity that is not addressed by normal CPP approaches. Fuzzy logic is a valuable tool for describing variable factors in optimisation problems because of its ability to handle ambiguity. Fuzzy-based CPP provides an adaptable solution for real-world scenarios by modelling variable edge weights and incorporating subjective elements like prioritization.

3 PROBLEM DESCRIPTION

In our study, we confront the challenge of the Chinese Postman problem within the realm of undirected weighted graphs, as visually depicted in the accompanying figure 3. This problem introduces an intriguing layer of complexity by incorporating priority assignments to specific nodes. Our task involves postmen who need to visit certain priority nodes, and the order in which they visit these nodes is determined by a fuzzy ranking system. This ranking system takes into consideration factors like customer needs, the availability of these nodes, and the state of the roads and traffic conditions.

Now, the Chinese Postman problem is all about planning the best route. We want to make sure we cover all the roads in our network at least once and then return to where we started. And, of course, we want to do this while keeping the cost as low as possible, whether that’s in terms of distance or time. This problem is quite intricate, highlighting the importance of smart route planning and finding the right balance between giving priority to specific locations and making the whole journey as cost-effective as possible.

4 METHODOLOGY

4.1 Mathematical formulation of CPP

The Chinese Postman Problem (CPP) can be mathematically formulated as follows:

Input: a connected, undirected graph G = (V, E), where V is the set of vertices and E is the set of edges. Each edge e in E is associated with a non-negative integer length (or weight) w(e). The graph may contain multiple edges between the same pair of vertices (i, j), and the lengths of these multiple edges may vary.

Objective: Find a closed walk (a path that starts and ends at the same vertex) that traverses every edge in the graph at least once (a minimum length tour), minimizing the total length or weight of the walk.

Decision Variables: Let x(e) be a decision variable that takes a binary value, indicating whether edge e is included in the postman's tour.

• x(e) = 1 (if edge e is selected (included in the tour).

• x(e) = 0 (if edge e is not selected (not included in the tour).

Constraints: The tour must include at least one visit to each vertex to ensure that all edges are traversed. For each vertex v in V, the sum of x(e) for all edges e incident to vertex v must be even if v has an even degree. If v has an odd degree, the sum must be odd. constraints to ensure that the tour is connected and minimizes the total length. Objective Function: Minimize the total length of the tour, expressed as the sum of the lengths of the selected edges.

4.1 The algorithm

We proposed a solution to the Chinese Postman Problem

$$min\Sigma\big(w(e) * x(e)\big)\forall e \in E. \quad (4)$$

(CPP). Which is a constrained priority ranking in a fuzzy environment based on the distance, customer availability, and condition of the road. Fleury's algorithm for constructing a Eulerian Walk and Dijkastra's computation of the shortest paths to reach the priority nodes. Some steps are mentioned to complete the whole process:

4.1.1 Node Selection and Prioritization

The algorithm starts by selecting a node with the highest priority based on criteria like distance, customer availability, and road conditions within a fuzzy environment.

4.1.2 Identification of Odd Nodes

Odd nodes are identified in the graph. These are nodes with an odd degree, meaning they have an odd number of edges connected to them.

4.1.3 Graph Augmentation

Shortest paths between pairs of nodes are added to the original graph, potentially introducing double edges. This step makes the graph more balanced in terms of node degrees. The resulting graph is denoted as G_0.

4.1.4 Minimum Matching

Any polynomial algorithm for finding a minimum matching in an auxiliary graph can be used. This matching helps pair up the odd nodes.

4.1.5 Eulerian Walk Construction

In the second phase, a Eulerian Walk on the new graph is constructed. The walk starts from the depot and follows the following rules:

Select a walk from the depot (v_0) to the first- priority node (v_1), ensuring that the graph G_0, without this walk, remains connected. This walk is denoted as P_1.

Remove P_1 from the graph ($G_1 = G_0 - P_1$)and find a walk (P_2) from the first-priority node (v_1) to the second priority node (v_2),ensuring that G_1, without P_2, remains connected. If the second priority node has already been visited, P_2 is empty.

Repeat this process for all priority nodes, visiting them in sequence.

4.1.6 Completion of the Walk

Finally, a walk back to the depot (V_0) is constructed, covering all the edges in the modified graph (G_k).

This algorithm combines graph modification, minimum matching, and a Eulerian Walk to find the shortest route that visits all edges in the graph while considering node priorities and the condition of the road. The specific details and implementation of this algorithm may vary, but it aims to solve a more complex variant of the Chinese Postman Problem. Using this construction, an optimal solution to the problem is found.

5 NUMERICAL EXAMPLE

Example: Given a graph in Figure 3 and the priority nodes F, G, and D, find the optimal solution for PCCPP.

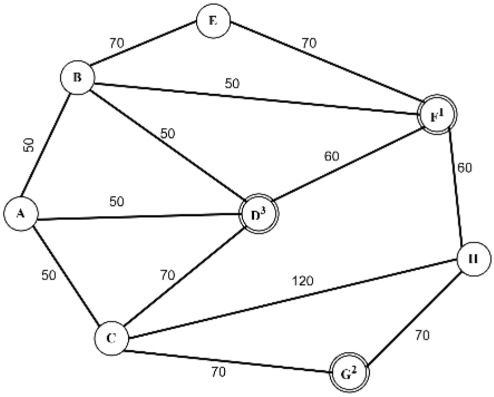

Figure 3. Route without Priorities

Let's imagine the starting point is node A. Our mission is to efficiently navigate all the roads, scattering along them as swiftly as possible while making sure to visit specific priority locations or segments in the shortest amount of time. These priority spots, in descending order of importance, are identified as follows: intersection F, intersection H, and intersection G.

To achieve this, we need to employ a specific algorithm that revolves around creating a Eulerian graph, which essentially means a graph where every node has an even degree (an even number of connections to other nodes). However, our initial graph might contain nodes with odd degrees. To rectify this, we introduce new connections to ensure that all nodes have even degrees.

The procedure involves selecting all the nodes with odd degrees and constructing an auxiliary graph. Since the edge weights in this auxiliary network match the distances between the nodes in the original graph, it is practically a complete graph. In our situation, we have identified two nodes with odd degrees and possess a matrix denoting edge weights in Table 1, or distances, between these nodes.

Table 1. Table of odd nodes

	A	**H**
A		3
H	3	

To identify a minimal match, we employ an algorithm designed for minimum-perfect matching with time complexity considerations. In our specific scenario, the most favorable solution involves connecting points $A \rightarrow H$. This optimal pairing can be swiftly determined, even by examining all potential cases. Consequently, we obtain a modified graph, as we have shown in Figure 4, denoted as G_0, in which every node now boasts an even degree.

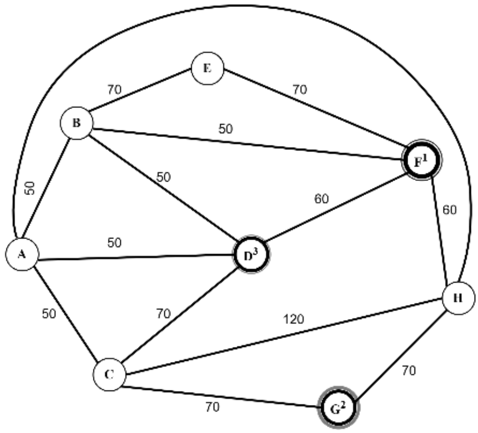

Figure 4. All Even nodes with Priorities

Once we've successfully constructed the Euler graph, the next step is to identify the Euler Walk, a task facilitated by Fleury's algorithm. It's worth noting that Fleury's algorithm provides substantial flexibility in walk construction, so long as we adhere to one essential constraint: we must avoid "crossing bridges." In simpler terms, we should not disconnect the remaining portion of the graph during our walk.

In our case, given the presence of priority nodes, our goal is to reach these nodes as swiftly as possible. However, it's important to note that we are not necessarily seeking the shortest path. Instead, we aim to identify the shortest route between those that ensure G1, where G1 is the result of removing the priority nodes P_1 from the initial graph G0, remains a linked graph.

To pinpoint these most direct routes, we can effectively utilize Dijkstra's algorithm, which calculates the distances di between the starting node and the final node. It's worth mentioning that Dijkstra's algorithm operates with a time complexity of O(n^2).In our specific example, we can compute the distances di for various nodes, as illustrated in the accompanying figure.

Finding the quickest route between points A and F (Figure 5), our approach begins with node F and systematically constructs the shortest path in a reverse manner. To clarify, let's denote di as the distance value from a given node to A, and u_(i,j) as the distance between node B and its neighboring nodes under observation. It becomes evident that those neighbors for which di equals d_F-u_(i,j) are part of the shortest paths from A to F.

When there are multiple nodes to consider, each one is uniquely labeled, and we repeat the same procedure for each of them. This process allows us to pinpoint the shortest path, denoted as P1, between A and F. Since G0 −P1 maintains its connectivity, we update our graph to G1 by subtracting P1 and continue our journey.

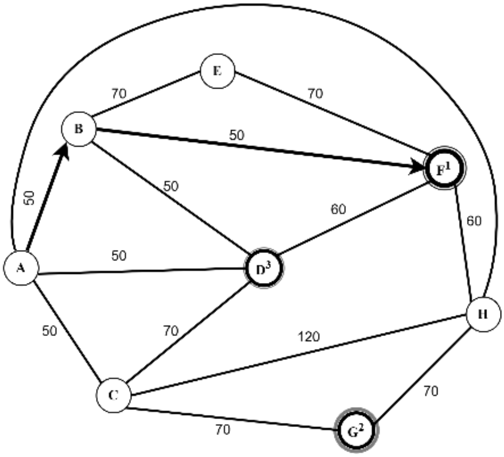

Figure 5. Tour of first priority

Now, with the next priority node, G, still unvisited, we proceed to find the shortest path from F to G, ensuring it adheres to the connectivity condition. By iterating through the above procedure once more, we successfully determine the shortest path from F to G, as depicted in Figure 6.

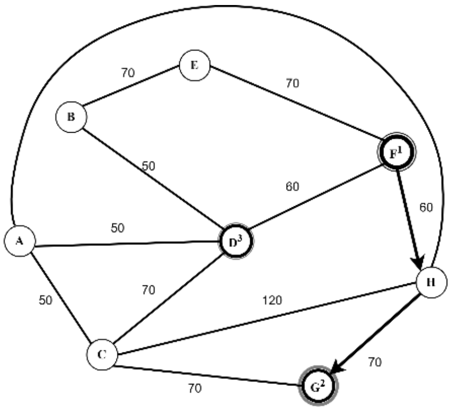

Figure 6. Tour of Second priority

In the final stage of our process, we revisit the procedure, this time starting from the last priority node, D. Our objective is to identify the shortest path between nodes G and D. As we carry out this step, we gradually refine our graph. The result is illustrated in the accompanying Figure 7.

Once we've successfully visited all the priority nodes, we arrive at the graph denoted as G3. At this point, G3 exhibits connectivity with precisely two nodes of odd degree: D and A. To trace the path connecting them, we employ Fleury's algorithm, which effectively guides us in traversing all the edges starting from node D and concluding at node A.

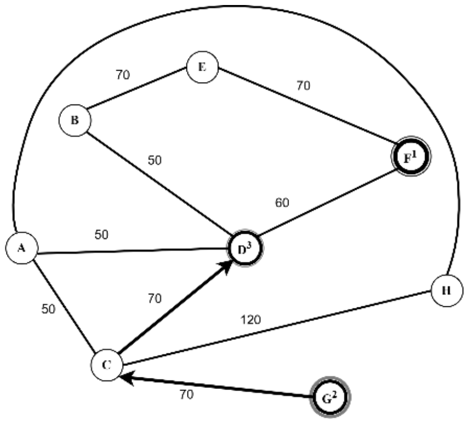

Figure 7. Tour of Third priority

In summary, our journey leads to the construction of the walk, as visually represented in Figure 8.

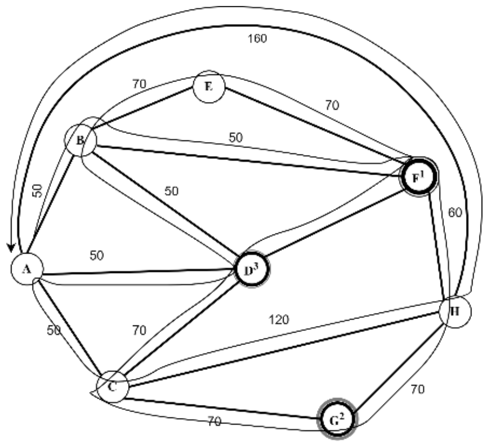

Figure 8. Tour of whole node

6 RESULTS

In this study, we have successfully obtained the optimal solution for the Chinese postman problem while incorporating a fuzzy-based priority approach. By employing this innovative methodology, we have achieved a comprehensive solution that not only minimizes the overall path length but also prioritizes specific nodes in a flexible and context-sensitive manner. For the first-priority node.

$A \rightarrow B \rightarrow F$ For the Second priority node
$F \rightarrow H \rightarrow G$ For the Third priority node
$G \rightarrow C \rightarrow D$ Remaining nodes walk to cover all the nodes using the shortest path and return the starting node.

$D \rightarrow F \rightarrow E \rightarrow B \rightarrow D \rightarrow A \rightarrow C \rightarrow H \rightarrow A.$

Incorporating fuzzy ranking-based priority while considering the shortest path approach has enabled us to achieve the optimal solution, ensuring coverage of all nodes in the network.

$A \rightarrow B \rightarrow F \rightarrow H \rightarrow G \rightarrow C \rightarrow D \rightarrow F \rightarrow E \rightarrow B \rightarrow D \rightarrow A \rightarrow C \rightarrow H \rightarrow A = 920$

Our results have demonstrated the effectiveness of this approach in efficiently navigating a network of roads, considering the varying importance of different segments. This optimization strategy allows for the early visitation of priority nodes, enhancing the overall quality and applicability of the solution.

Furthermore, our findings indicate that by adjusting the weighting factors in the fuzzy-based priority system, we can adapt the solution to different scenarios and prioritize nodes based on factors such as availability, road conditions, and traffic situations. This flexibility offers a valuable tool for decision-makers in transportation and logistics, allowing them to fine-tune the solution to meet specific real-world requirements.

7 CONCLUSIONS

In this study, our research presents a novel and effective approach to solving the CPP with fuzzy-based priority, providing an optimal solution that not only minimizes path length but also adapts to dynamic and context-dependent circumstances. This approach has the potential to significantly enhance the efficiency and cost-effectiveness of route planning in various practical applications.

The example demonstrates the algorithm’s ability to solve the Chinese postman problem while prioritizing the early visitation of crucial nodes. After applying fuzzy ranking to determine their relative importance, it becomes straightforward to identify the top-priority node. The algorithm then efficiently routes the path to reach this node as quickly as possible. The same principle applies to the remaining priority nodes, adhering to the previously established conditions. This is the graph of fuzzy ranking-based priority as shown in Figure 9.

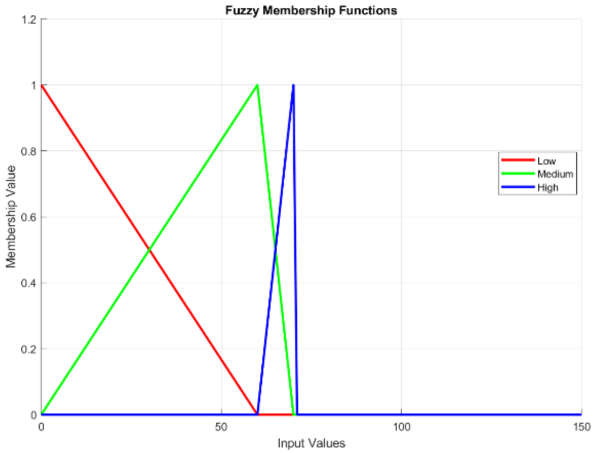

Figure 9. Fuzzy membership function

The algorithm, with its defined cost function, offers an optimal solution for the problem at hand. However, it's worth noting that in many practical scenarios, it may not be practical or necessary to strictly minimize the total path length, as in the case of the Chinese postman tour. Instead, assigning costs to individual edges and considering factors like availability, road conditions, or traffic situations can be more realistic. By varying the ratios between these cost factors, we can potentially derive various optimal solutions to the optimization problem, each tailored to the specific needs and priorities of the situation at hand.

AUTHORS' CONTRIBUTIONS

Jay Chandra Yadva: Methodology, Resources; Software, Validation; Formal analysis; Data curation; Resource acquisition; writing – original draft. Mohammad Rizwanullah: Conceptualization, methodology, Investigation, supervision, Writing (draft review and revision/correction).

FUNDING

This research did not receive external funding.

CONFLICTS OF INTEREST

The authors declare that they have no conflicts of interest.

REFERENCES

Campbell, J. F., Corberán, A., Plana, I., Sanchis, J. M., & Segura, P. (2021). Solving the length constrained K-drones rural postman problem. *European Journal of Operational Research*, *292*(1), 60-72. https://doi.org/10.1016/j.ejor.2020.10.035

Keskin, M. E., & Triki, C. (2022). On the periodic hierarchical Chinese postman problem. *Soft Computing*, *26*(2), 709-724. https://doi.org/10.1007/s00500-021-06213-2

Kramberger, T., & Zerovnik, J. (2005). Chinese postman problem with priorities. In *Proceedings of the International Conference on Systems, Operational Research and Informatics (SOR)* (Vol. 5, pp. 357-362).

Kramberger, T., & Zerovnik, J. (2007). Priority constrained Chinese postman problem. *Logistics & Sustainable Transport*, *1*(1), 33-46.

Nilofer, M., & Rizwanullah, M. (2020). An implementation of Chinese postman problem with priorities. *Journal of Intelligent & Fuzzy Systems*, *38*(3), 3301-3305. https://doi.org/10.3233/JIFS-190035

Sokmen, O. C., Emec, S., Yilmaz, M., & Akkaya, G. (2019, September). An overview of Chinese postman problem. In *Proceeding of the 3rd International Conference on Advanced Engineering Technologies*.

Tan, G., Cui, X., & Zhang, Y. (2005, October). Chinese postman problem in stochastic networks. In *Proceeding in Joint International Conference on Autonomic and Autonomous Systems and International Conference on Networking and Services (pp. 78-78).* IEEE. https://doi.org/10.1109/ICAS-ICNS.2005.31

Xin, J., Yu, B., D'Ariano, A., Wang, H., & Wang, M. (2022). Time-dependent rural postman problem: Time-space network formulation and genetic algorithm. *Operational Research*, *22*(3), 2943-2972. https://doi.org/10.1007/s12351-021-00639-0

Yadav, J. C., & Rizwanullah, M. (2022, December). Combinatorial Optimization of Type-l Fuzzy Travelling Salesman Problem. In *Proceeding of the 2022 International Conference on Computational Modelling, Simulation and Optimization (ICCMSO)* (pp. 108-116). IEEE. https://doi.org/10.1109/ICCMSO58359.2022.00033

Yılmaz, M. (2021). Hierarchical Chinese postman problem with fuzzy travel times. *Iranian Journal of Fuzzy Systems*, *18*(5), 87-105. https://doi.org/10.22111/ijfs.2021.6257

Yu, W., & Batta, R. (2010). Chinese Postman Problem. In*Wiley Encyclopedia of Operations Research and Management Science*. https://doi.org/10.1002/9780470400531.eorms0142

Zhang, B., & Peng, J. (2015). Chinese Postman Problem in Unecertain Network. *Proceedings of the Conference on Operations Research.*