زانكۆی سەلاحەدین - هەولێر
Salahaddin University-Erbil

Optimization in Graph Theory

Research Project

Submitted to the Department of Mathematics in partial fulfillment of the requirements for the degree of BSc. in MATHEMATIC

Prepared by:

Ayub Ali

Supervised by:

Assist. Prof. Dr. Ivan Subhi Latif

April – 2024

Certification of the Supervisor

I certify that this research was prepared under my supervision at the Department of Mathematics / College of Education / Salahaddin University-Erbil in partial fulfillment of the requirements for the degree of Bachelor of philosophy of science in Mathematics.

Signature:

Supervisor: Ivan Subhi Latif

Scientific grade: Lecturer

Date: 8/ 4 / 2024

In view of the available recommendations, I forward this report for debate by the examining committee.

Signature:

Name: Dr. Rashad Rasheed Haji

Scientific grade: Assistant Professor

Chairman of the Mathematics Department

Date: 8 / 4 / 2024

Acknowledgments

I would like to start by thanking my supervisor, Dr. Ivan, for her invaluable guidance and support throughout this research project. It has been an honor to work with her and learn from her expertise. Her insightful feedback and encouragement have been instrumental in completing this paper.

I also wish acknowledge the mathematical department and its teachers for providing me with a strong foundation in mathematics and for their support and encouragement throughout my studies.

Furthermore, I am grateful to my friend shady wahid for his assistance with my English writing skills. His feedback and support were truly helpful and greatly appreciated.

Finally, I would like to express my gratitude to my family for their unwavering support, encouragement, and love throughout my academic journey. Their support has been invaluable and has helped me to pursue my goals with confidence.

Abstract

Graphs are often used to model optimization problems. In optimization, the goal is to find the best solution from a set of possible solutions. Graphs provide a visual representation of relationships between entities, making them useful for modeling various optimization scenarios such as finding the shortest path, minimizing costs, maximizing efficiency, or optimizing resource allocation. Optimization algorithms often operate on graphs to efficiently search for the optimal solution by traversing nodes and edges based on certain criteria or constraints. Optimization, particularly in graph theory, offers a systematic approach to tackling complex urban issues like traffic congestion. Duhok, like many cities, faces challenges due to population growth and increased vehicular traffic, leading to congestion, delays, and pollution. This study proposes a solution for Duhok's traffic congestion, using principles from the Chinese Postman Problem (CPP) and advocating for a modern train system. By applying CPP methodology, the study aims to identify optimal vehicular routes, reducing congestion and improving travel experiences for residents.

Table of Contents

List of Figures

CHAPTER ONE

INTRODUCTION

1.1 History of graph theory

Graph theory, an essential field across various disciplines like mathematics, engineering, sciences, linguistics, and computer science, dates back to 1735 when Leonhard Euler resolved the Königsberg bridge problem. It involves studying graphs, which consist of nodes (or vertices) interconnected by edges. Given its status as a branch of applied mathematics, it's not uncommon for the theory to have been independently conceived multiple times. (agarwal, et al., 2019)

Who invented graph theory:

Euler, born in 1707 and passing in 1782, presented a paper in 1736 where he tackled the Königsberg bridge problem, igniting the emergence of graph theory. His solution in 1736 is often considered the starting point of graph theory, earning him the title "Father of Graph Theory." (agarwal, et al., 2019)

Nothing more was done in this field for the next 100 years.

Then, in 1847, G. R. Kirchhoff (1824-1887) developed the tree theory in electrical networks for their applications. Kirchhoff's research into electric networks led to the development of the fundamental concepts and theorems relating to trees in graphs.

10 years later, a. Cayley (1821-1895) thought about trees that arose from the counting of organic chemical isomers and discovered trees while trying to count the isomers of saturated hydrocarbons C_nH_{2n+2}.

Two other milestone in graph theory were established around the time of Kirchhoff and Cayley.

1. One milestone is the four-color conjecture, and
2. The other milestone is puzzle invented by Hamilton. (agarwal, et al., 2019) **Four-color conjecture:**

According to the four-color conjecture, a minimum of four colors suffices to color any atlas (a map on a plane) in such a way that countries sharing borders are assigned distinct colors.

In 1840, the inception of the four-color problem is credited to A. F. Möbius (1790-1868) through his lectures. Approximately a decade later, A. De Morgen (1806-1871) delved into this quandary in London. By 1879, Cayley introduced this problem in the "inaugural volume of the proceedings of the Royal Geographic Society," catapulting it to widespread recognition. Subsequently, the renowned four-color conjecture gained prominence and has since retained its allure.

since. (agarwal, et al., 2019)

Puzzle invented by Hamilton:

In 1859, Sir E. R. Hamilton (1805-1865) devised a puzzle methodology relating to graphs and sold it to a gaming enterprise in Dublin for guineas.

The puzzle comprised a wooden regular dodecahedron—a polyhedron with 12 faces, all of which are regular pentagons, along with 20 corners and 30 edges. At each corner, three edges intersected. The names of 20 significant cities adorned the corners.

The objective of the puzzle entailed discovering a route along the dodecahedron's edges that traversed each of the 20 cities precisely once. However, despite the simplicity of solving this specific problem, no one has yet managed to establish a necessary and sufficient condition for the existence of such a route (known as a Hamiltonian circuit) in any graph.(agarwal, et al., 2019)

After this fruitful period, there was a half-century of relative inactivity.

Subsequently, in the 1920s, a renewed wave of enthusiasm for graphs emerged, with D. koning emerging as one of the trailblazers during this era. He not only synthesized the contributions of other mathematicians but also advanced his own research, culminating in the publication of the inaugural book on the subject in 1936.The past three decades have witnessed a surge of activity in graph theory—spanning both theoretical and practical domains. A substantial volume of research has been undertaken and persists in this field. Over the last decade alone, thousands of papers and more than a dozen books have been published, attesting to the ongoing vibrancy and significance of graph theory (agarwal, et al., 2019)

1.2 Use of the graph theory

1.2.1 Transportation Networks:

In mathematics, the term "graph" can have two different meaning, depending on the context. The first meaning is related to the graphical representation of mathematical function or a relation, which shows the relationship between the input and output values of the function or the relation.

However, the second meaning of the term "graph" is the one most commonly used in graph theory, which is a mathematical structure consisting of a set of vertices(also called nodes or points) and a set of edges (also called links or arcs) connecting them. This type of graph is used in a wide range of applications including transportation networks, social networks, computer networks, and many others.

In transportation networks, graph theory is a powerful tool for solving problems such as finding the shortest route between two points, determining the optimal location of facilities such as airports or train station, and analyzing traffic flow patterns. By representing the transportation network as a graph, mathematicians and transportation planners can use graph theory to develop models that help them understand and optimize transportation systems. (Mrs.M.durgadevi & Ms.Ch, 2018)

1.2.2 Intelligent Transportation System:

Sensing technologies play a crucial role in the operation intelligent transportation systems (ITS). These technologies are used to collect real-time data on traffic conditions and transmit information to the ITS control center for analysis and decision-making.

For example, in-vehicle sensors can detect changes in speed, acceleration, and braking, which can be used to predict potential accidents or traffic congestion. Roadside sensors can detect the presence of vehicle and pedestrians and can provide real-time data on traffic flow and volume. Additionally, video vehicle detection technologies can capture images of vehicles passing through a designated area and transmit this information to the control center for analysis.

The use of sensing technologies in ITS can lead to significant benefits, such as improved traffic flow, reduced congestion, and increased safety for drivers and pedestrians. Real time traffic data can be used to adjust traffic signal timings and optimize traffic flow, which can reduce travel times and improve fuel efficiency. Additionally, sensing technologies can be used to detect and respond to accidents or other incidents quickly, reducing the risk of further accidents and improving emergency response times.

Overall, the integration of sensing technologies in ITS can help to address many of the challenges associated with modern transportation systems, such as traffic congestion, safety concerns, and environmental impacts. As technology continues to advance, we can expect to see further innovations in sensing technologies and their applications in intelligent transportation systems. (Mrs.M.durgadevi & Ms.Ch, 2018)

CHAPTER TWO

Background

Definitions 2.1: (Bondy, 2006) Graph, a graph is basically a collection of dots, with some pairs of dots being connected by lines. The dots are called vertices, and the lines are called edges.

Definitions 2.2: (Bondy, 2006) Vertex, a vertex is a 'dot' in a graph. The plural of vertex is 'vertices', 'this graph has five vertices. Other synonyms for vertex are node and point. Here are the vertices of a graph.

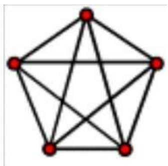


Figure 1 graph showing vertices

Definitions 2.3: (Bondy, 2006) Edge, an edge connects two vertices in a graph. We call those vertices endpoints of the edge. Other synonyms for edge are arc,

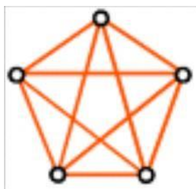


Figure 2 graph showing Edges

link and line. here is the edge of a graph.

Definitions 2.4: (Bondy, 2006) Adjacent, two vertices are adjacent if they are connected by an edge. We often call these two vertices' neighbors. Two adjacent vertices:

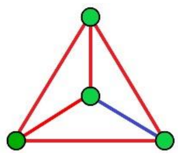


Figure 3 graph showing adjacent vertices

Definitions 2.

5: (Bondy, 2006) Bridge, an edge in a graph whose removal (leaving the vertices) results in a disconnected graph.

Definitions 2.6: (Bondy, 2006) Bipartite Graph, a graph is bipartite if the vertices can be partitioned into two sets, X and Y, so that the only edges of the graph are between the vertices in X and the vertices in Y.

Definitions 2.7: (Bondy, 2006) Size, the size of the graph is the number of edges it has.

Definitions 2.8: (Bondy, 2006) Degree, the degree of a vertex is the size of its neighborhood. The degree of a graph is maximum degree of all of its vertices.

Definitions 2.9: (Bondy, 2006) Pendent Vertex, a vertex of degree 1. Also known as a leaf.

Definitions 2.10: (Bondy, 2006) Order, the order of a graph is the number of vertices it has.

Definitions 2.11: (Bondy, 2006) Walk, a walk is a series of vertices and edges.

Definitions 2.12: (Bondy, 2006) Closed Walk, is a walk from a vertex back to itself.

Definitions 2.13: (naduvath, 2017) Trial, a trial is a walk that does not pass the same edge twice. A trail might visit the same vertex twice, but only if its comes and goes from a different edge each time.

Definitions 2.14: (naduvath, 2017) Tour, a tour is a trial that begins and ends on the same vertex.

Definitions 2.15: (Bondy, 2006)Cycle, cycle is a closed walk with no repeated vertices (expect that the first and last vertices are the same).

Definitions 2.16: (Bondy, 2006) Path, a path is a walk where no repeated vertices.

Definitions 2.
A **u-v** path is a path beginning at u and ending at v.

17: (stephanie, 2019) **(**U-V) Walk, a (u-v) walk would be a walk beginning at (u) and ending at (v).

Definitions 2.18: (stephanie, 2019) Directed Graph, a directed graph is a graph where the edges have direction, that is, they are ordered pairs of vertices.

Definitions 2.19: (stephanie, 2019)Loop, a loop is an edge or arc that joins a vertex to itself.

Definitions 2.20: (stephanie, 2019) Multigraph, a multigraph is a graph without loops, but which may have multiple edges.

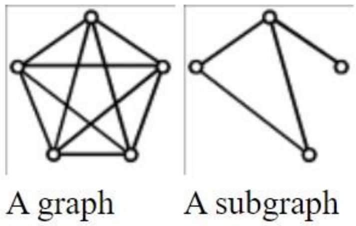
Definitions 2.21: (stephanie, 2019) Null Graph, a null graph is a graph with no edges. It may have one or more vertices.

Definitions 2.22: (stephanie, 2019) Simple Graph, a simple graph that doesn't have any loops or multiple edges.no multiple edges means that no two edges have the same endpoints.

Definitions 2.23: (stephanie, 2019) Trivial Graph, a trivial graph is a graph with only one vertex.

Definitions 2.24: (stephanie, 2019) Undirected Graph, an undirected graph is a graph where none of the edges have direction, the pairs of vertices that make up each edge are unordered.

Definitions 2.25: (Bondy, 2006) Subgraph, a subgraph of graph is some smaller portion of that graph. Here is an example of a subgraph:



Definitions 2.

Figure 4 Graph, showing all vertices and edgesFigure 4 Subgraph, showing a subset of the vertices and edges from

26: (Bondy, 2006) Spanning Subgraph, a subgraph of the graph G which contains of all the vertices of G.

Definitions 2.27: (Bondy, 2006) Tree, a connected graph containing no circuits.

Definitions 2.28: (Bondy, 2006)Spanning Tree, a spanning subgraph of a graph which is also tree.

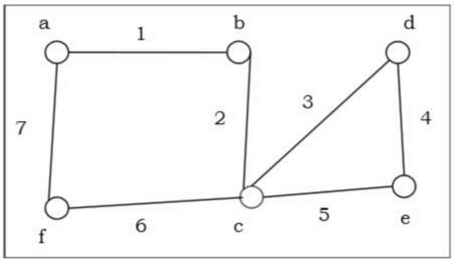
Definitions 2.29: (Bondy, 2006)Circuit, in a graph, a circuit is a simple, closed walk.

Definitions 2.30: (Bondy, 2006)Connected, a connected graph is one in which every pair of vertices are joined by a chain.

Definitions 2.31: (Mohtashim, 2019) Euler Graph, a connected graph G is called an Euler graph, if there is a closed trial which includes every edge of the graph G.

Definitions 2.32: (Mohtashim, 2019) Euler Path, a Euler path is a path that uses every edge of a graph exactly once. a Euler path starts and ends at different vertices.

Definitions 2.33: (Mohtashim, 2019) Euler Circuit, a Euler circuit is a circuit that uses every edge of a graph exactly once. A Euler circuit always starts and ends at the same vertex a connected graph G is a Euler graph if and only if all vertices of G are of even degree, and connected graph G is Eulerian if and only if its edge set can be decomposed into cycle.



Definitions 2.

Figure 5 graph showing a Euler circuit

The above graph is a Euler graph as a 1 b 2 c 3 d 4 e 5 c 6 f 7 g covers all edges of the graph.

Theorem 2.1: (Sinha, 2015) The edge connectivity of a graph G cannot exceed the degree of the vertex with the smallest degree in G.

Proof:

The vertex v_i be the vertex with the smallest degree in G. Let $d(v_i)$ be the degree of $v_{i.}$ Vertex v_i can be separated from G by removing the $d(v_i)$ edges incident on vertex $v_{i.}$ hence the theorem.

Theorem 2.2: (Sinha, 2015) The vertex connectivity of any graph G can never exceed the edge connectivity of G.

Proof:

Let a denote the edge connectivity of G. Therefore, Ǝ a cut set S in G with α edges. Let S partition the vertices of G into subsets V_1 and V_2. By removing almost α vertices V_1 or V_2 which the edges in S are incident, we can affect the removal of S (together with all other edges incident on these vertices) from G. Hence the theorem.

Theorem 2.3: (Sinha, 2015) The maximum vertex connectivity one can achieve with a graph G of n vertices and e edges ($e \geq n-1$) is the integral part of the number 2e/n. [2e/n]

Proof:

Every edge in G contributes two degrees. The total (2e degree) is divided among n-vertices. Therefore, there must be at least one vertex in G whose degree is equal to or less than the number 2e/n. The vertex connectivity of G cannot exceed this number.

To show that this value can actually be achieved, one can first construct an n-vertex regular graph of degree equal to the integral part of the number [2e/n] and then add the remaining e-(2/n). [integral part of the number 2e/n] edges arbitrary.

Thus, we can summarize as follows:

$Vertex\ connectivity \leq Edge\ connectivity \leq 2e/n$ and maximum vertex connectivity possible = [integral part of the number 2e/n]. thus, for a graph with 8 vertices and 16 edges, we can achieve a vertex connectivity (and therefore edge connectivity) s high as four$(= 2\frac{16}{8})$

The traffic sensors can be placed on each edge in a cut-set of G determined by its edge connectivity as well as on each vertex of G determined by its vertex connectivity. These sensors will provide complete traffic information for the control system. Thus, optimal locations for the traffic sensors can be obtained by using edge connectivity and vertex connectivity of the compatibility graph G.

Theorem 2.4: (Mrs.M.durgadevi & Ms.Ch, 2018) **One way street problem: Robin's theorem,** the problem of orienting every edge in a graph in such a way that it remains possible to travel between any two vertices is known as Robin's Theorem, named after mathematician guy Robin who first proved it in 1969.

To solve this problem, we can use the concept of strong connectivity in a directed graph. A directed graph is strongly connected if there is a directed path between every pair of vertices. In other words, we can get from any vertex to any other vertex by following the direction of the edges.

To apply Robin's theorem to the problem of making every street in a city oneway, we first represent the streets as edges of a graph and the street corners as vertices. We then start with the initial undirected graph, where each edge represents a two-way street.

To make every street one-way, we need to assign a direction to each edge in such a way that the resulting directed graph is strongly connected. This means that we need to ensure that for any two vertices in the graph, there is a directed path between them.

One way to achieve this is to choose a spanning tree of the initial graph and orient all edges in the tree to point away from the root of the tree. This ensures that there is a directed path from the root to any other vertex in the graph. We then orient the remaining edges arbitrarily, as long as they do not create cycles in the graph.

By using this method, we can ensure that it is still possible to get from any place to any other place in the city, even after making every street one-way. This can help to alleviate traffic congestion and reduce air pollution in urban areas.

Chinese postman's problem (CPP): (Mrs.M.durgadevi & Ms.Ch, 2018) in 1962, a Chinese mathematician called Kuan mei-ko was interested in postman delivering mail. The (CPP) is a classic problem in graph theory that seeks to find the shortest possible route that visits every edge of an undirected graph at least once. In the context of the problem you described, this could be interpreted as finding the shortest route for a postman to deliver mail to a number of streets.

There are several approaches to solving the CPP, but one common method is to add additional edges to the graph to make it Eucerin (i.e., having an even number of edges at every vertex). Once the graph is Eulerian, the postman can traverse every edge exactly twice (once in each direction) and return to the starting point, thus ensuring that all streets are visited.

To add the necessary edges, we can use the following algorithm:

1. Identify all vertices in the graph with odd degree (i.e., an odd number of edges incident to the vertex).
2. For each pair of odd-degree vertices, odd an edge between them with a weight equal to the shortest distance between them.
3. Find the minimum-weight perfect matching on the set of new edges (i.e., the set of edges connecting the odd-degree vertices).
4. Add the matching edges to the original graph to create a new, Eulerian graph.

5. Find a Eulerian circuit on the new graph using any standard algorithm (e.g., Hierholzer's algorithm).
6. Traverse the Eulerian circuit, taking each edge exactly twice (once in each direction), to visit every edge in the original graph.

By following these steps, the postman can be sure to cover very street in the shortest possible distance.

CHAPTER THREE

Introduction About Duhok

Duhok province, one of the most beautiful cities in Kurdistan region in North Iraq. The original name of the city was Nuhadra, an Assyrian name, but later it was called Duhok after its geographical status. The city has a population of 1,772,367 mainly by Kurds with an Arab, Assyrian, and Armenian minority. Historically, there was a Jewish population in the region as well. The main religious groups are Muslims, Yazidis and Christians. Duhok is well-known for its wonderful and natural Landscape, and it is one of the most attracted area for tourist where the residents can sense the four seasons. Duhok Dam, Zaweta, Sarsang, Amedy, Akre, Zakho and Sulav are the most important places that attract tourists. Duhok, located in the Kurdistan Region of Iraq, has an economy driven by agriculture, trade, and tourism, with recent influences from oil discoveries. It borders Turkey and Syria, facilitating cross-border trade, and interacts economically with neighboring cities like Erbil and Sulaymaniyah within the Kurdistan Region. (HAMO, 2018) This is the map of Duhok.

Figure 6 map of Duhok city

Like many growing cities, Duhok has been facing traffic congestion issues due to a rapidly increasing population, the expansion of the city, and an increase in the number of cars on the roads. This has led to longer commuting times, increased air pollution, and reduced road safety.

To address this problem, the government of Duhok has implemented several measures, including building new roads and highways, improving public transportation, and introducing carpooling schemes. Additionally, the city has been working to improve its urban planning, such as developing pedestrian friendly zones and bike lanes.

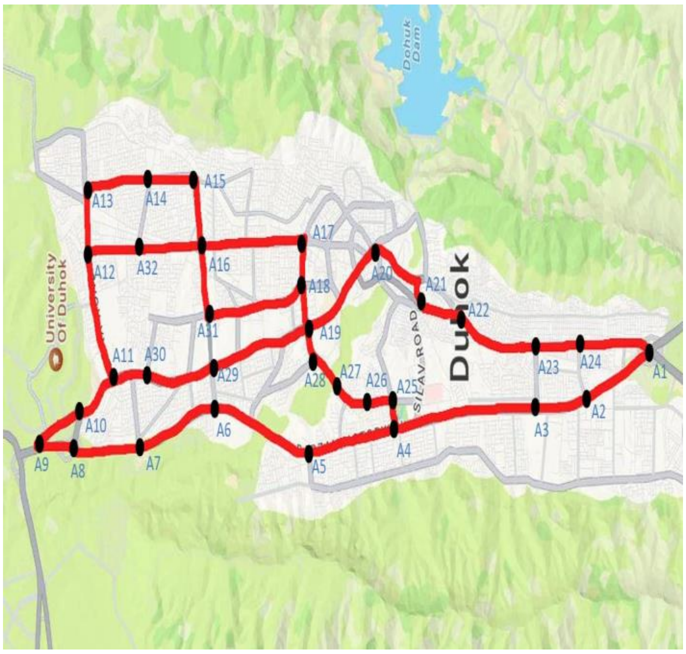
Despite these efforts, traffic congestion remains a significant issue in Duhok, and the city continues to work towards finding sustainable solutions to address the problem.

The Chinese postman problem is a mathematical problem that seeks to find the shortest possible route that visits every edge of a given undirected graph at least once. It is often used in transportation and logistics to optimize delivery routes and reduce travel time.

While the Chinese postman problem could potentially be used to address traffic congestion in Duhok by optimizing the routes of public transportation and delivery vehicles, it may not necessarily solve the problem of traffic congestion caused by private vehicles.

Private vehicles are often a major contributor to traffic congestion in cities, and reducing the number of cars on the roads can be challenging. Some potential solutions could include promoting the use of public transportation, encouraging carpooling and ridesharing, and improving infrastructure to support nonmotorized modes of transportation like cycling and walking.

Ultimately, a combination of solutions will likely be necessary to address traffic congestion in Duhok and other growing cities. Following example: In the graph 1 where each intersection is a vertex and each street is an edge with a length equal to the length of the street. The delivery guy needs to find the shortest path that visits every vertex (intersection) at least once and returns to the starting vertex (A_4).

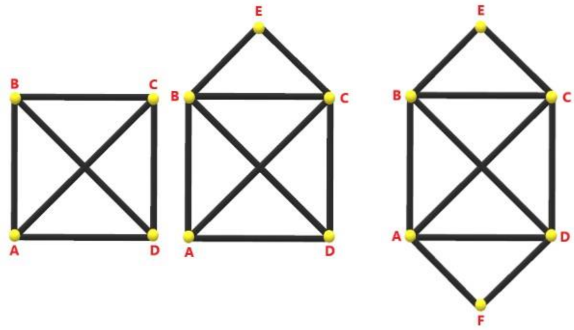


graph 1

Figure 7 Duhok city Main roads map

In the graph 1,the barzan highway as located in south of duhok city contain vertex A_1, A_2 , A_3, A_4, A_5, A_6, A_7,and A_8 .the zakho duhok highway contain vertex A_9 , A_{10} , A_{11} , A_{29} , A_{30} .the duhok erbil highway contain vertex A_{20}, A_{21}, A_{22}, A_{23}, A_{24}.the ganjina street contain vertex A_{25}, A_{26}, A_{27}, A_{28}.the ashty street contain vertex A_{17}, A_{18}, A_{19}.the zirka road way contain vertex A_{12}, A_{13}.the qazi mohammad street contain vertex A_{32}.the amad city contain vertex A_{15}, A_{16}, A_{31}.the chanal kurdu street contain vertex A_{14}.   the starting vertex is A_4.

To find the shortest path that visits every vertex at least once and returns to A_4 , we can use the Chinese postman algorithm, we will return to solving this actual problem later, but initially we will look at drawing various graphs. The Chinese postman is traversable graphs given below.



Graph 1 graph 2 graph 3

Figure 8 Graph 1 has four vertices with an odd number of edges connected to them. Graph 2 has two vertices with an odd number of edges connected to them, and three vertices with an even number of edges connected to them. Graph 3 has six vertices with an even number of edges connected to them.

we find:

- It is impossible to draw graph 1 without either taking the pen off paper or re-tracing an edge.
- We can draw graph 2, but only by starting at either A or D, in each case the path will end at the other vertex of D or A.
- Graph 3 can be drawn regardless of starting position and you will always return to the start vertex.

In order to establish the differences, we must consider the order of the vertices for each graph. The following when the order of all the vertices is even the graph is Travers able. When there are two odd vertices, we can draw the graph but the start and end vertices are different. When there are four odd vertices the graph can’t be drawn without repeating an edge.

vertex	order
A	3
B	3
C	3
D	3

Graph 1

vertex	order
A	3
B	4
C	4
D	3
E	2

Graph 2

vertex	order
A	4
B	4
C	4
D	4
E	2
F	2

Graph 3

We can use the following algorithm:

1. Identify all vertices in the graph with odd degree (i.e., an odd number of edges incident to the vertex).
2. For each pair of odd-degree vertices, odd an edge between them with a weight equal to the shortest distance between them.
3. Find the minimum-weight perfect matching on the set of new edges (i.e., the set of edges connecting the odd-degree vertices).
4. Add the matching edges to the original graph to create a new, Eulerian graph.
5. Find a Eulerian circuit on the new graph using any standard algorithm (e.g., Hierholzer's algorithm).
6. Traverse the Eulerian circuit, taking each edge exactly twice (once in each direction), to visit every edge in the original graph.

To find the shortest path in (figure 2.10), while each node represents a street intersecting, and each edge represents a street. the numbers on the edges represent the distance between two intersections for example, the edge between A_{25} andA_{30} has a distance of 1.6 kilo meter.

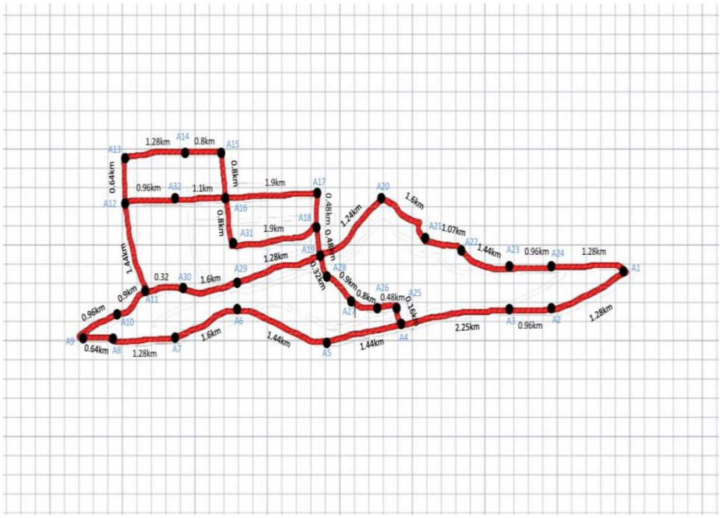


Figure 9 graph showing street map of Duhok

First, you need to identify the odd-degree vertices in the graph, which are A_4, A_{11}, A_{12}, and A_{18}, then we need to find a minimumweight matching of these vertices.

Since there are more than two odd-degree vertices, of each degree, there is more than one way to matching them, we can compute the weights of the edges connecting the odd-degree vertices:

$(A_4, A_{11}) = 5.86$,$(A_4, A_{12}) = 7.28$,$(A_4, A_{18}) = 2.82$,$(A_{11}, A_{12}) = 1.44$,

$(A_{11}$,$A_{18})$ =3.68 ,$(A_{12}$,$A_{18})$ =4.44 ,$(A_i, A_i) = \infty$ when $i =$ 4,11,12,18

Using an algorithm to find a minimum-weight. Perfect matching, we can select the following edges:

(A_4, A_{18}) ,(A_{11}, A_{12})

The total weight of this matching is:

$1.44+2.82+1.44+2.82 = 8.525$

In the matching that we have chosen for the odd-degree vertices in the graph, the total weight of the four edges connecting these vertices is 8.525 kilometers. This represents the minimum amount of additional distance that the delivery guy would need to travel in order to visit all edges in the graph at least once, and return to the starting point.

To calculate the total weight of the graph, we need to sum up the weights of all the edges in the graph. From the given graph, we can see that the weights of the edges are as follows:

$(A_1, A_2) = 1.28$, $(A_2, A_3) = 0.96$, $(A_3, A_4) = 2.25$, $(A_4, A_5) = 1.44$,

$(A_5, A_6) = 1.44$, $(A_6, A_7) = 1.6$, $(A_7, A_8) = 1.28$, $(A_8, A_9) = 0.64$,

$(A_9, A_{10}) = 0.96$, $(A_{10}, A_{11}) = 0.9$, $(A_{11}, A_{12}) = 1.44$, $(A_{12}, A_{13}) = 0.64$,

$(A_{13}, A_{14}) = 1.28$, $(A_{14}, A_{15}) = 0.8$, $(A_{15}, A_{16}) = 0.8$, $(A_{16}, A_{17}) = 1.9$,

$(A_{17}, A_{18}) = 0.48$, $(A_{18}, A_{19}) = 0.48$, $(A_{19}, A_{20}) = 1.24$ $(A_{20}, A_{21}) = 1.6$,

$(A_{21}, A_{22}) = 1.07$, $(A_{22}, A_{23}) = 1.44$, $(A_{23}, A_{24}) = 0.96$, $(A_{24}, A_1) = 1.28$,

$(A_4, A_{25}) = 0.16$, $(A_{25}, A_{26}) = 0.48$, $(A_{26}, A_{27}) = 0.8$, $(A_{27}, A_{28}) = 0.9$,

$(A_{19}, A_{28}) = 0.32$, $(A_{19}, A_{29}) = 1.28$,

$(A_{11}, A_{30}) = 0.32$, $(A_{29}, A_{30}) = 1.6$, $(A_{16}, A_{31}) = 0.8$, $(A_{18}, A_{31}) = 1.9$,

$(A_{16}, A_{32}) = 1.1$, $(A_{12}, A_{32}) = 0.96$

Adding up all these weights, we get:

$$1.28 + 0.96 + 2.25 + 1.44 + 1.44 + 1.6 + 1.28 + 0.64 + 0.96 + 0.9 + 1.44$$

$$+0.64+1.28+0.8+0.8+1.9+0.48+0.48+1.24+1.6$$
$$+1.07+1.44+0.96+1.28+0.16+0.48+0.8+0.9+0.32+1.28$$
$$+1.6+0.32+0.8+1.9+1.1+0.96=39.59$$

Therefore, the total weight of the graph is 39.59 kilometers.

After adding the edges from, the minimum-weight matching, the resulting graph is:

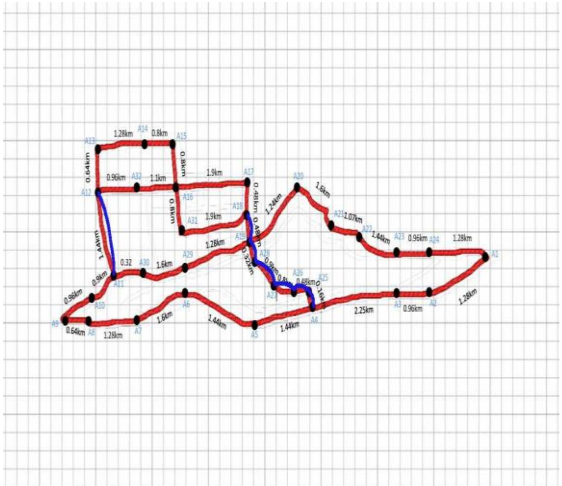


Figure 10 graph showing street map of Duhok *after adding edges to odd vertices*

the graph now has no odd-degree vertices, which means that it has an Eulerian circuit. The delivery guy can simply traverse the circuit to visit every edge in the graph exactly once and return to A_4. one possible Eulerian circuit is:

$$A_4 \to A_5 \to A_6 \to A_7 \to A_8 \to A_9 \to A_{10} \to A_{11} \to A_{12} \to A_{13} \to A_{14} \to A_{15} \to A_{16} \to A_{17} \to A_{18} \to A_{19} \to A_{20} \to A_{21} \to A_{22} \to A_{23} \to A_{24} \to A_1 \to A_2 \to A_3 \to A_4$$
$$\to A_{25} \to A_{26} \to A_{27} \to A_{28} \to A_{19} \to A_{18} \to A_{31} \to A_{16} \to A_{32}$$
$$\to A_{12} \to A_{11} \to A_{30} \to A_{29} \to A_{19} \to A_{28} \to A_{27} \to A_{26} \to A_{25} \to A_4$$

The total distance traveled is the sum of the lengths of the edges in the circuit:

$$Distance = A_4 A_5 + A_5 A_6 + A_6 A_7 + A_7 A_8 + A_8 A_9 + A_9 A_{10} + A_{10} A_{11}$$

$+ A_{11}A_{12} + A_{12}A_{13} + A_{13}A_{14} + A_{14}A_{15} + A_{15}A_{16} + A_{16}A_{17} + A_{17}A_{18}$

$+A_{18}A_{19} + A_{19}A_{20} + A_{20}A_{21} + A_{21}A_{22} + A_{22}A_{23} + A_{23}A_{24} + A_{24}A_{1} + A_{1}A_{2} + A_{2}A_{3} + A_{3}A_{4} + A_{4}A_{25} + A_{25}A_{26} + A_{26}A_{27} + A_{27}A_{28} + A_{28}A_{19} + A_{19}A_{18} + A_{18}A_{31} + A_{31}A_{16} + A_{16}A_{32} + A_{32}A_{12} + A_{12}A_{11} + A_{11}A_{30} + A_{30}A_{29} + A_{29}A_{19} + A_{19}A_{28} + A_{28}A_{27} + A_{27}A_{26} + A_{26}A_{25} + A_{25}A_{4}$

$Distance =$ 1.44 + 1.44+ 1.6 + 1.28 + 0.64 + 0.96 +0.9 + 1.44

+0.64+1.28+0.8+0.8+1.9+0.48+0.48+1.24+1.6+1.07

+1.44+0.96+1.28+1.28+0.96+2.25+0.16+0.48+0.8+0.9+0.32+0.48

+1.9+0.8+1.1+0.96+1.44+0.32+1.6+1.28+0.32+0.9+0.8+0.48+ 0.16=43.85

Therefore, the shortest path that visits every vertex at least once and returns to A_4 is 43.85 kilometers.

train system can potentially help alleviate traffic congestion in certain areas, as it provides an alternative mode of transportation that can reduce the number of vehicles on the road. However, it is important to consider several factors before implementing such a system.

Firstly, the cost of building and maintaining a train system can be substantial, and the government would need to carefully consider the financial implications of such a project. Additionally, the train system would need to be convenient and accessible to users, with stations located in areas where there is a high demand for transportation.

Another factor to consider is the existing infrastructure in the area. The train system would require a dedicated track, and the government would need to ensure there is sufficient space and resources to accommodate the new system without disrupting existing traffic flows or infrastructure.

Furthermore, the train system would need to be integrated with other modes of transportation, such as buses, taxis, and private vehicles, to ensure seamless and efficient travel for commuters. The government would need to consider how the train system would fit into the overall transport network and how it would impact the existing transport options available to the public.

Conclusion

The Chinese Postman Problem stands as a classic puzzle in graph theory, entailing the quest for the shortest closed path that traverses every edge within a given graph. This challenge extends its reach into practical domains like transportation, logistics, and network design To tackle the Chinese Postman Problem, we commence by identifying the odd-degree vertices within the graph. Subsequently, we seek a minimum-weight matching for these vertices, thereby introducing even-length paths into the graph, thereby morphing it into an Eulerian graph. With this transformation, we employ algorithms to unearth an Eulerian circuit within the adapted graph, effectively outlining the shortest closed path that touches every edge within the original graph. Various algorithms, such as Hierholzer's or Edmonds', serve as tools for computing solutions to this problem. However, it's worth noting that some graphs may not yield a solution due to an odd number of odd-degree vertices, impeding their pairing in a minimum-weight matching The Chinese Postman Problem emerges as a captivating and consequential enigma within graph theory, bearing significant implications across scientific and engineering realms.

In evaluating paths or routes, the weight of a graph emerges as a crucial measure, indicative of efficiency and cost. Specifically, within the context of the Chinese Postman Problem, graph weight mirrors the cumulative distance a delivery individual would traverse to visit each edge at least once and return to the starting point Understanding the total weight of a graph enhances comprehension of its structure and attributes, facilitating informed decisions regarding navigation and optimization with in the graph.

In conclusion, while introducing a train system presents a promising avenue to alleviate traffic congestion in specific locales, its implementation demands meticulous planning, considering factors such as cost, accessibility, infrastructure, and integration with existing transportation systems.

References

1. agarwal, l., agarwal, s., niharika & agarwal, p., 2019. *history of graph theory.* [Online] Available at: https://prinsli.com/history-of-graph-theory/ [Accessed 13 November 2024].

2. Bondy, J. A., 2006. *Graph Theory with Application.* third edition ed. New Delhi: NEW AGE INTERNATIONAL (P) LIMITED .

3. Hamo, D. S., Malaika, M., & Namo, Q. (2018). Planning to control traffic congestion in Duhok University campus. Journal of University of Duhok, 21(1), 97-110. https://doi.org/10.26682/sjuod.2018.21.19

4. Mohtashim, M., 2019. *Eulerian Graphs.* [Online]
Available at: https://www.tutorialspoint.com/eulerian-graphs#
[Accessed 28 march2024]

5. Mrs.M.durgadevi & Ms.Ch, S. H., 2018. use of the graph theory in transportation problems and different networks. *applied science and computations*, 5(12), pp. 32-40.

6. naduvath, s., 2017. *Lecture Notes on Graph Theory.* first ed. kerala,inia: Centre for Studies in Discrete Mathematics.

7. Sinha, R., 2015. *connectivity of grapgs.* [Online] Available at: https://www.slideshare.net/aswathymaths/a-study-on-connectivity-in-graphtheory-june-18-123e [Accessed 1 april 2024].

8. stephanie, 2019. *Graph theory: Definitions for common terms.* [Online] Available at: //www.statisticshowto.com/graph-theory/ [Accessed 29 january 2024].

پوختە

زۆرجار گرافەکان بەکاردەهێنرێن بۆ مۆدێلکردنی کێشەکانی باشکردن. لە پرۆسەی باشکردندا ئامانج ئەوەیە کە باشترین چارەسەر لە کۆمەڵێک چارەسەری ئەگەری بدۆزرێتەوە. گرافەکان نوێنەرایەتییەکی بینراوی پەیوەندییەکانی نێوان یەکەکان دابین دەکەن، ئەمەش وایان لێدەکات کە بەسوود بن بۆ مۆدێلکردنی سیناریۆکانی باشکردنی ژیرانە وەک دۆزینەوەی کورتترین ڕێگا، کەمکردنەوەی تێچوونەکان، زۆرترین کارایی، یان باشکردنی تەرخانکردنی سەرچاوەکان. زۆرجار ئەلگۆریتمەکانی باشکردن لەسەر گرافەکان کاردەکەن بۆ دۆزینەوەی چارەسەری گونجاو بە شێوەیەکی کارا بە تێپەڕاندنی گرێ و لێوارەکان لەسەر بنەمای هەندێک پێوەر یان سنووردارکردن. باشکردن، بە تایبەت لە تیۆری گرافدا، ڕێبازێکی سیستماتیکی پێشکەش دەکات بۆ چارەسەرکردنی پرسە ئاڵۆزەکانی شار وەک قەرەباڵغی هاتوچۆ. دۆهوک وەک زۆرێک لە شارەکان بەهۆی زیادبوونی ژمارەی دانیشتووان و زیادبوونی هاتوچۆی ئۆتۆمبێلەکانەوە ڕووبەڕووی ئاستەنگ دەبێتەوە، کە دەبێتە هۆی قەرەباڵغی و دواکەوتن و پیسبوون: ئەم توێژینەوەیە چارەسەرێک بۆ قەرەباڵغی هاتوچۆ لە دۆهوک پێشنیار دەکات، بە بەکارهێنانی بنەماکانی کێشەی پۆستەچی چینی ((CPPو داوای مۆدێرنیزاسیۆنی سیستماتیکی دەکات . ڕاهێنان دەکات. بە بەکارهێنانی میتۆدۆلۆژیای CPP، توێژینەوەکە ئامانجیەتی ڕێگاکانی ئۆتۆمبێلی گونجاو دەستنیشان بکات، قەرەباڵغی کەم بکاتەوە و ئەزموونی گەشتکردن بۆ دانیشتووان باشتر بکات

ملخص

غالبًا ما تُستخدم الرسوم البيانية لنمذجة مشكلات التحسين. في عملية التحسين، الهدف هو إيجاد أفضل حل من بين مجموعة من الحلول الممكنة. توفر الرسوم البيانية تمثيلاً مرئيًا للعلاقات بين الكيانات، مما يجعلها مفيدة لنمذجة سيناريوهات التحسين الحكيمة مثل العثور على أقصر مسار، أو تقليل التكاليف، أو زيادة الكفاءة إلى الحد الأقصى، أو تحسين تخصيص الموارد. غالبًا ما تعمل خوارزميات التحسين على الرسوم البيانية للبحث عن الحلول المثلى بكفاءة من خلال تجاوز العقد و حواف بناء على بعض المعايير أو القيود. يقدم التحسين، وخاصة في نظرية الرسم البياني، نهجا منظما لمعالجة القضايا الحضرية المعقدة مثل الازدحام المروري. تواجه دهوك مثل العديد من المدن تحديات بسبب النمو السكاني وزيادة حركة مرور المركبات، مما يؤدي إلى الازدحام والتأخير والتلوث: تقترح هذه الورقة حلاً للازدحام المروري في دهوك، وذلك باستخدام مبادئ مشكلة ساعي البريد الصيني (CPP) والدعوة إلى التحديث المنهجي. وباستخدام منهجية CPP يدرب. تهدف الدراسة إلى تحديد المسارات المثلى للمركبات وتقليل الازدحام ، وتحسين تجربة السفر للمقيمين