Finding the Shortest Route in Sagaing Region Network

Khine Yin Myint[1]

Abstract

In this paper, an algorithm related to the shortest route problems is discussed. Then, the shortest route from Sagaing Township to Banmauk Township in Sagaing Region is computed by using Dijkstra's algorithm. Finally, the minimum distance of travelling along the ten townships in Sagaing Region to traverse each route exactly once is calculated.

Keywords: Network, Algorithm, Shortest Route, Transport, Euler circuit.

Introduction

Networks have always been important in transportation and telecommunication. In solving problems in transportation networks, Graph Theory in mathematics is a fundamental tool. Dijkstra's algorithm is a search algorithm that finds the quickest ways from one point to another in a network. Finding the shortest route in a network is very important in transportation. It can be used as a real-time application for new technologies such as computer network and map related system.

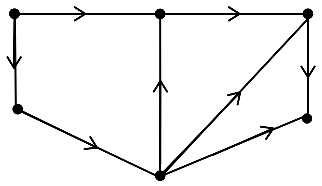
The purpose of this paper is to find the shortest route from one place to another place by reducing the distance, time and cost for the transportation services. The required datas (the distances between two townships) are acquired from google map.

Preliminaries

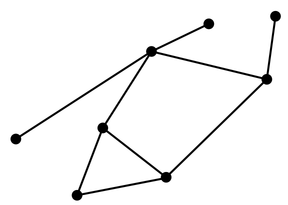
A **graph** (undirected graph) consists of a non-empty set of points (vertices) and a set of lines (edges) connecting the vertices. A **directed graph** is a graph in which the edges have a direction. An edge joining a vertex to itself is called a **loop**. If there are two or more edges joining two distinct vertices, then the graph is said to have **multiple edges**. A graph that has no loops but may have multiple edges is called a **multigraph**. A graph having neither loops nor multiple edges is called a **simple graph**. A **weighted graph** is a graph in which a number (the weight) is assigned to each edge. A **walk** in a graph is a sequence of vertices and edges. A walk is said to be **closed** if the first and last vertices are the same. A **route** (path) is a walk with distinct vertices. A **circuit** is a path that begins and ends at the same vertex. A graph is said to be **connected** if any two vertices are connected by a route. The number of edges linked to a vertex is called the **degree** of that vertex. An **Euler circuit** is a circuit that uses every edge of a graph exactly once. A graph is **Eulerian** if it contain an Euler circuit.

A **network** is a set of objects (points) that are connected together. A **connected network** is such that every two distinct vertices are linked by at least one route.

As an example, a directed network and undirected network are shown in Figure 1.



(a) A directed network



(b) Undirected network

Figure 1. A directed network and undirected network.

[1] Dr., Associate Professor, Department of Mathematics, Sagaing University

A **transportation network** is a simple, weight, directed graph with the following properties.
(i) There is one source with no incoming edges.
(ii) There is one destination with no outgoing edges.
(iii) The weight of the directed edges is a non-negative number.

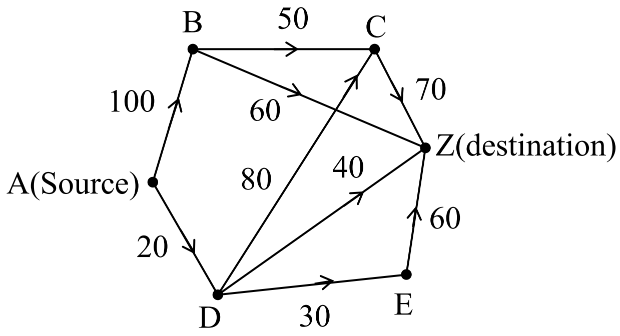


Figure 2. A transport network.

In a transportation network, each edge has a weight that can represent the real-life quantities such as distance, time, cost, data, etc.

Algorithm Dijkstra

Dijkstra's Algorithm is a search algorithm that computes the shortest route problem (a problem of finding the shortest route from a starting point to a final destination) for a graph. The algorithm is as follows:

step(1) The start vertex is chosen as the permanent vertex. Designate this vertex as the current vertex.

step(2) From this start vertex, consider the distances of the neighbor vertices. Determine the new vertex that has the smallest distance. Designate this new vertex as the current vertex.

step(3) Replace the "step 2" until the destination vertex is reached.

Implementation

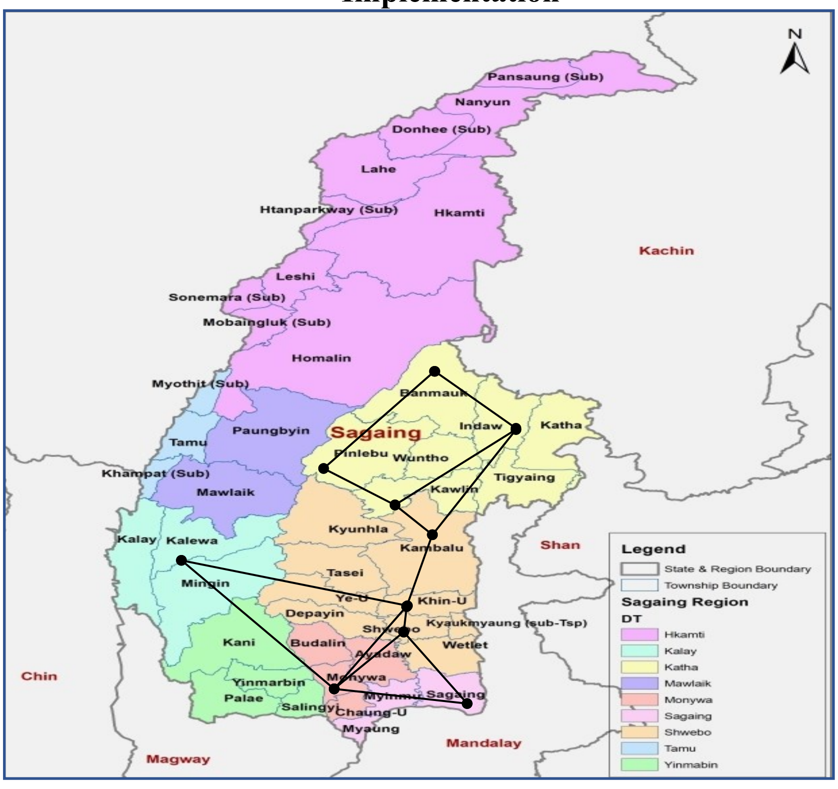


Figure 3. Sagaing Region Map.

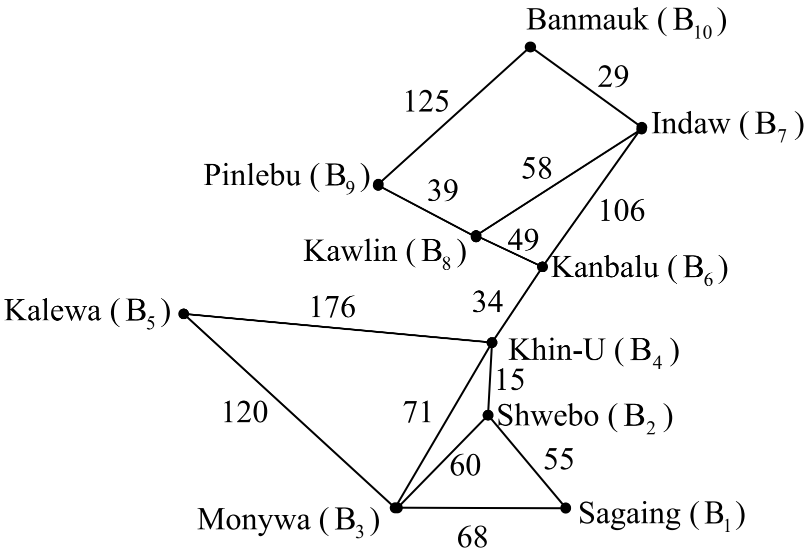


Figure 4. Representation of outpute route in Sagaing Region.

The distance between two vertices (two townships) is measured in miles. The start vertex is B_1 (Sagaing Township). The distance for all other vertices from vertex B_1 is assumed as ∞. From B_1, consider the distances of neighbor vertices B_2 (Shwebo Township) and B_3 (Monywa Township).

The result is summarized in Table 1.

Table 1.

Townships	B_2	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}
Distances	55	68	∞	∞	∞	∞	∞	∞	∞

B_1 —— B_2

In Table 1, the smallest distance "a" is 55 and the corresponding vertex "v" is B_2. Since the vertex B_2 is the neighbor of the start vertex B_1, thicken the edge (B_1, B_2). For vertex B_2, vertex B_3 (Monywa Township) and vertex B_4 (Khin-U Township) are the unselected neighbors.

Then, $a + d_{B_2B_4} = 55 + 15 = 70$, where $d_{B_2B_4}$ is the distance from vertex B_2 to the vertex B_4. The distance to vertex B_4 is updated as 70.

$$a + d_{B_2B_3} = 55 + 60 = 115.$$

The result is summarized in Table 2.

Table 2.

Townships	B_3	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}
Distances	68	70	∞	∞	∞	∞	∞	∞

B_1 —— B_3

In Table 2, the smallest distance "a" is 68 and the corresponding vertex "v" is B_3. Since B_3 is the neighbor of the start vertex, thicken the edge (B_1, B_3). For vertex B_3, vertex B_5 (Kalewa Township) and vertex B_4 (Khin-U Township) are the unselected neighbors.

Then $a + d_{B_3B_4} = 68 + 71 = 139$,

$a + d_{B_3B_5} = 68 + 120 = 188.$

The distance to vertex B_5 is updated as 188. The result is summarized in Table 3.

Table 3.

Townships	B_4	B_5	B_6	B_7	B_8	B_9	B_{10}
Distances	70	188	∞	∞	∞	∞	∞

B_2 — B_4

In Table 3, the smallest distance "a" is 70 and the corresponding vertex "v" is B_4. The vertex selected in that iteration is B_2. So thicken the edge (B_2, B_4). For vertex B_4, vertex B_5 (Kalewa Township) and vertex B_6 (Kanbalu Township) are the unselected neighbors.

Then $a + d_{B_4B_5} = 70 + 176 = 246$,

$a + d_{B_4B_6} = 70 + 34 = 104$.

The distance to vertex B_6 is updated as 104. The result is summarized in Table 4.

Table 4.

Townships	B_5	B_6	B_7	B_8	B_9	B_{10}
Distances	188	104	∞	∞	∞	∞

B_4 — B_6

In Table 4, the smallest distance "a" is 104 and the corresponding vertex "v" is B_6. The vertex selected in that iteration is B_4. So, thicken the edge (B_4, B_6). For vertex B_6, vertex B_7 (Indaw Township) and vertex B_8 (Kawlin Township) are the unselected neighbors.

Then $a + d_{B_6B_7} = 104 + 106 = 210$.

The distance to vertex B_7 is updated as 210.

Then $a + d_{B_6B_8} = 104 + 49 = 153$.

The distance to vertex B_8 is updated as 153. The result is summarized in Table 5.

Table 5.

Townships	B_5	B_7	B_8	B_9	B_{10}
Distances	188	210	153	∞	∞

B_6 — B_8

In Table 5, the smallest distance "a" is 153 and the corresponding vertex "v" is B_8. The vertex selected in that iteration is B_6. So, thicken the edge (B_6, B_8). For vertex B_8, vertex B_7 (Indaw Township) and vertex B_9 (Pinlebu Township) are the unselected neighbors.

Then $a + d_{B_8B_7} = 153 + 58 = 211$,

$a + d_{B_8B_9} = 153 + 39 = 192$.

The distance to vertex B_9 is updated as 192. The result is summarized in Table 6.

Table 6.

Townships	B_5	B_7	B_9	B_{10}
Distances	188	210	192	∞

B_3 — B_5

In Table 6, the smallest distance “a” is 188 and the corresponding vertex “v” is B_5. The vertex selected in that iteration is B_3. So, thicken the edge (B_3, B_5). For vertex B_5, vertex B_4 is the selected vertex. From vertex B_4, vertex B_6 is the selected vertex. From vertex B_6, vertex B_8 is the selected vertex and vertex B_7 is the unselected vertex.

Then $a + d_{B_6B_7} = 188 + 106 = 294.$

The result is summarized in Table 7.

Table 7.

Townships	B_7	B_9	B_{10}
Distances	210	192	∞

B_8 —— B_9

In Table 7, the smallest distance “a” is 192 and the corresponding vertex “v” is B_9. The vertex selected in that iteration is B_8. So thicken the edge (B_8, B_9). For vertex B_9, vertex B_{10} (Banmauk Township) is the unselected neighbor.

Then $a + d_{B_9B_{10}} = 192 + 125 = 317.$

The result is summarized in Table 8.

Table 8.

Townships	B_7	B_{10}
Distances	210	317

B_6 —— B_7

In Table 8, the smallest distance “a” is 210 and the corresponding vertex “v” is B_7. The vertex selected in that iteration is B_6. So thicken the edge (B_6, B_7). For vertex B_7, vertex B_{10} is the unselected neighbor.

Then $a + d_{B_7B_{10}} = 210 + 29 = 239.$

The result is summarized in Table 9.

Table 9.

Townships	B_{10}
Distances	239

B_7 —— B_{10}

In Table 9, the smallest distance “a” is 239 and the corresponding vertex “v” is B_{10}. The vertex selected in that iteration is B_7. So thicken the edge (B_7, B_{10}). Vertex B_{10} is the destination. Therefore the shortest route is $B_1 — B_2 — B_4 — B_6 — B_7 — B_{10}$. That is, the shortest route from Sagaing Township to Banmauk Township is Sagaing — Shwebo — Khin-U — Kanbalu — Indaw — Banmauk and the corresponding distance is 239 miles. The result is presented in Figure 5.

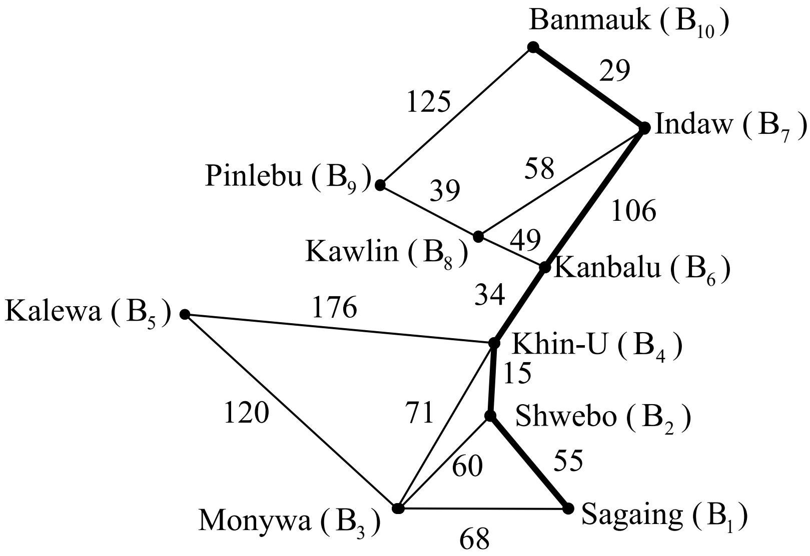


Figure 5. Shortest route (Bold edges) from Sagaing Township to Banmauk Township in Sagaing Region.

Theorem

Connected graph G is an Eulerian graph if and only if each vertex of G has even degree.

Proof: See [Bondy, J.A. and Murty, U.S.R., 1976].

The Route Inspection in Sagaing Region

We wish to travel from the starting Sagaing Township (B_1) to Shwe Bo Township (B_2), Monywa Township (B_3), Khin-U Township (B_4), Kalewa Township (B_5), Kanbalu Township (B_6), Indaw Township (B_7), Kawlin Township (B_8), Pinlebu Township (B_9) and Banmauk Township (B_{10}) respectively. We want to traverse along every routes in the following figure, using the shortest route possible.

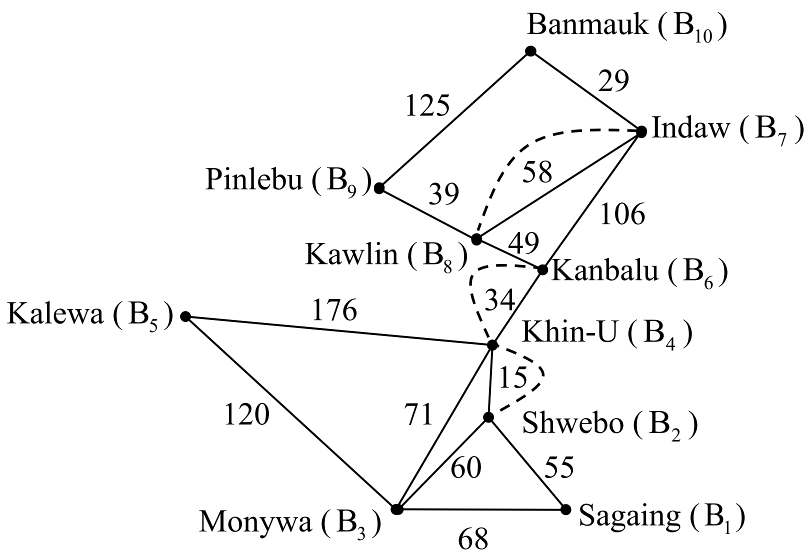


Figure 6. Representation of Route Inspection in Sagaing Region.

We desire to travel along all 14 routes and return to Sagaing Township (B_1) without travelling along the same route twice.

If the graph has an Euler circuit, it is possible to traverse the graph when beginning and ending at the same point. If there is no such Euler circuit, the smallest number of copies of edges are added to obtain a multigraph which has an Euler circuit. By adding smallest number of edges on this routes, the minimum possible distance over our desire can be computed.

The odd degree vertices are B_2, B_6, B_7 and B_8. There is only one way to pair these odd degree vertices. The shortest route of joining B_1 to B_{10} is to use the routes B_2B_4, B_4B_6 and B_7B_8 which is a total of 107 miles. The sum of all the routes in the original network which is 1005 miles. Hence the length of the minimum distance is 1112 miles.

One possible route for this minimum length is $B_1, B_3, B_2, B_4, B_3, B_5, B_4, B_6, B_7, B_8, B_7, B_{10}, B_9, B_8, B_6, B_4, B_2, B_1$. That is, from the starting Sagaing Township, the minimum distance of traversing each route exactly once is Sagaing — Monywa— Shwebo — Khin-U— Monywa — Kalewa — Khin-U — Kanbalu — Indaw — Kawlin — Indaw — Banmauk — Pinlebu — Kawlin — Kanbalu — Khin-U — Shwe Bo— Sagaing and its minimum distance is 1112 miles. Many other possible routes can be found.

We observe that we traverse each route exactly once and return to the any starting township with minimum distance.

Conclusion

The feasibility of using Dijkstra's algorithm accommodate user's preferences for a public transportation network is discussed. Dijkstra's algorithm helps to find the shortest route between two points in a network. This paper intends to deal with the difficulities faced when the user wants to visit the desired target in the given road map. This paper supports the travelling process by giving the shortest distance, minimum cost and minimum time between desired source and destination.

Acknowledgements

I would like to thank Dr Soe Myint Aye, Rector, and Dr Phyu Phyu Myint Pro-Rector, Sagaing University for their permission. I would like to thank gratefully Dr Cho Cho, Professor, Head of Mathematics Department, Dr Than Than Hlaing, Professor, Head of Department of Mathematics and Dr Ni Ni Khaing, Professor, Department of Mathematics, Sagaing University.

References

Arockiamary, A. and Pravina, G., (2014) **"Application of Graph Theory to Find Shortest Path of Transportation Problem"**, Volume: 03, 2014, Pages: 833-837, www.academia.edu.

Bondy, J. A. and Murty, U. S. R., (1976) **"*Graph Theory with Applications*"**, Macmillan Press Ltd., London, 1976.

https // www.google map.com

Map of Sagaing Region [Online]. Myanmar Information Management Unit, https://themimu.info.