INTERNATIONAL JOURNAL OF ADVANCE RESEARCH AND DEVELOPMENT
(Volume 5, Issue 5)
Available online at: www.ijarnd.com

Optimal route-finding system for Myanmar Country

Aye Aye Maw
Faculty of Information Science, University of Computer Studies, Mandalay, Myanmar

ABSTRACT

Traffic congestion is becoming a serious problem in more and more modern cities. So, this system is proposed to solve the traveler's facing problems in finding the shortest path from one place to another place. Encouraging the traveling salesperson is one of the most effective and economical ways to reduce the ever-increasing congestion problem on the roads. This system is proposed as the optimal route-finding system for Myanmar country. The proposed system is able to find the optimal route from the starting location to the intended location in short time by using Prim algorithm.

Keywords— *Optimal Route, Graph, Prim*

1.INTRODUCTION

As the technology grows rapidly, many people take a great interest in computer and then computer base methods are increasingly used to improve the transportation services. In computer science, graph theory plays an important role because it provides an easy and systematic way to find the optimal route for the transportation services. Methods in graph theory consist of breadth-first search, depth-first search, shortest path and minimum spanning tree algorithms. Among them, this system proposes shortest path and minimum spanning tree algorithms.

To help the traveling salesperson, this system uses the Prim algorithm. Prim algorithm is the minimum spanning tree algorithm. In this system, this algorithm is used for solving transportation problems.

This system is implemented by using locations within Myanmar country as the vertices of an undirected Graph. In this system, the associated distances between each location are represented as weight of the edges of the graph. This system suggests the shortest distance and minimum cost of the paths for the user's convenience.

2. RELATED WORK

P. W. Eklund, S. Kirkby and S. Pollitt [1] proposed the implementation of Dijkstra's classic double bucket algorithm for path finding in connected networks. This work reports on a modification of the algorithm embracing both static and dynamic heuristic components and multiple source nodes. The modified algorithm is applied in 3D Spatial Information System (SIS) for routing emergency service vehicles.

M. Tommiska and J. Skytta [2] discussed the suitability of computing architectures of different network routing methods. As an example of the speedup offered by reconfigurable logic, the implementation of Dijkstra's shortest path routing algorithm is presented and its performance is compared to a microprocessor-based solution.

D. Eppstein [3] presented Prim's algorithm that is faster on dense graph in which the number of edges is close to the maximal number of edges while Kruskal's is better on a graph with only a few edges called sparse graph. Prim's algorithm is the method of choice for dense graphs that all the other methods perform within a small constant factor of the best possible for graphs of intermediate density and that Kruskal's method essentially reduces the problem to sorting for sparse graphs.

3. GRAPHY THEORY

Graphs are among the most ubiquitous models of both natural and human-made structures. They can be used to model many types of relations and process dynamics in physical, biological and social systems. Many problems of practical interest can be represented by graphs [4]. A graph G, consists of two sets: V and E. V is a finite, nonempty set of vertices. E is a set of pairs of vertices; these pairs are called edges. V(G) and E(G) will represent the sets of vertices and edges, respectively, of graph. G = (V, E) to represent a graph [3, 4]. There are two types of graphs: (1) Directed graphs and (2) Undirected graphs.

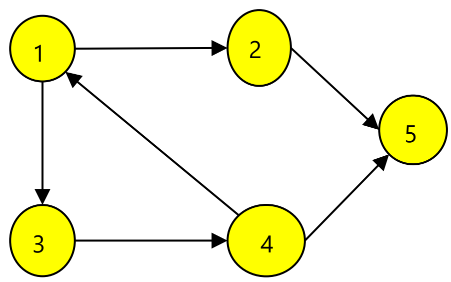

Fig. 1: Directed Graph

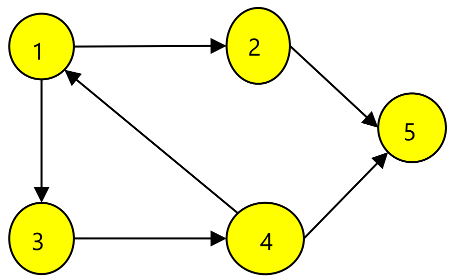

Fig. 2: Undirected Graph

In the directed graph G = (V,E), E is composed of ordered pairs of vertices; i.e. the edges have direction and point from one vertex to another. In the undirected graph G = (V, E), E is composed of unordered pairs of vertices; i.e. the edges are bidirectional. Directed graph is shown in Figure 1. Undirected graph is shown in Figure 2.

4. MINIMUM SPANNING TREE

In the graph theory, four searching methods are breadth-first search, depth-first search, minimum spanning tree and shortest path. Spanning trees are important in parallel and distributed computing, as a way of maintaining communications between a set of processors. A minimum spanning tree (MST) is a spanning tree of an edge weighted graph having lowest total weight among all possible spanning trees. The generic minimum spanning tree algorithm maintains an acyclic sub-graph F of the input graph G, which an intermediate spanning forest will be called. F is a sub-graph of the minimum spanning tree of G, and every component of F is a minimum spanning tree of its vertices.

The intermediate spanning forest F induces two special types of edges. An edge is useless if it is not an edge of F, but both its endpoints are in the same component of F. For each component of F, associate a safe edge - the minimum-weight edge is associated with exactly one endpoint in that component. Different components might or might not have different safe edges. Some edges are neither safe nor useless. All minimum spanning tree algorithms contain every safe edge and no useless edges.

A minimum-cost spanning tree is a spanning tree that has the lowest cost. Minimum-cost spanning tree of a graph defines the cheapest subset of edges that keeps the graph in one connected component. To construct minimum-cost spanning trees, it uses a least-cost criterion. This solution must satisfy the following constraints:

- Use only edges within the graph.
- Use exactly n-1 edges.
- May not use edges that produce a cycle [6].

Minimum spanning tree algorithms consists of Kruskal's algorithm, Prim's algorithm and Boruvka's algorithm [7].

4.1. Prim Algorithm

Prim's algorithm begins by choosing any edge with smallest weight, putting it into the spanning tree. Successively add to the tree edges of minimum weight that are incident to a vertex already in the tree and not forming a simple circuit with those edges already in the tree. Stop when n – 1 edges have been added.

Prim's algorithm constructs the minimum-cost spanning tree edge by edge. However, in all times during the algorithm, the set of selected edges forms a tree. Prim's algorithm begins with a tree T that contains a single vertex. This vertex can be any of the vertices in the original graph. Then a least–cost edge (u, v) is added to T such that T U {(u, v)} is also a tree. This edge-addition step is repeated until T contains n-1 edges. Edge (u, v) is always such that exactly one of u and v is in T. The procedure is as follows:

Begin
TV = {0};
for (*T*=0; *T* contains fewer than *n-1* edges; add (*u*, *v*) to *T*)
{
Let (*u*, *v*) be a least-cost edge such that ***u* Є *TV*** and ***u* Є *TV***;
If (there is no such edge) **break**;
Add ***v*** to ***TV***;
}
If (T contains fewer than n-1 edges)
cout << "no spanning tree"<< endl;
End

Prim's algorithm also provides for the possibility that the input graph may not be connected. In this case there is no spanning tree. The minimum spanning tree is built by connecting each edge. A new edge is found by attaching to a single growing tree at each step.

Prim algorithm can give probably optimal solutions. Prim algorithm is perhaps the simplest MST algorithm because the number of edges is closed the maximal number of edges on dense graph [5].

5. PROPOSED SYSTEM DESIGN

At first of the system, the user can extract the desired road map from the database. If the user is the traveling salesperson, this system finds the minimum cost road path by using Prim algorithm. After finding the minimum cost route path, this system produces the minimum cost route as the optimal path for the travelling salesperson. Proposed system design is shown in Figure 3.

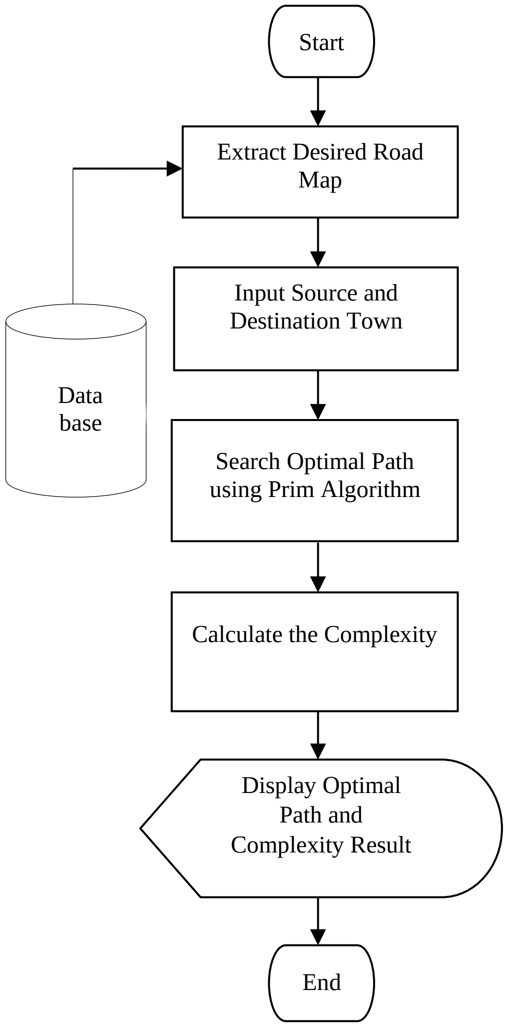

Fig. 3: System Flow Diagram

5.1. Calculation Step of Prim Algorithm

As a sample, this system searches the shortest path from the source town "ေရနံေခ်ာင္းၿမိဳ႕" to the destination town "ေအာင္လံၿမိဳ႕" within Magway division. In this sample, there are eight processing steps. Each step is shown in Figure 4, 5, 6, 7, 8, 9 and 10.

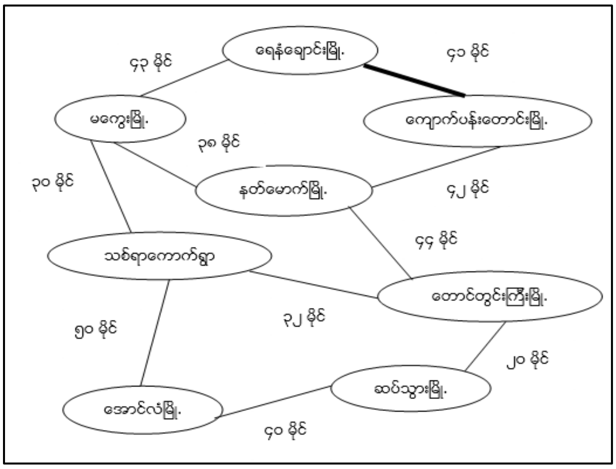

Fig. 4: Step 1 of Prim Algorithm

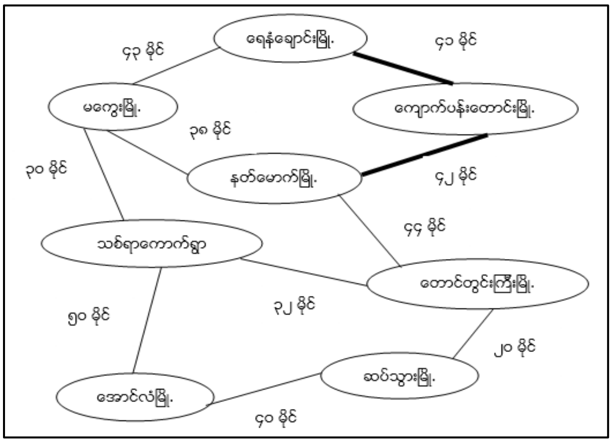

Fig. 5: Step 2 of Prim Algorithm

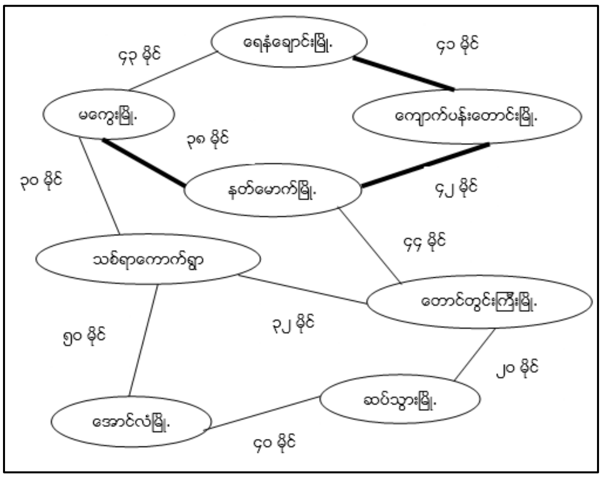

Fig. 6: Step 3 of Prim Algorithm

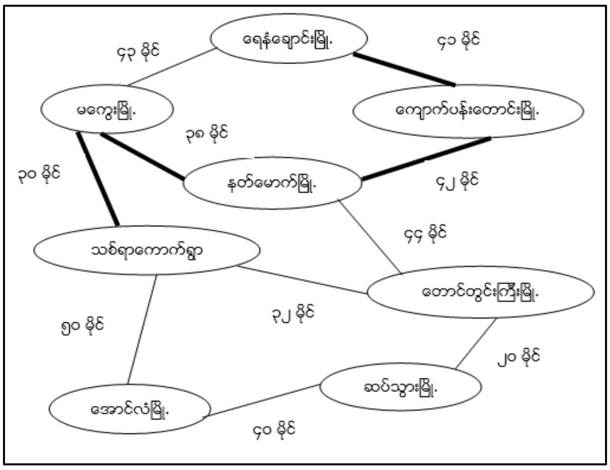

Fig. 7: Step 4 of Prim Algorithm

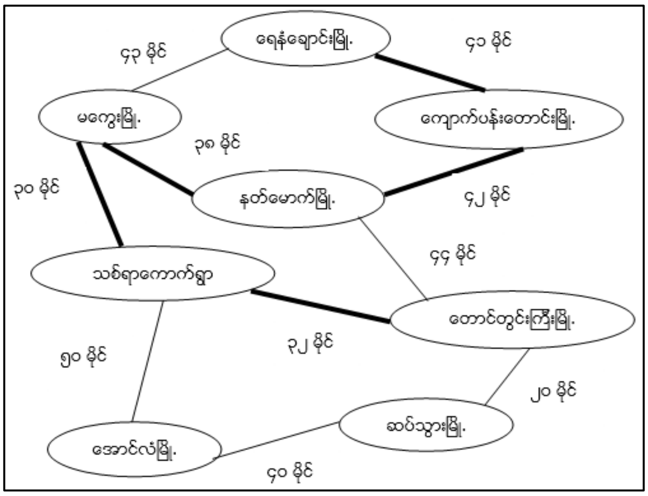

Fig. 8. Step 5 of Prim Algorithm

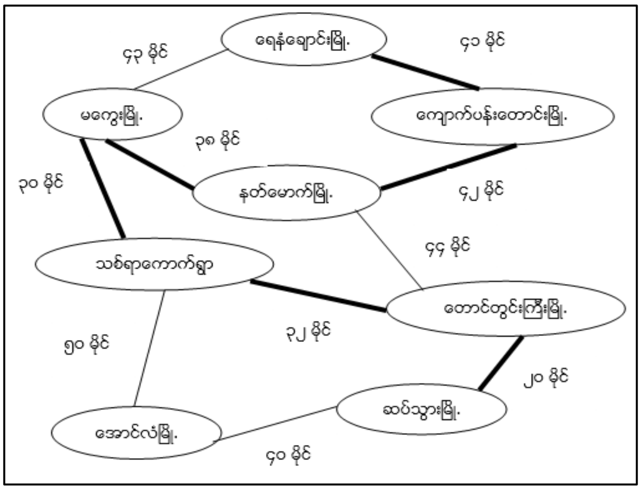

Fig. 9. Step 6 of Prim Algorithm

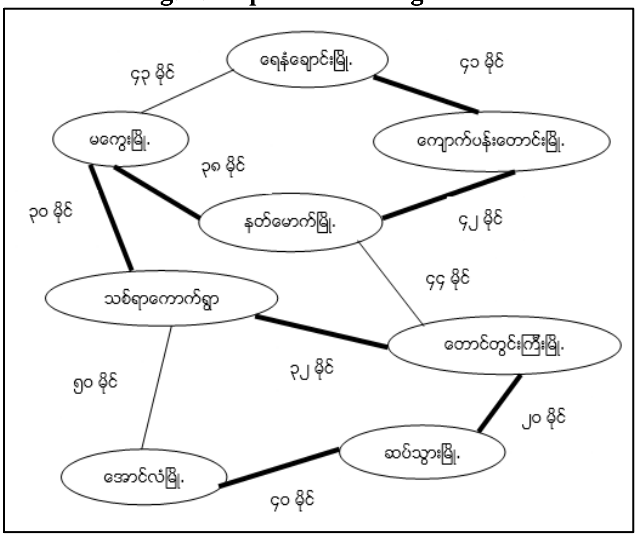

Fig. 10. Step 7 of Prim Algorithm

5.2. Experimental results of the system

This system measures the complexity of Prim algorithm. The complexity of Prim algorithm is O (|E| * log |V|). E is number of edges and V is number of vertices.

Prim's Algorithm

Travelling Route:

ရေနံချောင်းမြို့ ---> ကျောက်ပန်းတောင်းမြို့ ---> နတ်မောက်မြို့ ---> မကွေးမြို့ ---> သစ်ရာကောက်ရွာ ---> တောင်တွင်းကြီးမြို့ ---> ဆပ်သွားမြို့ ---> အောင်လံမြို့

Distance: 0+41+42+38+30+32+20+40=243 Miles
Complexity: 6.32156

6. CONCLUSION

This system intends to deal with the difficulties faced when the user wants to visit the desired cities in the given road map. In this system, minimum spanning tree algorithm is used to extract the optimal route within Magway division. Moreover, this system analyzed the performance of this algorithm by producing the optimal route. For the unfamiliar public users, the system can help choosing the path from many existing routes as they wish.

7. REFERENCES

[1] P. W. Eklund, S. Kirkby and S. Pollitt, "A Dyanmic Multi-source Dijkstra's Algorithm for Vehicle Routing", Australian and New Zealand Conference on Intelligent Information Systems (ANZIIS), 1996.

[2] M. Tommiska and J. Skytta, "Dijkstra's Shortest Path Routing Algorithm in Reconfigurable Hardware", in Proceedings of 11th International Conference on Field-Programmable Logic and Applications (FPL-2001), August 2001, PP. 653-657.

[3] D. Eppstein, "Representing all Minimum Spanning Trees with Applications to Counting and Generation", December 15, 1995.

[4] R.Diestal, "Graph Throry", Springer-Verlag, New York, 1997.

[5] Ellis Horowitz, Sartaj Sahni, Dinesh Mehta, "Fundamentals of Data Structures in C++", Suneel Galgotia for Galgotia Publications Pvt.Ltd, 5, Ansari Road, Darya Ganji, New Delhi-110002, Volume 2, 2003.

[6] R. G. Gallager, P. A. Humblet; P. M. Spira, "A distributed algorithm for minimum-weight spanning trees", ACM Transactions on Programming Languages and Systems 5 (1): 66–77, 1983.

[7] David Joyner, Minha Van Nguyen, Nathann Cohen, "Algorithmic Graph Theory", 19 March 2010.

