Optimal Route Identification on Complex Road Network by Using Advanced Dijkstra's Algorithm

K-zin Phyo

University of Computer Studies, Yangon

November, 2019

Optimal Route Identification on Complex Road Network by Using Advanced Dijkstra's Algorithm

K-zin Phyo

University of Computer Studies, Yangon

A thesis submitted to the University of Computer Studies, Yangon in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

November, 2019

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

1.11.2019

………………………………………… ……………………………………………

Date K-zin Phyo

ACKNOWLEDGEMENTS

First of all, I would like to thank His Excellency, the Minister for the Ministry of Education, for supporting full facilities during the Ph. D. Course at the University of Computer Studies, Yangon.

I would like to express very special thanks to Dr. Mie Mie Thet Thwin, Rector of the University of Computer Studies, Yangon, for allowing me to develop this thesis and giving me general guidance during period of my study.

I would like to say a very big thank to my supervisor, Dr. Myint Myint Sein, Professor, University of Computer Studies, Yangon, for her excellent guidance, caring, patience, and providing me with excellent ideas for doing research. I have been extremely lucky to have a supervisor who cared so much about my work. Without her guidance and constant feedback, this PhD would not have been achievable. I will always remember her for being a mentor to me.

I would like to extend my special appreciation to Dr Mi Mi Nge, Managing Director, Realistic Infotech Group Co. Ltd. for the useful comments, sharing knowledge, giving advice, and insight which are invaluable to me.

I would also like to extend my special appreciation and thanks to Dr. Khine Moe Nwe , Professor and Course coordinator of Ph.D. 9^{th} Batch, University of Computer Studies, Yangon, for her useful comments, advices and insight which are invaluable to me.

I would like to express my respectful gratitude to Daw Aye Aye Khine, Associate Professor, Head of English Department, for her valuable supports from the language point of view and pointed out the correct usage in my dissertation.

My sincere thank also goes to all of my respectful Professors for giving us valuable lectures and knowledge during Ph.D. course work.

I also thank my friends from Ph.D. 9^{th} Batch for providing support and friendship that I needed.

I am very much indebted to my family for always believing in me, for their endless love and support**.** They are always supporting and encouraging me during the years of my Ph.D. study.

ABSTRACT

Emergency events never happen with prior alert and notification, and can damage people and cause permanent loss as consequences. For that reason, how to report the emergency service organizations, to distinguish the location of emergency case exactly and to drive that place in time become the key factors in developing countries. The performance of emergency services will be the best if they reach the incident site without delay where lives of damaged people and properties can be saved. In order to get the incident site finally, emergency vehicles use the road network. Therefore, the optimal route finding problem for emergency vehicle becomes the imperative component in evacuation processes. But, according to the lack of effective emergency route response system and good structure road network, there are many difficulties for drivers to go and give the rapid response and rescue actions immediately. Calculating the optimal route becomes one of the main tasks for emergency service transportation, which aims to provide the route from an emergency service location to the emergency event location on a road map. Choosing a suitable route finding methods amongst the existing ones that apply on actual road networks is also an important task for emergency service transportation. According to these inspirations, the optimal route finding system for emergency vehicles is developed on the principles of client-server architecture. This architecture includes reporting and identifying the exact location of emergency case, calculating the nearest emergency services and calculating the optimal route between emergency services and incident place. The proposed route finding system is mainly focused on Fire Emergency Event. In this research work, the importance of road network structure is presented, the problems faced by the drivers on the road network and the issues of route finding process are discussed and the solutions to these problems are described and explained in detail. To develop the proposed work, the database of Yangon Region Road Network is created with Open Street Map (OSM), and Quantum Geographic Information System (QGIS) tool is applied on it to create the usable data format in the system. The location of emergency services and streets condition data such as the streets that are not wide enough to pass in or one-ended streets which are supported by Myanmar Fire Service Department and some data are collected from satellite images and some are composed by crowd sourcing. The well-organized database of proposed system can be used for verifying the exact location emergency case, providing the nearest emergency services locations and the optimal route to go to the emergency location. The optimal route identification system for emergency vehicles is developed

based on Web Service by using well-organized database. In this research work, the distinction of the original route finding method and the proposed route finding method are presented, and these two methods are applied on real road network data in order to compare and analyze their operational performance. In this proposed work, distance and time are used as cost factors to calculate the optimal route. The proposed optimal route finding system is implemented as Web Based Application, and the results are displayed on Google map that must guide the drive way to reach the preferred place in time. The system can offer various kinds of optimal route finding system for other applications by altering and using appropriate geo-spatial databases.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF EQUATIONS

CHAPTER 1

INTRODUCTION

An emergency is a terrifying situation that risks to people and the environment, and it requires a quick response from emergency service organizations and rescues teams. The emergency situation and disaster case are different that both need quick actions but can prepare for an emergency and not for disaster [70]. Emergencies may affect an individual to an entire population in the area whereas disaster can affect large scale destruction of lives and properties. The proposed work is focused on emergency situations and the emergency cases might be carried out from accidents, fires, explosions and so on. Emergency events can take place anytime and everywhere lacking significant inform with alarm or message. Generally, urban fire is a violent problem for developing countries. Because of increasing the buildings, population and social development, the role of the emergency response services become more demanding. The most frequent emergency event in Myanmar is Fire case and almost 900 fire cases happened in every year. Therefore, the huge amount of possessions and survives are unfortunately vanished through fire each year.

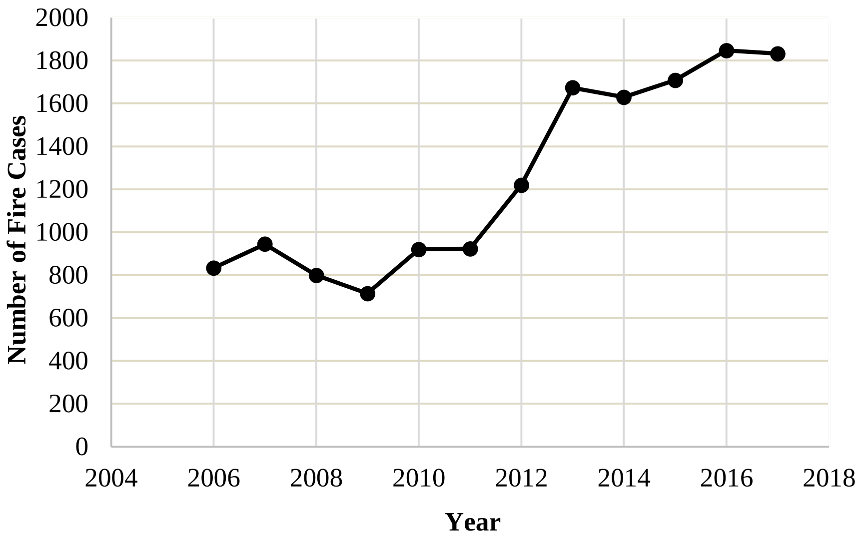


Figure 1.1 Number of Fire Incident Cases in Myanmar

Fig. 1.1 and Fig. 1.2 shows the number of fire incident cases and total damaged caused by fire in Myanmar between 2006 and 2017 [66]. When the fire emergency case happened, it is very important to have precise data and take rapid actions by rescues teams and emergency services organizations such as police, fire brigade and medical service as rapidly as possible.

To reply the rapid action for fire event, the proper road network infrastructure and the optimal path determination will be the main portions. If the optimal route and direction to go to the incident place is unidentified, it is relatively difficult to reach and give effective rescue services and can happen delays for evacuation processes.

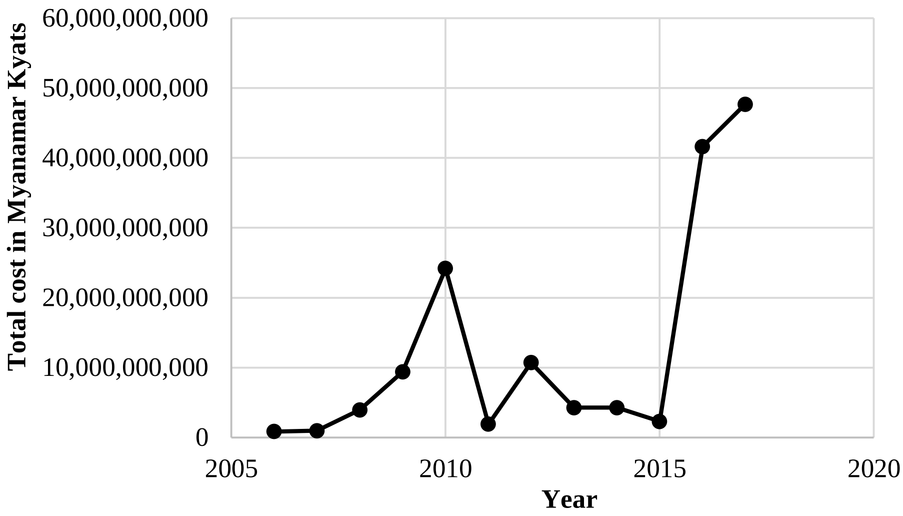


Figure 1.2 Cost Damaged by Fire

1.1 What is the Optimal Route?

The optimal route can well-define as a route between two locations on the road network such that the safe and convenient route to get from one location to another on the road network without delay along the way.

1.2 Optimal Route Finding

The optimal finding is intended to allow users and businesses to discover the safest and convenient way and direction between any two points (locations) on road networks (edges) of Google map. Most traditional optimal path finding methods are available to the user to compute the optimal route between the locations only as of the feature with minimum distance. The traditional route-finding approaches usually used the shortest route with travel distance time as selection criteria. To find the optimized route, a distinctive optimization algorithm was advanced and implemented to provide the safest and convenient routes.

1.3 Problem Statements

Due to the lack of an emergency response system, it is difficult to find the accident site as the received emergency call information and to give the rapid response in time. And also the metropolitan areas in Myanmar have the complex road network structure. Basically, the structure of the road network is affected by evacuation processes. Due to the structure of road network hours of vehicle delay increases to give effective emergency response activities. The delay is due to two major reasons:

1. One ended streets which are blocked or do not pass through the other places.
2. Narrow streets which do not have the sufficient wide to enter the vehicles.

If the vehicle travels from origin to destination using these ways it takes longer time. Consequently, the appropriate road network structure is also important for emergency transportation services. And, the issue of optimal route calculation on a complex road network becomes important for the evacuation process in our country. There are many approaches to solve optimal route finding problem but these methods are still challenging to apply on the large and complex road network. Note that, the road condition between two locations may be separated by the river or the wall, and also the roads cannot wide enough or the roads can block to enter the vehicles. How can we do to overcome these problems?

1.4 Motivations of Research

In today's world, emergency services play a major role when emergency cases occurred and the need arises to save valuable human life and properties. Because of these cases can cause serious damage to people and permanent loss. Evacuation of victims to the safe places and to provide the required facilities to incident place look like relatively simple but it is more difficult if the road network structure is complex and the optimal route is unknown. The complex road network can cause delay for drivers and can increase damage level. And also due to lack of optimal route identification system for an emergency case, it is difficult to identify the accurate location that take place emergency case when only received the crisis call and the close emergency services to provide the rapid response. There are many optimal route finding approaches but these methods still lead to some issues.

1.5 Objectives of Research

The main objective of the proposed work is to develop an optimal route finding system based on the web technology for emergency vehicles especially for fire truck when the fire accident occurs on unstructured road networks. Other objectives are described as follows:

i. To develop an optimal route identification system for emergency event based on GIS by combining with web services
ii. To discuss and explore the importance of optimal route for emergency service transportation

iii. To highlight the importance of road network structure
iv. To highlight and solve the problems of the emergency services providers traveling on the complex road network structure
v. To examine the close emergency service for the emergency event location
vi. To find and support the optimal route between the fire stations and the fire incident place by using geospatial data and display the route result with detailed direction
vii. To modify and improve the original Dijkstra's Algorithm in order to apply the complex and large road network by reducing memory and time consumption
viii. To optimize the workflow of emergency transportation services

1.6 Contributions of Research

This research work implements the optimal route identification system for emergency vehicles which always use the unstructured road network for travelling and transportation. In order to implement the proposed work, the road network database of Yangon Region is well organized and prepared. This database consisted of the location and addresses of emergency services, the whole road network of tested region and the detailed data of road conditions. The proposed system is intended to provide the nearest emergency services and the safest and most convenient route between incident place and emergency service location. Choosing the suitable route finding method and calculating the optimal route on the complex and large road network is major issue for transportation. Most of the original routes finding methods to calculate the evacuation route are mainly seek for the shortest distance and that technique does not guarantee the capacity of route result and will not satisfy the demand for emergency transportation. And also route result does not present the bottlenecks and delays that exist on the way to go to the potential incident place. In the research work, the improved version of Dijkstra's algorithm is proposed and used to calculate the optimal route on the Yangon Region road network. In this proposed method, the delay conditions of road network are considered and both travelling distance and time are added as parameters for optimal route analysis. The propose method can reduce the processing time than original method and give the accurate and effective optimal route result for emergency vehicles.

The main contributions of the proposed work are described as follows:

1. Create road network database of Yangon region
2. Develop the system that provide nearest emergency services and optimal route between incident location and emergency service location

3. Solve the issues of emergency vehicles that travelling on complex structure road network
4. Propose modified Dijkstra's Algorithm
5. Analyze the route result of original and proposed method
6. Compare the time complexity of Dijkstra's algorithm and modified Dijkstra's algorithm

1.7 Organization of the Research

This research is structured with seven chapters, including a general introduction including optimal route and optimal route finding. The related works of the proposed system are surveyed in Chapter 2 and background theory is described in Chapter 3. The nature between original route-finding method and the improved methods are completely described in Chapter 4. Chapter 5 presents the general architecture, the study area and data preparation of the proposed system. The development and investigational results of the proposed system, and the comparison of the proposed method and the original method are carefully discussed in Chapter 6. Finally, Chapter 7 concludes this research with suggestions to advance the proposed work in the future.

CHAPTER 2

LITERATUREREVIEW

In this chapter, the techniques and material used in the proposed work are discussed. The literatures related with the proposed work were also presented and summarized which were studied along subject matter of the research work.

2.1 Google Map

A web mapping service application technology that is provided by Google is Google Map [52]. It has the ability to provide comprehensive information about geographic areas and locations around the world. It can also support not only conventional road map but also aerial and satellite views of many places [44].

2.2 Nature of GIS and GIS Tools

GIS can perform a lot of major things within small amount of time and effort. It can also provide a platform with the ability to update geographical data and database manually without wasting time [7] .GIS is closely interrelated with the models building and representing the real world [11]. GIS offers the services that contain managing the data, entering the data, displaying the data, retrieving and analyzing the data and the presentation of results in graphical format and report format [15].

GIS is a dominant implement in the investigation and strategy of road network transportation. It has the ability to demonstrate not only visualizing of the various locations and routes but also to categorize in which they are positioned [36]. GIS technology is also suitable for a variety of conventions including resource management, land surveying, and traffic planning. Among the majority demand capabilities in GIS [42], some examples that related with the nature of route finding are described as follows:

i. Calculating the shortest path between the locations s and d
ii. Calculating the optimal path between the locations s and d
iii. Calculating the lowest cost path between the locations s and d

Open Street Maps (OSM) have the ability to produce the plenty of worldwide geographic information that include the transportation systems (e.g., road networks).In order to get the workable road network data format in the proposed system, one of the GIS protocol is needed to apply on OSM data [39]. ArcGIS and QGIS are two popular tools to create desired data format from road network files easily [26].

2.2.1 Role of GIS in Emergency Response

Geographic information system was developed to support geographical investigation and spatial decision making based on data related to geographic coordinates (longitude, latitude) [25]. It has the capability to provide the quick data access and to response the queries of geographic location and attribute data that based on the database information and also assist in complex analysis. In the emergency management, GIS becomes an effective platform with these abilities. The important role of GIS in emergency response directly rises with the combination of a technology that designed for spatial decision making to address numerous critical spatial assessments [16].

2.3 Transportation and Its Network

In our daily lives, transportation is one of the critical sectors. Transportation is a large part of our daily lives and it can provide as assistance in emergency situation. Generally, transportation networks are complex and not easy to analyze and describe. In order to develop web-based transportation systems, it is necessary to use graphs to study and represent their features and things on spatial realities [68]. Geography of transportation graph theory is related to develop the representations of transportation networks that consist of points (vertices) and lines (edges).The resulting graph is not a map, a precise model of the real world and it is an abstract representation of reality in order to develop an idealized structure of actual transportation systems [71].

2.3.1 Role of Transportation in Emergency Case

Transportation affects various part of civilization. Major portion includes the finding of the optimal route in transportation [2]. It is one of the approaches to satisfy the demands of emergency events. Transportation is a main part to provide the services of emergency response system that include the following.

i. To access the facilities for accommodation, medicinal and food
ii. To verify areas where the emergency case happened
iii. To evacuate from the regions before or immediately after happening an emergency event, if needed

If the route choice of emergency transportation is optimal, then the emergency service work flow becomes convenient and efficient [6].

2.3.2 Impacts of Road Network Structure

Road network structure has a massive impact on sustainable development because it connects with the public transportation and transportation efficiency of urban populations [58]. Nevertheless, the irrational road network infrastructure generates harmful effects, like that the emergency transportation, increased traffic misfortunes and so on. Therefore, it is necessary to have the well-structured road network construction to reduce the traffic conjunction, emergency delay and public transportation issues.

2.3.3 Role of Road Network in Transportation

Road network plays as vital role for improving local transportation which is directly connected to the financial growth of the city and people's livelihood [13]. Road networks are generally considered the most important ones in an emergency because it will support evacuation, emergency response, relief and recovery operations, access to the affected populations and reduced the terrible event. The ability to provide critical emergency services can threaten when the road networks are unstructured or disrupted by a hazardous event [22]. Nowadays, the delays are highly rising for emergency transportation services due to the weak structured road network in many developing countries [61].

2.3.4 Network Analysis in Transportation

Generally, the road segments and street intersections are represented as edges and nodes in the graph of transportation network analysis. Network analysis alters the complex transportation structure to its basic element structures through the use of basic mathematics from graph theory of travel claim models.

In graph theory, there are two types of network analysis:

i. Transport network analysis in convention
ii. Transport network analysis in syntax

2.3.4.1 Transport Network Analysis in Convention

The transportation network is conventionally represented as graph and the edges represent the links of the network and the intersections become vertices of the graph. It is possible to examine the structure of network and to capture the connectivity by using various graph theoretical indicators.

2.3.4.2 Transportation Network Analysis in Syntax

Space syntax is another method of transportation network analysis. It identifies the connection components in a blueprint which may have the substantial spatial occurrence. Generally, the streets are significant spatial entities in urban street network [24]. In this analysis, land used regions signify as node and streets represent as edges in urban structure. Therefore, the convex spaces which are the highest convex polygon and axial lines which are the longest line of sight are used as spatial elements in syntax analysis. These components are defined by geometry of the local borderline in diverse sections of the space maps. The space maps may translate into the comprehensive set of interconnecting axial lines or intersecting convex spaces produces the axial map or overlapping convex map individually. The resulted axial map permits a network amenable to graph arithmetic to be carried out in a well-defined mode to analyze the urban networks. The basic method of analysis summarized to identify the axial lines that have certain communication to lines of route movements and converting these lines into the nodes or vertices of a graph whereas the axial intersections become the edges. This renovation generates the principal graph structure of network. The resultant graph may be explored by using predictable graph-theoretic methods.

2.4 Categories of Networks

The three types of networks on distinctive interest to the mission areas are as follows: [57].

i. Sparse Networks
ii. Planar Networks
iii. Road Networks

2.4.1 Sparse Network

In this network, the number node is smaller than the number of links. The sparse network would be defined as the network with two hundred nodes and eight hundred links. Most of the public transportation networks are sparse network. There can be approximately four links leaves from one node. In this case, a sparse network can be presented with a matrix form and only four places would be used for each row of the matrix and the rest would be left idle. There are many efficient algorithms to handle for matrix network representation of sparse network. Although using the matrix based algorithms is appropriate for the sparse network, it is not highly inefficient with matrix representation. The spare network information can be stored by using tree building algorithms by using lists.

2.4.2 Planar Networks

The planar graphs are the graph that can draw on the two-dimensional plane and which has no crossed edges each other [34]. The computational bound of the shortest path algorithms are higher than the methods characteristic to planar graphs. Meanwhile the transport network and road network are usually defined as planar networks. More efficient solutions to these problems can be taken by using the application of the algorithms from this group. Conversely, the crossing and bridges can destroy the planarity and consequently incompetent, limit or complicate the application of these algorithms. For this reason, the methods for the planar graphs will not be considered. The appropriate algorithms for planar networks are still proposed by researchers.

2.4.3 Road Networks

In a road network representation, nodes represent the junctions or cities and the certain relations of which are linked by roads or edges [54]. In general, the amount of nodes is huge and the cost of links are always positive in the road network and the ration of the amount of nodes and number of links is almost 3 such that the total amount of nodes 3000 and the total number of links 9000. The road networks are usually defined as planar and sparse. The loops, only of non-negative cost, can contain in road networks. The distinctive feature of the road networks is non-negative links length property.

2.5 Categories of Routing Queries

The centralized GIS server can submit the various types of routing queries. Many approaches have been established to solve and satisfy the states and necessities of routing inquiries. There are two types of routing queries to find optimal route to calculate the optimal route to known-destination from the current location and to locate the nearest facility of the travel distance with time feature without knowing the destination implicitly [9].

2.5.1 Known Destination Routing Query

In this query, the users want to acquire the optimal route to a definite destination [59].The optimal route will change during emergency case when road conditions are provided because some road can be closed or blocked in some situations. For example, when the emergency car driver wants to drive from the rescue station to the incident place, the complete optimum route preceding to departure based on the existing situation of transport system. But it might not be absolute optimum route because of road conditions. Therefore, the route midway is needed to change and it is needed to plan the innovative route from the present location to the target based on the condition of road. This query is more complicated than the traditional perception because the drivers who are not familiar with the road network cannot know the condition of road in emergency area.

2.5.2 Unknown Destination Routing Query

This enquiry can be categorized as the nearest neighbor problem. In this query, the drivers might request the location of the nearest facility without knowing the destination, such as the nearest restaurant, cinema, shopping mall, hospital, gas station or hotel. The nearest facility is defined as the feature of time or distance that travel on the road network. Actually, both of nearest target and related optimum path need to be recognized on transportable time. If the road conditions provide, the best route will need to recalculate and the nearest destination can also change in risky condition. For example, the user wants to find the nearest restaurant after he/she has checked into a hotel within unknown resident area. According to this example, the user expects the navigation facility not only to offer the adaptive direction leading to it but also to approve the rationality of the nearest restaurant. The nearest neighbor and route searching methods are needed to solve this problem [57].

2.6 Geo Spatial Database

The collection of geo-spatial objects and graph of road network are consisted in the geo-spatial dataset. The real word geographic entities are represented by objects which might be spatial or non-spatial. The spatial attributes include width, height and shape and address and name are characterized as non-spatial attributes. The objects are characterized as points which are represented by a polygon or circle shape and these points are locations and they are unique. There are different locations for different objects. Each edge in the graph G represents a segment of a real world road and each of it has a length and the condition such as one way or blocked way. The distance of an edge is the measurement of the road segment it represents. A path of graph is a sequence of adjacent node and these are nodes are connected by edge of graph. The distance of path is the sum of the distances of edges. The distance between two objects can be calculated by using with efficient method over a road network.

2.7 Related Works with Current Research

Emergency events never come with formerly warnings in the daily life. According to the increase in accidents and emergency events, many lives and properties are lost because of delayed response. This delay is happening due to many reasons that the emergency location is not exactly identified based on the emergency call, and one of the main reasons is that the emergency vehicle driver does not know the optimal route to go to the emergency place. These cases are real challenges to get the effective emergency response actions. The problem of finding the optimal route is also a main issue for the emergency response system. Geographic information system (GIS) is used as the suitable technology in emergency management to determine the optimal route for emergency vehicle like fire, ambulance and other service vehicle, etc. This study was done for developing the optimal route finding for emergency vehicles utilized in case of any emergency situation that occurred on road network. The optimal route is calculated by using the proposed routing method and the results of this proposed system are displayed on Google Map.

Recently, the varieties of research works have been taken to address the problem of optimal route finding by using many approaches. Some of these works are described below. The paper "Implementation of Graph Theory in Computer Networking to Find an Efficient Routing Algorithm", by M.Nayeem Shah, analyzed the computer routing protocol that executes most efficiently because data transmission is grounded on the routing procedure [46]. The protocol chooses the optimum paths between any two nodes in the computer network and different kinds of routing protocols are applied in the precise network environments. One routing protocol is not efficient when the numbers of nodes are varied even though it seems to be efficient in one network. Open shortest path first, routing information protocol and enhance interior gateway routing protocol are assembled around the single source shortest path algorithm. The one eldest routing protocols still in service is RIP, OSPF is used as used interior gateway protocol in huge enterprise networks and EIGRP is used as Cisco's proprietary routing protocol. To create complex consequences and define which algorithms are finest for graphs of variable lengths MATLAB software is used. According to information of experimental data the brute-force algorithm is efficient for very small network and Dijkstra's Algorithm is more suitable for large networks. This paper showed that for medium and small sized networks enhanced interior gateway routing protocol and open shortest path first calculate the shortest path within the comparatively close to total amount of time. The very large networks need to be selected the routing protocol but the efficiencies of improved interior gateway protocol and open shortest path first are very similar in most case.

This paper [53] described the importance role of road for daily lives transportation and travelling and presents the key problem of shortest path computation on road network. And also discussed the various shortest path algorithms that are previously developed in recent years for wide range of applications [33]. S. Sivakumar and C.Chandrasekar proposed the new MSDP algorithm by using multiple parameters. This method is intended to stun some issues in the existing Dijkstra's algorithm and the authors compared the efficiency of their proposed method with the original algorithm. The proposed procedure is established by considering the several issues existing in the existing improved Dijkstra's algorithms. In proposed method, numerous factors are added to discover the effective shortest path for road networks instead of single factor in original one. The result came out from the MDSP algorithm proved that the proposed method competently discovered the result of shortest path within lowest time complication.

N.Kai, Z.Yao-ting and M.Yue-peng proposed the emergency response system based on Dijkstra's Algorithm in [25]. The authors discussed and highlighted the critical issue in emergency situation and the practical problem in networking, computer science and further application areas. The proposed work presented the summary of shortest route inquiry for an operative emergency response tool to reduce harmful activities. Theories of graph and network analysis in GIS were discussed to model and analyze the traffic networks. In this proposed work transportation network was regarded as weighted graph consisted of set of vertices and edges. The distance of shortest path from source to all other lasting vertices in the graph was calculated by using Dijkstra's algorithm. This proposed work provided optimal route finding based on the web services from the locations of specified response team locations to incidents place based on the combination of web services technology and GIS. But, road conditions and traffic congestion are not considering in the current application. The supplementary research is dedicated on incorporating this system with real time traffic flow to show more active, consistent and precise routes to emergency management centers.

N.Amalina, M.Sabri ,A.S.H.Basari , B. Husin and K. A.F. A. Samah proposed "Simulation Method of Shortest and Safest Path Algorithm for Evacuation in High Rise Building" [45]. They concentrated on high construction during dangerous occurrence and evacuation. The authors discussed problems faced by the evacuees and carried out to overcome the problem. The aim of this work was to recognize the appropriate shortest and safest path algorithm for evacuation process. The evacuation preparedness simulation model is designed and developed via shortest path algorithm to offer the appropriate leaving route to save the evacuees. To achieve the objective five steps of methods were carried out by implementing Dijkstra's shortest path algorithm by using MATLAB software. After importing the matrix data, the source and destination ID was followed. The shortest path was generated in third step and the edges are removed in fourth step. The safest path was generated in final step. The result of proposed work can offer the shortest and harmless path for the evacuees. As future work, the system was enhanced by using other approaches such as Ant Colony Optimization (ACO) and Particle Swarm Optimization (PSO).

M.Sharma, J.K.Gupta and A.Lala surveyed the route choice models in transpiration networks [48]. It was not an easy job to find the shortest and appropriate route by taking into account all the significant factors. All the integrated variables in the problem were directly affected the accurateness and effectiveness of the route. To develop the route guidance system, many research have been commenced and numerous techniques have been developed for this tenacity. In this paper, different route choice models were discussed and surveyed. This paper provided an ephemeral study of all the accessible clarification for the choice of the shortest route.

M.T.Zar and M.M.Sein proposed the public transportation system for Yangon downtown area by applying A Star method is appropriated when knowing both preliminary point and target point [62]. For urban areas, most of the residents use the public transportation so developing the public route transportation system is the key technology. The spatial database of this system was organized with bus stops, routes of bus, bus information and the related positions. OSM was used to make as base map for the proposed work. Users can get the bus route information founded on the present position and preferred destination. If the user uses an admissible heuristic function, A Star method is completed that it always discovers the minimum distance path. The proposed public bus transportation system provided the user that the bus information which is needed for them and also provide the planning system to plan the route and transportation mode with cost. For the future work, the proposed system will be extended and added with other important transportation factors such as waiting time at the bus stop, walking time to the bus stop.

T.Chondrogiannis,P.Bouros,J.Gamper and U.Leser proposed k-shortest path finding algorithm with limited overlap (k-SPwLO) [9]. In the various applications of researches and industries, shortest path finding on road networks is major problem. But, the result of providing individual shortest path is often not enough for some cases. The alternative paths which may be longer or shorter but have other interested things, have been interested by users. This paper proposed the k-shortest paths finding method by using limited overlap query. This is aimed to compute paths that are appropriately different to each other, and as small as possible. Firstly, the authors proposed an exact algorithm namely Multipass which traverses the network k-1 times and include two pruning criteria to decrease the number of paths that have to be examined. They also proposed two approximate algorithms, to achieve better performance and scalability, which trade accuracy for efficiency. OnePass+ traverses the network only once but works the same pruning criteria as Multi-Pass. The process of ESX is that it finds the alternative path by eliminating edges from the road network incrementally and then runs the shortest path inquiries on the restructured road network. An general investigational examination on real road networks shows that OnePass+ tracks pointedly faster than MultiPass and its result is near to the exact solution, MultiPass overtakes state-of-the-art exact algorithms for computing k-SPwLO queries and ESX is faster than OnePass+ but slightly fewer precise and scales for large values of k and large road networks. For the future work, alternative routing is extended by adding other constraints and plans to observe the calculation of various different paths inn dissimilar natures of systems such as social networks and web graphs.

This paper discussed the issue to determine the shortest path in traffic networks and proposed the dynamic routing system that based on the integration of real time traffic conditions and geographic information system technology [3]. This system was developed to find shortest for some locations of Kumasi Metropolis in the Ashanti Region of Ghana. The significant objective of this paper was to offer the optimal route for emergency services especially in Kumasi. The distance between 51 locations of the towns with the major roads was measured and a matrix was expressed. To determine the shortest distances from any location to any destination within the Kumasi Region Dijkstra‘s Algorithm was selected. The outcome shows significant reduction in the actual distance as compared with the normal routing and also indicates the importance of algorithms in the optimization of network flows. The shortest distance from any area in Kumasi metropolitan area to another can easily be calculated using this system to reduce the typical loss of lives in emergency case.

This paper described optimal path finding in real urban road network with criteria decision based approach [55]. Most of the methodology which conducted in several years only considered and implemented the system of cost function with single variable such as transportable distance or time. In this propose method, seven diverse attributes like distance, time, traffic volume, road width, no. of intersection, parking and encroachment on road are used as cost factors for decision making technique and AHP is demonstrated to obtain the optimal path by considering above mentioned traffic attributes over specified zones [28].

These seven variables are cooperated by using Multi-Dimensional Cost Model. The model is developed and implemented in Nagpur city of India. This model is deliberated to determine the optimum path among different Origins and Destinations in real urban traffic network. Bu using this model, the optimal path between various zones of Nagpur city is found out. From the tentative results, it is perceived that in most cases the optimal path is prominently obtained for travel distance or travel time as evaluation criteria. This methodology concretes the way for more intelligent traffic system. In this proposed work, the relative importance of quantifiable attributes such as the travel distance, travel time, traffic volume, road width, no. of intersections etc., were modeled using their relative level to standards and as future work the relative importance of unquantifiable attributes such as parking condition and encroachments on road are added to implement.

A.Jain1,U.Datta,Neelam and Joshi implemented the modified Dijkstra's Algorithm to calculate the shortest path for 'N' node with limitation [23].All over around the whole world used roads and it is one of the most useful means used most commonly for transporting and travelling. Road network also plays the key role in people daily lives activities such travel to work, go to schools and to transport and deliver goods and products. The manipulation of shortest paths for many application areas between various locations and places seems to be a main tricky on road networks. The massive kind of submissions and inventions were presented to solve the problems by developing numerous shortest path procedures. Although, shortest path finding problems still exist and need to solve. To overwhelm the shortest path problem, the authors proposed Modified Dijkstra's Shortest Path using Priority Queue with Linked List algorithm using multiple parameters by making some changes and generate. To prove the better efficiency of proposed algorithm, existing algorithm used to compare. The proposed algorithm is increase the performance to calculate the shortest path using minimum iterations to find the minimum cost.

N.Gupta, K.Mangla,A.K.Jha and M.Umar discussed the routing process by using Dijkstra's Algorihtm [19]. The main focus of this paper is to compute the shortest path between source node and desired target node. In this paper, the combination of two or more nodes were connected with each other is defined as network. In this network, nodes can exchange the data from each other within the data connections. The process of routing is that to calculate the path from source to destination via request of data transmission. There are different route finding methods that provide to find the path and its distance over the traffic network but Dijkstra's algorithm is the best shortest path finding algorithm [57]. This method is used the connection matrix and weight matrix to find the shortest path. In the matrix, the path from source node from other nodes is consisted and can choose the column that contains destination and finally can get the result of shortest path. This approach is applied to calculate the possible shortest path between one location and another location in Google Map, computer network and routing systems. In this proposed work, the shortest route and shortest distance were easily calculated with Dijkstra's Algorithm by giving source and destination from users. This method repeatedly selected the nodes with smallest distances and moves to the target node. After getting the target node, it will display the result of shortest path with smallest distance.

In this paper, the authors proposed and implemented by using PgRouting, provides the structure that calculate cost parameters dynamically, with Dijksta's algorithm in Varanasi road network [41]. According to the testing result of proposed work, they also verifies their proposed method is very effective and appropriated with dynamic updated cost. It is proper for road condition investigation, journey planning and road network traffic conditions and so on. By using PostgreSQL, they expand and discover the alternative route by avoiding the obstacles and normal route with other routing functions based on the basis factors such shortest, fastest and traffic free. The proposed application is applied on the high perception technologies of Geo-server and Geo-explorer web platforms and PyWPS server is developed based on the technology and implemented by using PgRouting to calculate shortest path. The proposed work is beneficial for Varanasi city when the traffic congestion is too high.As future work, the proposed web based system is advanced by using enhanced method to increased processing time and execution speed within low investment cost. The proposed method would be improved by adding additional features like date, email alerts, SMS, hospital, university and so on. In GIS and LBS application, finding the optimal path in transportation network is most important analysis. At the present time, metropolitan traffic overcrowding is a complex and ubiquitous problem.

Travelling cost is continuously changed based on the traffic variations so the optimum path identification in time dependent transport network become the inspiring issue [1]. Using the static methods to calculate the optimal path is not efficient for dynamic network. In this paper, the authors proposed the optimal route finding system of a road network within continuous traffic congestion and also evaluated the features and constraints. And also they analyzed the new proposed method that based on the space-time partitioning and the conventional route finding algorithms. Some heuristic utilities were taken out from the graph structures and used to guide the answer in every panel and other heuristic methods are used to optimize for this version. Finally, the business model to collect traffic data is introduced. The proposed system might support the user to get the greatest path in the metropolitan expeditions.

The experimental result of system testing proved that this approach can be used as the new version to determine the shortest path. Nevertheless, more research findings are required to mature and to get the exact model.

In recent years, the usage of Geographic Information Systems has enlarged considerably and one of most demanding application is searching shortest path [42]. There are many classic shortest path search algorithms and most of the research works use Dijkstra's algorithm but it is not well appropriate to find the shortest path exploration in very huge graphs. This is the main reasons why Dijkstra's algorithm has many modified versions by several authors. Therefore, the authors, R.R.Puente and M.S.Cortés, proposed the modified Dijkstra's Algorithm by using reduce graphs. The output result of shortest path with proposed method is compared with two methods, A* algorithm and Dijkstra's algorithm [7]. In this testing, the resulted cost of the shortest path used by proposed method is equivalent to the cost of path by using Dijkstra's Algorithm for the original path. The result of proposed method is obtained the resulted path in a comparable or even in fewer time than when using heuristic procedures. The tentative results prove that the proposed algorithm is more proficient than the Dijkstra's algorithm in the big graphs. The proposed method is mainly appropriate to GIS, due to the way in which users execute a shortest path search in this kind of systems and use of reduced graphs considerably decreases the response time in the shortest path searching process.

In this paper, C.Bagchi, K.Chopra and M. Yamuna discussed the ambulance facility is one of the most important service to be functioned in every time [4]. Therefore, the authors Yamuna proposed and developed the ambulance service system by using modified Dijkstra's algorithm. The time decides the life and survival of the patient so the ambulance services are needed to reach in time and to carry the patient to hospital within short time is most important. The authors proposed the new method namely modified Dijkstra's algorithm to implement ambulance service by using predefined data with distance feature to reach the nearest ambulance services. The proposed system decided the best ambulance service at the needed of the time based on emergency case. If the nearest ambulance service is not available at needed time, it will be able to choose the subsequent closest ambulance service and so on.

N.Kumar, M.Kumar, D.M.Denis, S.K.Srivastava and O.S.Srivastva pointed out one of the research questions that how to transport students safely and conveniently [29, 30]. To resolve this question they reported the school bus routing in Allahabad region. Since school bus transportation is a great challenge and it needs to be reliable, safe and competent way of journey. There are many parameters connected with transportation network such as travel time, travel distance, travel cost, driving speed, road resistance, turning movement and so on. The proposed system was developed based on geographic information system that can solve the large routing network problems with great precision and geocoding techniques that can considerably assist different types of route finding problems. The proposed work provided an efficient methodology to support in the optimal school bus routing system implementation by combining with GIS technology. The proposed system decided the best routes from one source to many destinations and it can contribute fastest, shortest and harmless route to reach schools with minimum distance and time. It also supported in school transport management to plan the shortest and fastest bus routes which can have the benefit in reducing the fuel consumption and saving travel time. This system was implemented and tested to calculate the optimal path for different schools in Allahabad region from SHIATS. According to the tested results, the proposed work proved that technologies may be very useful to solve shortest path problem in space time. The path result provided the fastest path coordinate schedule for constrained origin and destination with planned expected arrival or time departure time. Nevertheless, it was possible to decide the fastest and shortest way using GIS based network analysis however it may not work always as the connection on a real road network with different levels of congestion through varied time of a day.

The calculation of global optimal path is growing interest in transportation network when the network undergoes from unforeseen disturbance [49]. L. Shen, H.Shao, L.Zhang, and J.Zhao studied the problem of global optimal path finding with stochastic travel time in the network.Most of the paths finding algorithms are usually used the travel time as selected feature but many uncertain factors are accurately affected by traffic condition and the travel time is stochastic in reality. The travelers suffered from the risk of being late because of these circumstances. In path finding, the behavior of user's risk taking cannot capture by original path finding method based on smallest predictable travel time. In this paper, the authors proposed the new path finding algorithm under uncertainty by combining with K-shortest path algorithm grounded on the Backtracking method. The reliable path finding algorithms are proposed to overcome the above restriction. The proposed method can be guaranteed for the global optimum. To prove the correctness and proficiency of the suggested method some numerical examples are conducted and tested.

Transportation becomes the core part of our life and path finding is required in transportation, navigation, city planning and other real life application areas [8].There are many different algorithms to calculate shortest path problem such as Brute Force Algorithm, Bellman Ford Algorithm, Dijkstra algorithm, A-star algorithm, Hierarchical A-star algorithm and so on.The working processes and nature of each algorithm varies from each other. Dijkstra's algorithm starts from selected source vertex and then chooses from the unvisited vertex with minimum distance that through it to each unvisited neighbor vertex is calculated and updated with smaller distance. The heuristics value is used to find shortest path in A-star and Hierarchical A-star algorithm. In traditional Dijkstra's algorithm, memory usage is large enough and many redundant calculations are executed which may or may not be useful for the result of shortest path. The requirement of memory and computation can decrease by dividing into region of the map which added hierarchy. This can be used when region of map in attention is large adequate. Based on the application, it can also be used to reduce the load on the system memory. A.Chandak , R. Bodhale and R. Buradstudied all this three cases in terms of taken time to find shortest path. In this paper, efficiency and performance of different shortest path algorithms are also considered.

In many real world applications such as social network, traffic network and knowledge used graph to describe complex network model [62]. Solving the shortest path problem is an vital issue over the graph and it is needed to well study. In this paper, authors discussed shortest path finding within the vertices set identified by users. Most of the current route finding methods computes the variations of the given vertices and the shortest path calculation is one of these variations. When the graph size or numbers of vertices is large, the computation cost is extremely expensive. Therefore, Y.Yang , Z.Li , X.Wang and Qinghua Hu propose the novel heuristic algorithm by using best-first search method and give the two optimization techniques in order to improve the efficiency in route finding process. The authors also proposed an approximate heuristic algorithm to apply over large road networks in polynomial time and also show that the ratio bound value for the estimated algorithm is 3. The confirmed and experimented the efficiency of these methods by using real-life datasets. The experimental results of the proposed algorithms verify that it always overtake than the existing methods when the size of graph is large or the number of vertices is large.

In the previous few years, a number of heuristic algorithms have been implemented for faster path inquiries but the precision of these algorithms are always distant under sustaining [56]. Correct and fast path calculation is important for applications in onboard direction-finding systems and road traffic network routing. Q.Song, M.Li and X.Li developed an agglomerative graph separating technique for making high composed traverse distance partitions. In this proposed work, the authors discussed the problematic of capable path calculation on huge metropolitan street networks. They also constituted a three-level graph prototype established on the graph partition scheme for structuring the urban road network. The road network, modeled with three-level graph configuration decomposed the composite search problematic into numerous light searches on the graph hierarchies. After that, they also proposed an innovative classified route calculation algorithm, which benefits from the categorized graph prototype and develops a region cropping policy to considerably decrease the search space without compromising the accuracy. The proposed method is computationally very effective and is readily appropriate for path calculation problems for very big node-weighted graphs. The experimental evaluation was tested on the actual metropolitan road network of New York City. The effectiveness of the proposed graph partitioning and path computation algorithms is confirmed to produce optimal paths in real-time routing applications. This paper discussed a general method for exact optimum path calculation on large weighted graphs based on the hierarchical model framework and it can be simply stretched to edge-weighted graphs. As upcoming work, the proposed approach would be valuable for other kinds of networks such as social and communication networks.

There is the shortest path finding problem in many application areas [60]. Among them, Dijkstra's Algorithm is the well-known problem and also called label algorithm. By addressing the issues of this algorithm and the algorithm can be significantly improved. In this paper, W.S.Xi , investigated Dijkstra's algorithm , pointed out the insufficiencies of the procedure and proposed the improved method and conducted a series of targeted experiments. The experiments of the proposed method indicated the issues of original method have been effectively resolved. The improved algorithm was better than the original algorithm according to these reasons, the improved algorithm can escape dropping into an endless loop, it can acquire contiguous vertices in the shortest path and it can also solve the problem of more than one vertices obtain "p-label" at the same time. The research application will develop by using proposed algorithm as future work.

Computing the optimal route between the given source and destination with fast and efficient way is needed in dynamic route guidance system [9]. The original route finding algorithms are not sufficient to apply in large road networks because there are many factors that are not suitable for route computation in the large number of vertices. In this paper, G.R. Jagadeesh and T.Srikanthan proposed the generic method to establish the specified road network with various layer hierarchy that an competent hierarchical routing method. The nature of the proposed method is working that it breakdowns the route searching into a number of distinct explorations in small sub-networks. To advance the performance of the proposed method, heuristic layer-switching technique is combined with compromising the accuracy. The efficiency of proposed method is tested and verified on the Singapore road network and the results to be similar to the optimum smallest cost paths and established on the three layer hierarchy. The performance of the proposed method is favorably compared with the conventional route finding algorithm. The optimal path of the hierarchical routing algorithm is only 3.5% longer than the result route by using non-hierarchical algorithm. The hierarchical routing algorithm finds the near-optimal route within logarithmic time but the original route finding algorithms find in quadratic time [25].

Web mapping websites such as Google Maps and MapQuest use shortest path algorithms to find the driving directions [37]. The problem of shortest path finding is sometimes entitled the minimum interval path problem and usually tied with an extensive path problem in the networking or telecommunication. In order to consider these two operations, the general approach to be those of semiring. The exponentiation of semiring is done laterally the path and between the paths. The common work is called as the numerical path issue. The classic shortest path algorithm and the new ones are formulated as resolving the linear system based on the algebraic structures. The column generation technique can be solved the shortest path problems form the foundation of an entire class of optimization problems. There are master problems in survival network design problem and vehicle routing problem in which each column represents a path and the master problem is repeatedly resolved. This tricky turns out to be a shortest path problem usually with side limitations or negative arc lengths rendering the problem NP-Hard. When the different convenient path is found, it is added to the unique major problem that is now re-solved over a larger subclass of paths leading to gradually enhanced results. The computation time complexity of the Dijkstra's, Floyd-War shall and Bellman-Ford algorithms prove that these methods are acceptable in terms of their general routine in solving the shortest path problem however all of these algorithms produce only one solution. Producing the number of different optimal solution is the significant benefit of GA over these algorithms. Fuzzy logic and neural networks can also be executed in order to make them more intelligent and more proficient to improve the existing shortest path algorithms. For the future work, the proposed framework is needed to extend, to enhance and to determine shortest path between two locations in maps that represents the several types of networks.

2.8 Chapter Summary

This chapter mainly described Google Map, nature of geographic information system, types of networks, types of routing queries and geo spatial database which are the techniques and material used in the proposed work. It also highlights the importance role of GIS in emergency response, the importance role of road network and transportation during emergency case. The categories of network, routing queries and network analysis for transportation are also discussed. The proposed work is related with GIS and Google Map, road network, and geo spatial database are mainly used to implement the optimal route analysis for emergency vehicles. Finally, the literature reviews of the proposed work are described with the summarized form.

CHAPTER 3

BACKGROUND THEORY

This chapter discusses the concepts and nature of optimal route finding system related with GIS, and also describes the relation of graph theory and optimal route finding problems.

3.1 Graph Theory

Graph theory is the basis with an issue of Koinsber Bridge by L Euler Swiss mathematician in 1735 and it is widely used in network analysis, discrete mathematics, operations research, and combinatorial optimization. Graph theory is related to the graph properties in computer science and mathematics. Most of the definitions and concepts in graph theory are recommended by the graphical representation which helps to easily understand. Graphs provide an outstanding modeling tool which is used to model several classes of relations among several physical situations and several issues of the real world can be represented by using graphs [40]. Graphs provide a dominant tool to model objects and relationships among them. Generally, the graph is defined as a set of entities that symbolized nodes or vertices and connected by links which are defined as edges. In particular, a graph is a pair of G = (V, E) where the symbols V is the finite set of vertices and E denotes as the finite set of edges in graph G [65]. For example: ({A, B, C, D}, {(A, B), (B, D), (B, C), (B, D), (C, D)}). Some types of graph are described in Fig. 3.1.

Vertex - In the graph, vertex represents the intersection point where the street or edges meet together. The vertices simply happened when the intersect point is explicitly situated not when the two edges cross each other [31].

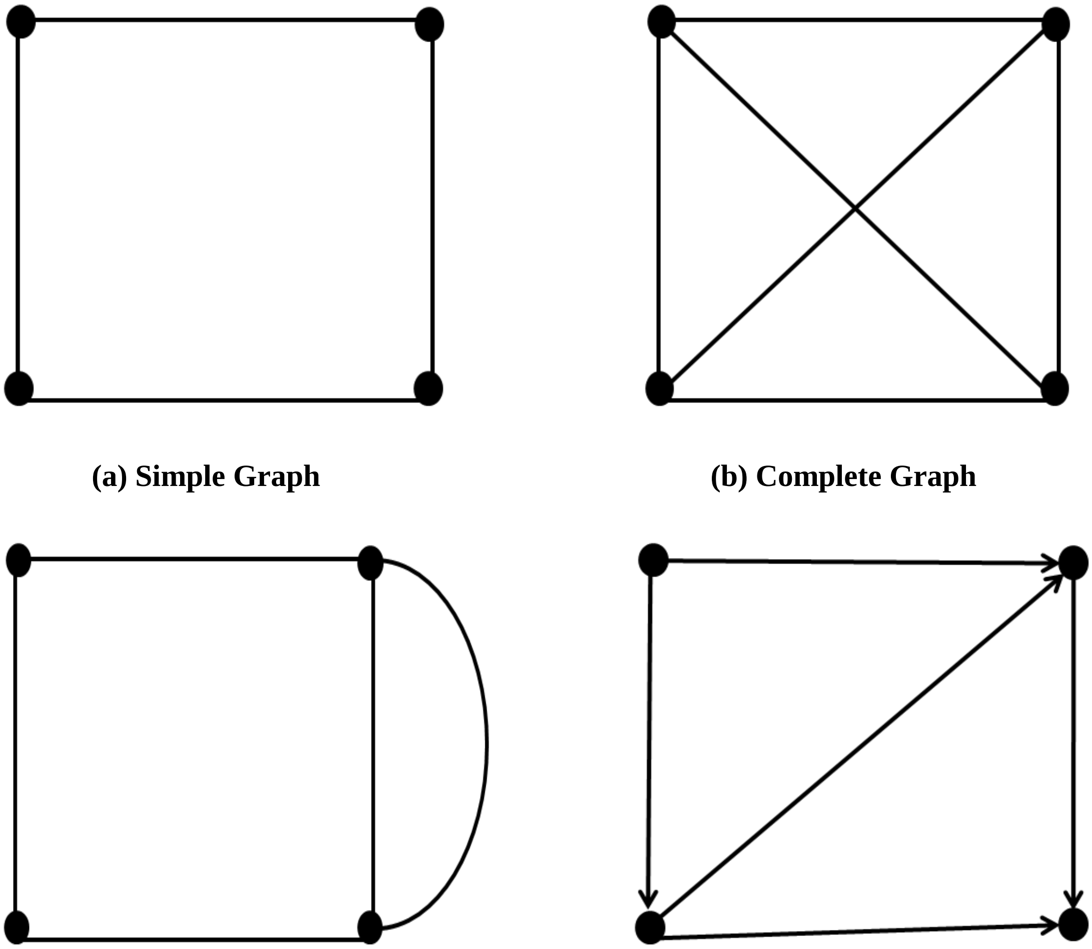
Edges – The pairs of vertices are joined by the lines or arcs are called edges. The connection between two locations such as streets or routes or two places is represented by edges.

Weights - The distance between two locations are represented as the weights with by using these attributes such as the travel time, distance and cost and the weights of the edges are assigned based on the problem. The weight is denoted by G= (V, e, w) in the graph.

Path - The path is a series of vertices using the edges [66]. In a graph, a path, from a source node S to a target node T, is defined as a sequence of nodes (v_0,v_1,v_2, ..., v_n) where S = v_0, T = v_n), and the links (v_0, v_1), (v_1,v_2), ..., (v_{n-1}, v_n) are present in E.

Weight of Path - The sum of the weight of the constituent edges on the path is called the weight of the path.

$$w\ (p) = \sum_{i=1}^{n} w(\ v_{i-1}, v_i) \quad (3.\ 1)$$



(a) Simple Graph (b) Complete Graph (c) Multi Graph (d) Directed Graph

Figure 3.1 Types of Graph

Simple graph: A simple graph is the graph which has maximum one edge between any two vertices.

Complete graph: In the completed graph, each vertex is linked by an edge to every other vertex in the graph.

Multigraph: There are any two vertices which are joined by more than one edges is called multigraph [65].

Directed graph: In the graph, the direction of each edge is assigned [14].

In the proposed work, the optimal route finding system for emergency vehicles is developed for Yangon region and Fig. 3.2 shows the sample part of tested region [73]. In order to determine the optimal route, firstly it needs to make the picture into a simple form which is easier to work that draw the lines for each street in the tested region as shown in Fig. 3.3, and place the dots where streets are intersected each other are displayed in Fig. 3.4. The simplified form is called a graph. The set of points or dots called vertices and the set of edges which connect the couples of vertices consist in the graph. Fig. 3.5 is equivalent to the figure illustrated above.

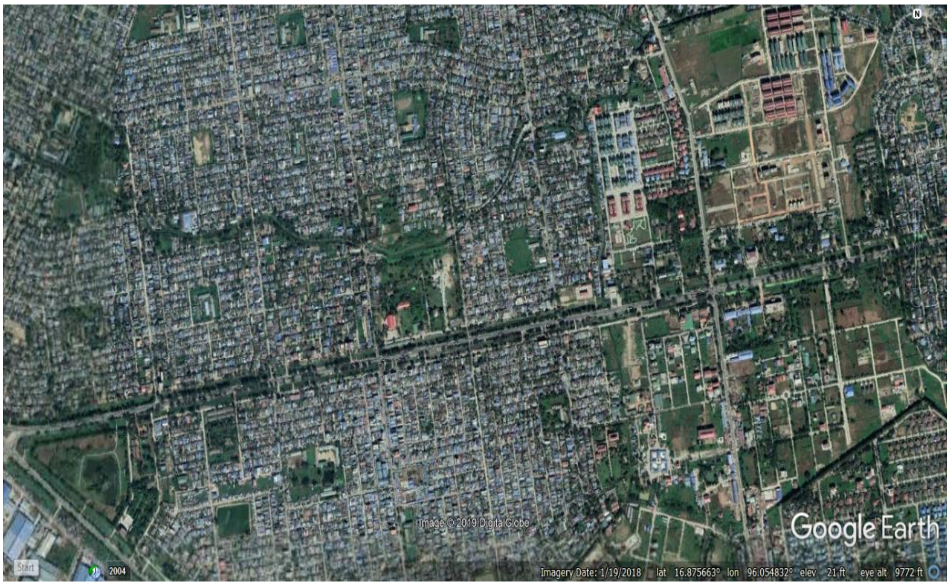


Figure 3.2 Part of Yangon Region

Figure 3.3 Line Shaping in Part of Yangon Region

Figure 3.4 Intersecting in Part of Yangon Region

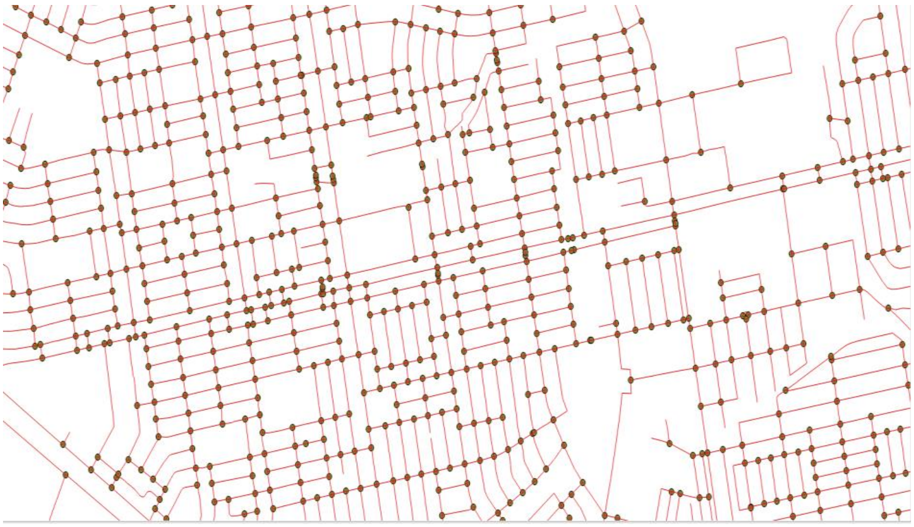


Figure 3.5 Graph Shaping in Part of Yangon Region

3.1.1 Application Areas Applied with Graph Theory

In various applications of different areas, graph theory models are extensively used to study and develop. In chemistry, graph theory is used to study the molecules and atoms, the constructions of bonds. In biology, graph theory is used for management efforts where a vertex represents regions where assured species exist and the edges denote migration track or development among the regions. This is essential when observing at breeding patterns or tracking the spread of disease, bedbugs and to study the influence of migration that distress other species. Likewise, graph theory is used in sociology for example to measure actors' reputation or to explore dissemination mechanisms. In computer science, graph theory plays an important role in automatic graph generation applications such as database design, software engineering, circuit designing, networks and data mining [50]. In engineering, economics and war science, the game theory is useful to the problems to find an optimal approach to achieve certain tasks in economical environments.

A digraph is used to represent the method of a finite game. In this situation, the vertices represent the positions and the edges signify the changes. Mostly, graph theory concepts are broadly used in operations research such as to model the transport networks, activity networks, game theory, and traveling salesman problems. It is also used in the shortest spanning tree in a weighted graph, obtaining an optimal match of jobs and locating the shortest path between two vertices in a graph [49]. There are many conventional problems that can be solved with graph theory, for instance, optimal path, shortest path, and longest path and traveling salesman problem. There is an important issue to provide the important information within a short time through the network by analyzing the geospatial data with search procedures and calculation processes. To provide the optimal path from rescue sites to the accident site throughout the road network is essential for emergency services. It is also important to construct a suitable transportation network to take prompt actions on a severe accident.

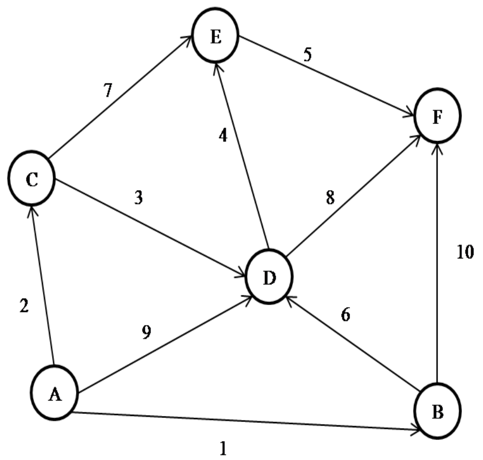
A transportation network is the movement of network demonstrating a movement of people, vehicles or goods. A transportation network can be defined as a weighted graph, or otherwise a road network. The weights could characterize the length of each road in the transportation network. If the road network consists of a finite set of vertices and paths is called a directed graph. In this road network, every path has the related cost that might be distance, cost or time for travel. The edge has start and end vertices and the sequence of vertices is the path between two nodes and the total weights of the edges of the path is called the length of the path. The problem of optimal path-finding is the conventional problem in road network analysis for the geographic information system. It highly depends on the nature of the transportation network and the distance between the target node and source node and it has the diversity of comprehensions. The graphs are geologically referenced and each vertex has well defined entire coordinate related to the earth [3]. Based on the primary model of networks, optimal route finding problems are regarded as graph problems. Optimal path finding problem calculates the path among many alternatives paths which is optimal on the road network.

3.2 Road Network Data Models

The road network system is symbolized by the sequence of nodes and connected edges with related weight to develop the road network model. In this model, the road network may represent two types of graphs as directed and undirected. This representation is used in the mathematical model to compute the optimal route in the road network system. In this model, the intersections are represented as nodes and the street and the links are represented as edges [67/68]. As a specialized type of graph, a suitable data structure plays an important role to represent the road network. The graph algorithms need to access nodes and links of the graph these are stored in the computer's memory. There are many data structures to represent the road network based on the structure and the algorithm used to manipulate [56]. Generally, graphs are generally represent by using incidence matrix, adjacency matrix and adjacency list [75].

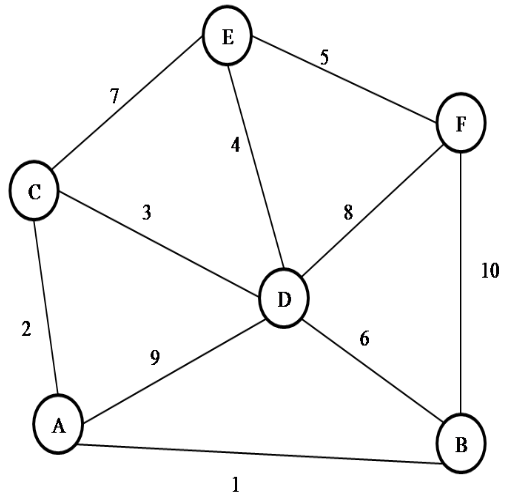
3.2.1 Incidence Matrix

The incidence matrix is represented with (*0*, *1*) for an undirected graph and this matrix has a row for each link and a column for each node. In this case, (u, v) = *1* if and only if, node v is occurrence upon link e and otherwise (u, v) = *0*. In this matrix representation, the summation of the degrees of all the vertices in a graph is equal to the number of ones of the undirected graph (no loops) is equal to [24].



	1	2	3	4	5	6	7	8	9	10
A	1	1	0	0	0	0	0	0	1	0
B	-1	0	0	0	0	1	0	0	0	1
C	-1	0	1	0	0	0	1	0	0	0
D	0	0	-1	1	0	-1	0	1	-1	0
E	0	0	0	-1	1	0	-1	0	0	0
F	0	0	0	0	-1	0	0	-1	0	-1

Figure 3.6 Incidence Matrix Representation of the Directed Graph



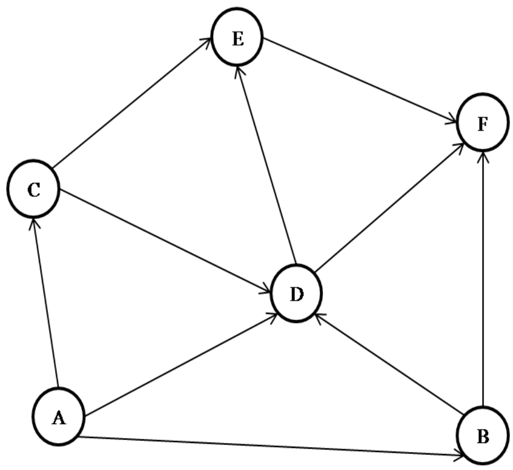
	1	2	3	4	5	6	7	8	9	10
A	1	1	0	0	0	0	0	0	1	0
B	1	0	0	0	0	1	0	0	0	1
C	1	0	1	0	0	0	1	0	0	0
D	0	0	1	1	0	1	0	1	1	0
E	0	0	0	1	1	0	1	0	0	0
F	0	0	0	0	1	0	0	1	0	1

Figure 3.7 Incidence Matrix Representation of Undirected Graph

The incidence matrix is represented as (u, v) = *1* or *-1* for directed graphs based on whether the link leaves node v or it enters node v [14]. Fig.3.6 and Fig.3.7 show the incidence of the matrix of the directed graph and undirected graph respectively.

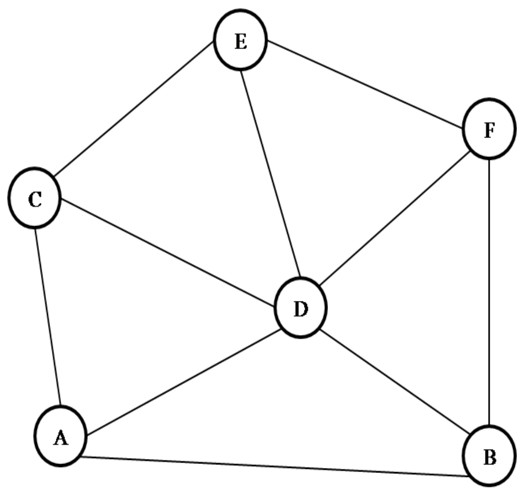
3.2.2 Adjacency Matrix

In this matrix, the graph is stored as m by m matrix with two-dimensional array and the rows and columns are labeled by the nodes of the graph. The nodes u and v are defined as adjacent if there is a link between them. Based on this condition, *0* or *1* is placed in the position u, v. The diagonal of this matrix must have *0* s for the graph with no self-loops.



	A	B	C	D	E	F
A	0	1	1	1	0	0
B	0	0	0	1	0	1
C	0	0	0	1	0	0
D	0	0	0	0	1	1
E	0	0	0	0	0	1
F	0	0	0	0	0	0

Figure 3.8 Adjacency Matrix Representation of Directed Graph



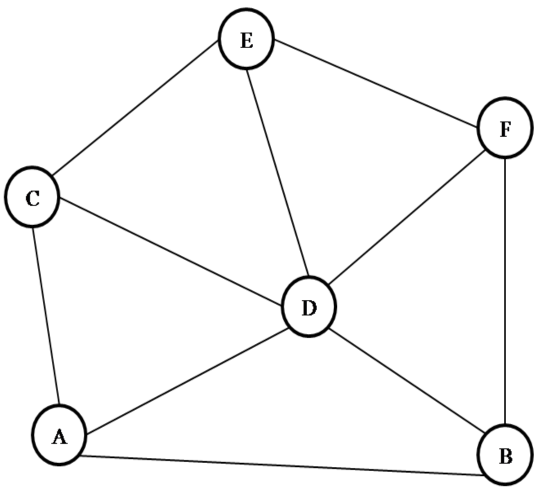
	A	B	C	D	E	F
A	0	1	1	1	0	0
B	1	0	0	1	0	1
C	1	0	0	1	1	0
D	1	1	1	0	1	1
E	0	0	1	1	0	1
F	0	1	0	1	1	0

Figure 3.9 Adjacency Matrix Representation of Undirected Graph

The adjacency matrix of the undirected graph is symmetric. In the matrix demonstration of the directed graph, the number of ones is equal to the number of edges [53]. The example presentation of the directed graph and undirected graph representation of the adjacency matrix is described in Fig. 3.8 and Fig. 3.9.

3.2.3 Adjacency List

In computer science, another form of graph representation is the adjacency list. This structure contains a list of all nodes in a graph. In addition, each node in the list is connected to its own list containing the names of all nodes that are adjacent to it and the distances to those nodes are also stored. In this representation, the size of the array is equal to the number of vertices in the graph [15]. Fig.3.10 and Fig.3.11 show the structure of the adjacency list for the undirected and directed graph.



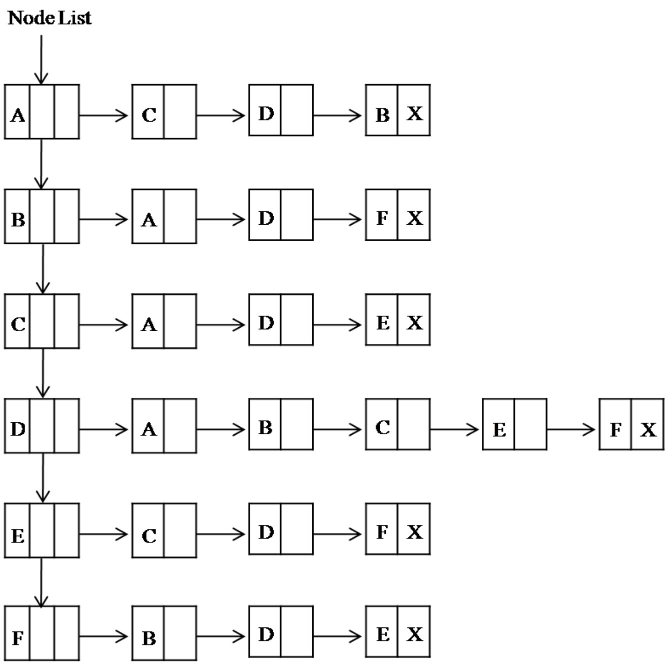
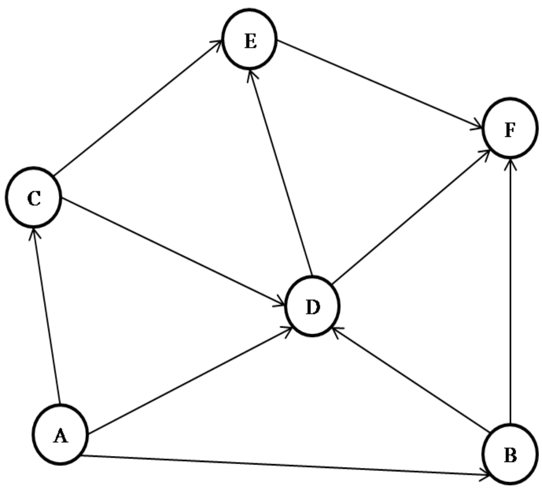


Figure 3.10 An Adjacency List for Undirected graph

According to studying the nature of the above three data structures, the representation of the adjacency list requires a small amount of space because the edges that do not exist is not required to represent. The space complexity of adjacency list is $O(|E| + |V|)$ and $O(|E| \times |V|)$ for incidence matrices and $O(V^2)$ for adjacency matrices. There are many 0s which are not used and empty storage in incidence matrix and adjacency matrix representations. In the route searching, finding the successor of the given node is the major process and determining all of its adjacent nodes is a major concern. Consequently, using the adjacency list is more appropriate to represent the road network. It is not only decreased the storage space of memory but also simplifies the routing computation [12].



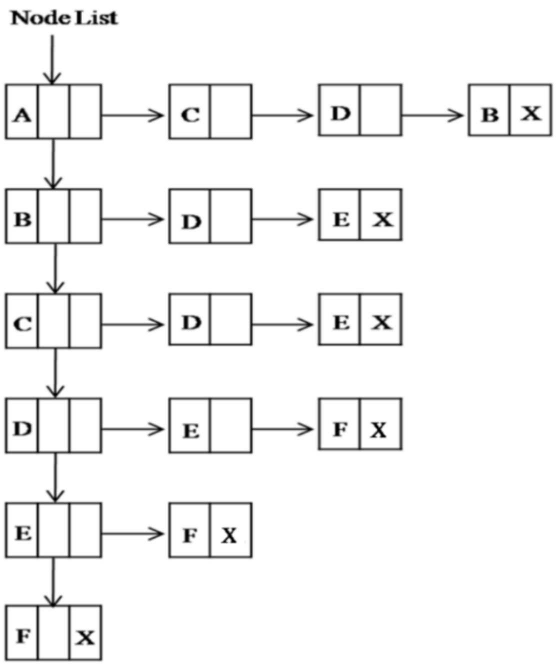


Figure 3.11 An Adjacency List for Directed graph

3.3 Route Finding Problems

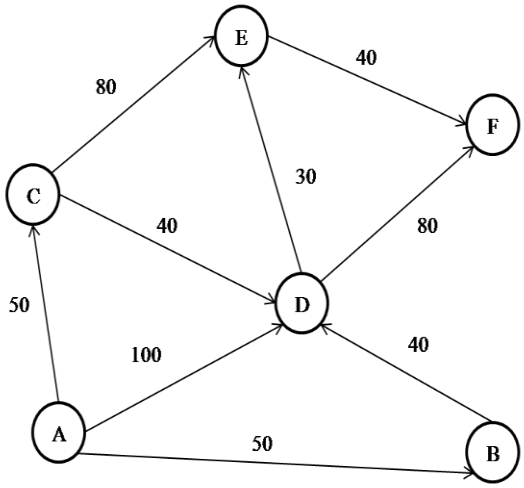
The problem of computing the optimal route is undoubtedly well-studied problems in computer science. In the transportation network model, route finding problems become a great deal of interest with both practical and theoretical reasons. The route finding-problem is important for a variety of works such as logistics, operation management, transportation, system analysis, and design, project management, game programming, network, and production line. In general, the problems related to route finding has two variants:

i. Single-Source Approach
ii. All-Pairs Approach

The choice of which approach to use is depended on the features of the graph and the mandatory application.

3.3.1 Single-Source Approach

The nature of Single Source Approach is that it calculates the shortest path from a node to all other nodes in the weighted graph. The first category is single-source shortest-path (SSSP), where the objective is to discover the shortest-paths from a single-source vertex to all other vertices.



Source	Total Distance
A	0
B	50
C	50
D	90
E	120
F	160

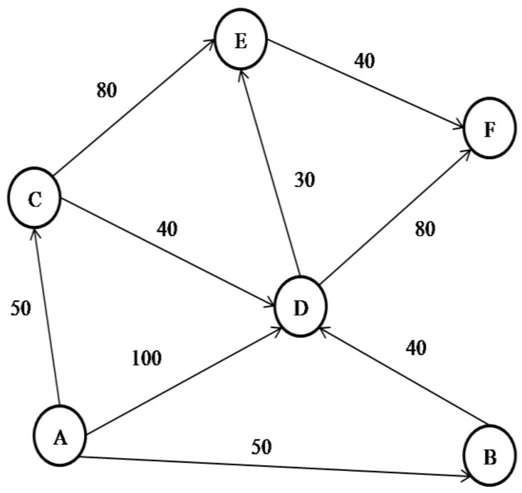
Figure 3.12 Test Graph and Result for Single Source Shortest Path

It is used to detect the link failures of protocol routing for IP network [47]. It does not support the graph with negative weight. This approach reports the distance between source vertex to all other vertices as shown in Fig. 3.12.

Definition: In a given graph G = (V, E), calculate all of the distances between source vertex s and destination vertex'v', where 's' and 'v' are elements of the set of 'V'.

3.3.2 All-Pairs Approach

The category of all-pairs shortest-path is to discover the shortest paths between all pairs of vertices in the given graph. The nature of All Pairs Approach is that it calculates the shortest path from a node to all other nodes of the weighted graph as described in Fig.3.13.



Source	Destination	Total Distance
A	F	100
C	F	90
B	F	90
A	E	80
C	E	70
B	E	80
A	B	50
D	F	50
A	C	50
A	D	50

Figure 3.13 Test Graph and Result for All Pairs Shortest Path

It is used to distribute the delivery of goods and the location of urban facilities [47]. And also use to determine the expected traffic load on different segments of a transportation grid.

Definition: In the given graph, $G = (V, E)$ and source $s \in V$, compute all distances $\delta\ (s, v)$, where $v \in V$. This method reports the distances between any two vertices in the graph.

3.4 Searching Strategies for Graph

Graph processing requires the ability to traverse graph also known as searching graph [64]. Traversal of the graph refers to the process of traversing every node exactly once in a systematic fashion**.** The key inspiration behind graph traversal is to mark each vertex when first visit and keep track of what other vertices have not yet entirely explored. Graph traversals are classified by the order in which the vertices are visited [35]. In general, there are two approaches to traverse a graph:

i. Depth First Search
ii. Breadth First Search

These two strategies are basic for graph searching and many other approaches are built based on them.

3.4.1 Depth-First Search (DFS)

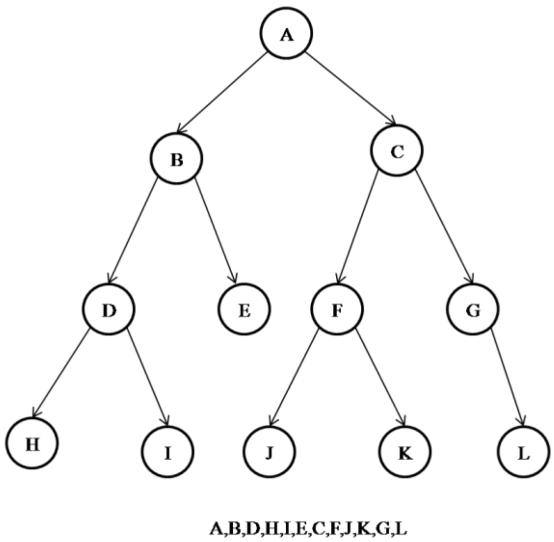


Figure 3.14 Graph Traversal of Depth First Search

This approach is the edge-based scheme and traverses from the root node and explores the search as far as possible from the root node. The exploration of a node is suspended when the new node is found. This traversing method uses the stack for storing the visited vertices. In this traversing, every vertex visit exactly once and each edge is inspected exactly twice. In the processing steps of DFS, firstly it selects the unvisited start node and regards as the current node. Then find the unvisited neighbor node from the current node and visit it and makes this node as the new current node. If the current node has no unvisited neighbor node, back off to the parent node and make it the new current node and the above two steps will repeat until there is no more to visit. If there are still unvisited nodes, repeat from the first stage [12]. The DFS traversal of Fig 3.14: is A,B,D,H,I,E,C,F,J,K,G,L .

3.4.2 Breadth-First Search (BFS)

This approach is a vertex-based scheme and traverses from the root node and inspects all the neighboring nodes in the graph with level by level process [69]. This approach uses a queue to store the visited vertices. In this method one vertex is selected at first then it is as mark as visited. The neighboring nodes with the visited nodes are then visited and store in the queue consecutively. The node exploration is completed before visiting any other node in the graph.

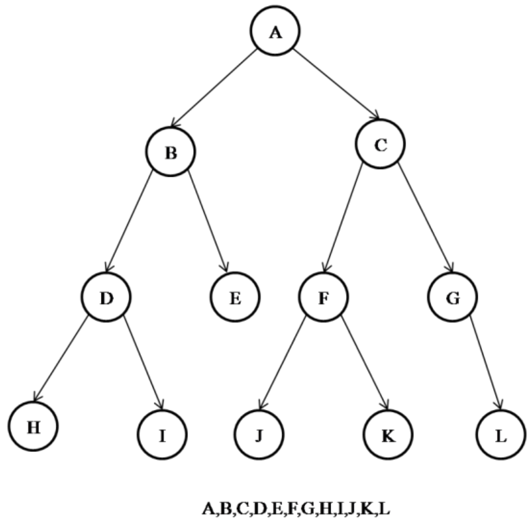


Figure 3.15 Graph Traversal of Breadth-First Search

In the processing steps of BFS, firstly, firstly it selects the unvisited start node that it is the root node and the level of start node is called as the current level. Then the start node visits all the unvisited neighbors' node in the current level. The newly visited node from this level becomes the next current level and will visits until there is no more to visit. If there are still unvisited nodes, repeat from the first stage [12]. Fig.3.15 shows the example of graph traversal of BFS with A,B,C,D,E,F,G,H,I,J,K,L.

3.5 Strategies of Distances Calculation

The calculation of the distance between two locations becomes a significant component of spatial analysis research areas and other sectors and the determination of the nearest distance is often a problem [20]. The suitable formulation is necessary to calculate distances between two locations or points. The nature of the data, application area, coordinates and the goals of analysis is associated with the proper formula. In this section, distance calculation in two dimensions and spherical surface are discussed.

3.5.1 Straight Line Distance Calculation

The simplest way to measure the distance between two points is the straight line distance. Nevertheless, straight-line distance can be calculated by using diverse formulas founded on the potential variation. Among them, Manhattan distance, Minkowski distance, and Euclidean Distance are often used [74].

3.5.1.1 Euclidean Distance

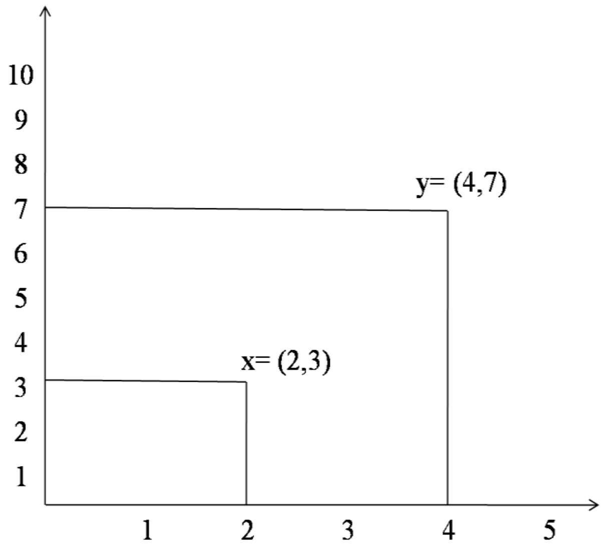


Figure 3.16 Euclidian distance

The normal distance that measures with a ruler between two points is defined as Euclidean distance and it is used to compute straight line distance between two points [72]. Let x and y be the two points that describe as $x_{1,}x_{2,}x_{3,} \dots, x_n$ and $y_{1,}y_{2,}y_{3,} \dots, y_n$ with n numeric attributes. Then, the Euclidean distance between two points x and y is defined as

$$d(x, y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2 + \cdots + (x_n - y_n)^2} \qquad (3.2)$$

Let $x = (2,3)$ and $y = (4,7)$ represent two points as shown in Fig 3.16. Then the Euclidian distance between two points is $\sqrt{(2)^2 + (4)^2} = 4.4721.$

3.5.1.2 Manhattan Distance

In the Manhattan distance is the sum of the absolute differences of two points of their cartesian coordinates. The distance between two points measured along axes at right angles.

Let x and y be the two points that describes as $x_1, x_2, x_3, \dots, x_n$ and $y_1, y_2, y_3, \dots, y_n$ with n numeric attributes. Then, the Manhattan distance between two points x and y is defined as

$$d(x, y) = |x_1 - y_1| + |x_2 - y_2| + |x_3 - y_3| + \dots + |x_n - y_n| \quad (3.3)$$

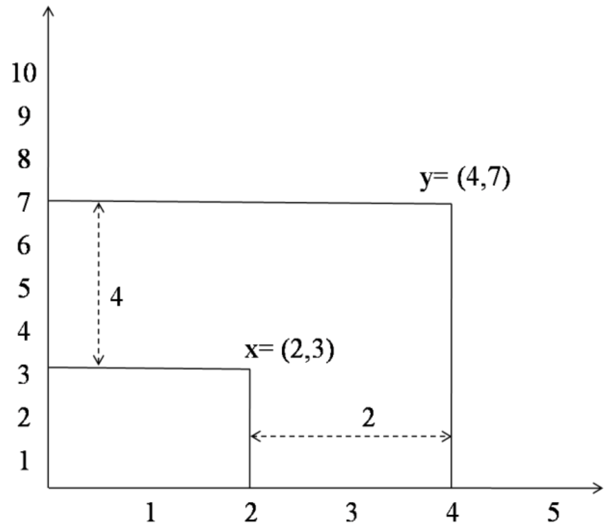


Figure 3.17 Manhattan distance

Let $x = (2,3)$ and $y = (4,7)$ represent two points as shown in Fig 3.17. Then the Manhattan distance between two points is $2 + 4 = 6$ [18].

3.5.1.3 Minkowski distance

The distance that recognized as the p-norm distance [10], the generalization of Euclidean distance and Manhattan distance is called Minkowski distance and it is defined as the following equation where m is the real number such that $m \geq 1$ and it represents the Euclidean distance when $m = 2$ and Manhattan distance $m = 1$.

$$d(x, y) = \sqrt{|x_1 - y_1|^m + |x_2 - y_2|^m + |x_3 - y_3|^m + \dots + |x_n - y_n|^m} \quad (3.4)$$

Euclidean and Manhattan distance is used because their implementation is relatively easy. In the spatial analytical models, it is more challenging to design algorithms implementation for distance of real road network [32].

3.5.2 Spherical Distance Calculation

The previous section discussed the distance measures based on two dimensions and it is a reasonable generalization for small areas. It is more challenging to implement the actual road network distance calculation in spatial systematic models [73]. To determine the distance between two points on the earth surface is affected by a certain degree of curvature. Therefore, the coordinate system comprised of latitude and longitude along with a particular formula is needed to calculate the geographical distance and also the radius of the earth is also need to use in the equation [63].

Figure 3.18 Latitude and Longitude Lines on Earth

The longest radius of the earth is 6378 km at the equator and the shortest is 6357km at the north and south poles. In general, the radius of the earth 6371 is used as the average value. The distinction between the latitude and longitude lines are described in Fig. 3.18. The line distance that alongside the equator line is called latitude line and the line distance among the west and east of the earth from the main meridian line is called longitude line. These coordinates are usually articulated in degrees, minutes, and seconds. To use in distance calculation, these coordinates needed to transform into the decimal degree. Equation 3.5 is used to convert into decimal coordinates.

$$\text{Decimal degrees} = \text{degrees} + \text{minutes}/60 + \text{seconds}/3600 \qquad (3.5)$$

3.5.2.1 Spherical law of cosines

The law of cosine is depended on the sides and angles of spherical triangles in spherical trigonometry [72]. The formula for spherical law of cosines is described as following and

$$a = \sin(\mathrm{lt1}) * \sin(\mathrm{lt2}) \quad (3.6)$$

$$\mathrm{b} = \cos(\mathrm{lt1}) * \cos(\mathrm{lt2}) * \cos(\mathrm{lg2} - \mathrm{lg1}) \quad (3.7)$$

$$\mathrm{c} = \arccos(\mathrm{a} + \mathrm{b}) \quad (3.8)$$

$$\mathrm{d} = \mathrm{R} * \mathrm{c} \quad (3.9)$$

, where R is the radius of the earth, lg1 is longitude coordinate of the start point, lt1 is latitude coordinate of the start point, lg2 is longitude coordinate of the endpoint and lt2 is latitude coordinate of the endpoint. In this formula, latitude and longitude must be degree values. As an example, calculate the distance between Hlaing Fire Station: latitude 16.850763 and longitude 96.124874 and Aung Myay Ter Zi Street latitude 16.83456345 and longitude 96.12692498 by spherical law of cosines. The radius of the earth is 6371 km.

$$\mathrm{lt1} = 16.850763, \mathrm{lg1} = 96.124874$$

$$\mathrm{lt2} = 16.83456345, \mathrm{lg2} = 96.12692498$$

$$a = \sin(16.850763) * \sin(16.83456345) = 0.08395188427$$

$$\mathrm{b} = \cos(16.850763) * \cos(16.83456345) \cos(96.12692498 - 96.124874)$$

$$= 0.9160480752$$

$$\mathrm{c} = \arccos(0.08395188427 + 0.9160480752 \ )$$

$$= \arccos(0.9999999595)$$

$$\mathrm{d} = 6371 * 0.0002846 = 1.8131866 \text{ km}$$

3.5.2.2 Haversine Formula

The haversine formula gives the great-circle distances between two points on the Earth's surface precised with longitude and latitude. It is a method that calculates the considered distance appropriately and accurately [5].

$$\mathrm{dlg} = \mathrm{lg2} - \mathrm{lg1} \quad (3.10)$$

$$\mathrm{dlt} = \mathrm{lt2} - \mathrm{lt1} \quad (3.11)$$

$$\mathrm{a} = \sin\text{^}2(\mathrm{dlt}/2) + \cos(\mathrm{lt1}) * \cos(\mathrm{lt2}) * \sin\text{^}2(\mathrm{dlg}/2) \quad (3.12)$$

$$\mathrm{c} = 2 * \arcsin\left(\min\left(1, \mathrm{sqrt}(\mathrm{a})\right)\right) \quad (3.13)$$

$$\mathrm{d} = \mathrm{R} * \mathrm{c} \quad (3.14)$$

where c is the great circle distance in radians and R is the radius of the earth value with 6371 Km [51], lg1 is longitude coordinate of the start point in degree, lt1 is latitude coordinate of the start point in degree, lg2 is longitude coordinate of the endpoint in degree and lt2 is latitude oordinate of the endpoint in degree. For example, calculate the distance between Hlaing Fire Station: latitude 16.850763 and longitude 96.124874 and Aung Myay Ter Zi Street Latitude 16.83456345 and longitude 96.12692498 by haversine formula.

Let $\quad lt1 = 16.850763,\ lg1 = 96.124874$

$lt2 = 16.83456345,\ lg2 = 96.12692498$

$dlg = 96.12692498 - 96.124874 = 0.00205098$

$dlt = 16.83456345 - 16.850763 = -0.01619955$

$a = \sin\text{^}2(-0.01619955\ /2) + \cos(16.850763) * \cos(16.83456345) *$
$\sin\text{^}2(0.00205098\ /2)$

$= 1.9984838e^{-8} + \ + 0.957063045 \ * \ 0.957144966 * \ 3.20344738e^{-10}$

$= 2.0278289^{-8}$

$c \ = 2 \ * \ \arcsin\left(\min\left(1, \text{sqrt}(2.0278289^{-8})\right)\right)$

$= \ 2 \ * \ \arcsin(\min(1{,}0.000142402))$

$= \ 2 \ * \arcsin(0.0001424) = 2 * \ 0.00014 = 0.00028$

$d \ = 6371 * 0.0002848 = \ 1.8144608$ km

The Haversine formula is more accurate than the spherical law of cosines formula in the problems associated with small distances [21].

3.6 Recent Improved Versions of Dijkstra's Algorithm

This section describes three improved versions of the original Dijkstra's Algorithm which proposed by other researchers.

3.6.1 Dijkstra's Algorithm Improved Version 1

The original method is considered on the appropriated data structure to store the information of the network before executing the shortest path search. When increased the number of nodes, it expands the correlation matrix for the number of elements by way of geometric series. It causes the certain drawback that it takes a lot of redundancy of space and searching efficiency can seriously reduce. This proposed algorithm is based on the searching strategy that presents the constraint functions r (n) for each searching node in state space in order to reduce the searching range by using core formula as follows.

$$r(n) = \omega \ * \cos(\theta_n) \tag{3.15}$$

$$D(n) = d(n) + r(n)\ ,\ \ -\frac{\pi}{2} \leq \theta_n \leq -\frac{\pi}{2} \tag{3.16}$$

where, $r(n)$ is the constraint function, ω is a weighted value denote the impact factor,θ_n is the angle between the start node to the present node vector and the start node from to the end node and $d(n)$ is the weight value of the shortest path from the start node to the present node. Let the vertices A, B, C, D, E, F, G, H and I are hospitals or medical service centers and the vertex 0 be the point that the ambulance service request has been made. Table 3.1 presents the shortest path using the original Dijkstra's algorithm. As shown in the table, the distance from vertex 0 to H is 1970 and the sequence of vertices along the path can determine by backtracking. The vertex H reaches from G, G from F, F from I, I from E, E from D, D from C, C from B, B from A and A to 0 and also the distances of close services and the distance of shortest route can also be considered by using the table.

Table 3.1 Shortest Path Result of Original Method

	0	A	B	C	D	E	F	G	H	I
0	0	∞	∞	∞	∞	∞	∞	∞	∞	∞
A	0	200_0	400_0	∞	∞	∞	∞	∞	∞	∞
B	0	200_0	400_0	∞	∞	∞	∞	∞	∞	∞
C	0	200_0	400_0	600_B	∞	∞	∞	∞	∞	∞
D	0	200_0	400_0	600_B	1050_C	∞	∞	∞	∞	∞
E	0	200_0	400_0	600_B	1050_C	1500_D	∞	∞	∞	∞
I	0	200_0	400_0	600_B	1050_C	1500_D	1560_E	1750_F	∞	1500_D
F	0	200_0	400_0	600_B	1050_C	1500_D	1560_E	1750_F	∞	1500_D
G	0	200_0	400_0	600_B	1050_C	1500_D	1560_E	1750_F	∞	1500_D
H	0	200_0	400_0	600_B	1050_C	1500_D	1560_E	1750_F	1970_G	1500_D

Table 3.2 shows the result of the same problem by using an improved algorithm. In this improved version, the searching range is reduced and the searching efficiency is upgraded because of the limitation of θ_n. When $d(n)$ is large, the value of impact factor should be big value because of the range of $\cos(\theta_n)$ is -1 to 1. The improved algorithm becomes original Dijkstra's algorithm when $r(n)$ is equal to zero or less than $d(n)$. Therefore, the constraint

value of the current node should not be overvalued and $r(n)$ should not be set to a too slight value. By using suffix attached along with the values, the path can determine by backtracking. The shortest path between 0 to I can know by checking the last row as I to D (from the suffix), then D to C (from row D and suffix C), C to 0 [4].

Table 3.2 Shortest Path Result of Improved Method

	0	A	B	C	D	E	F	G	H	I
0	0	∞	∞	∞	∞	∞	∞	∞	∞	∞
A	0	818.04_B	1502.11_0	∞	∞	∞	∞	∞	∞	∞
B	0	1.517_B	1502.11_0	717.64_0	∞	∞	∞	∞	∞	∞
C	0	1.517_B	1502.11_0	717.64_0	∞	∞	∞	∞	∞	∞
D	0	1.517_B	1502.11_0	717.64_0	2448.78_C	∞	∞	∞	∞	∞
E	0	1.517_B	1502.11_0	717.64_0	2448.78_C	1888.67_D	∞	∞	∞	2450_D
F	0	1.517_B	1502.13_0	717.64_0	2448.78_C	1888.67_D	744.04_E	1982.06_E	∞	2450_D
G	0	1.517_B	1502.11_0	717.64_0	2448.78_C	1888.67_D	744.04_E	1982.06_E	∞	2450_D
H	0	1.517_B	1502.11_0	717.64_0	2448.78_C	1888.67_D	744.04_E	1982.06_E	914.04_G	2450_D
I	0	1.517_B	1502.11_0	717.64_0	2448.78_C	1888.67_D	744.04_E	1982.06_E	914.04_G	2450_D

3.6.2 Dijkstra's Algorithm Improved Version 2

The computation of route distance between two locations is one of the most necessary complications on the road networks. Many people commonly face the route-finding problem while scheduling their trips with their own vehicles in unfamiliar areas [38]. In many recent years, there are many applications were implemented for solving this problem by developing different routing approaches. Although, the various approaches for pathfinding are developed to calculate the shortest path on the road networks it still remains the problems. After analyzing the existing path-finding algorithms, it is observed that Dijkstra's algorithm is the most suitable to calculate the shortest path in road networks. However, there are some weaknesses and need to modify the approach to improve efficiency and to decrease the computational complexity.

For this reason, a new algorithm is proposed and namely as Modified Dijkstra's Shortest Path algorithm (MDSP) is proposed by using multiple parameters were added and used to find the suitable shortest path rather than the use of single parameter in the existing algorithm. The flow of the proposed MDSP algorithm is described as follows:

```
Algorithm: Modified Dijkstra's Algorithm
begin
for each v in G do
        alt_path[i]:= Null
        d [v] = infinity
        weight_update(choice);
            if c = d then;
                else if c = t then d := d * s;
                    else d:= d* z;
                return d;
            end
    for each vertex v in G do
        if  v == start || v= end then
          for each neighbour u of v do
            if  alt_path[i]  >  dist[u]  + dist (u, v) then
                alt_path[i]:= dist [u] + dist (u, v);
            end if
          end for
         end if
    end for
end for
end
Algorithm : Adding Multiple Parameters
        begin
            if c = d then;
              else if c = t then
                  d := d * s;
```

```
                    else d:= d* g;
                    return d;
          end
```

where

c = choice

d = distance

t = time

g = congestion factor

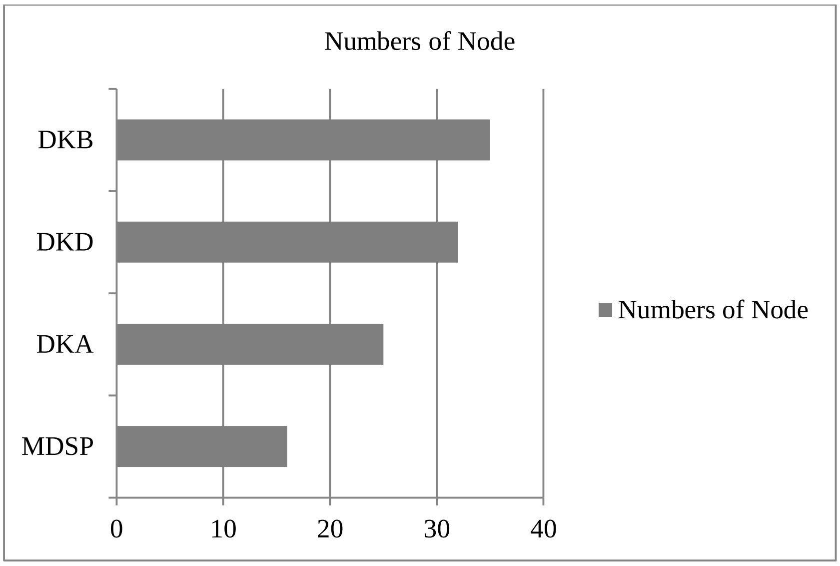


Figure 3.19 Number of Node Comparison

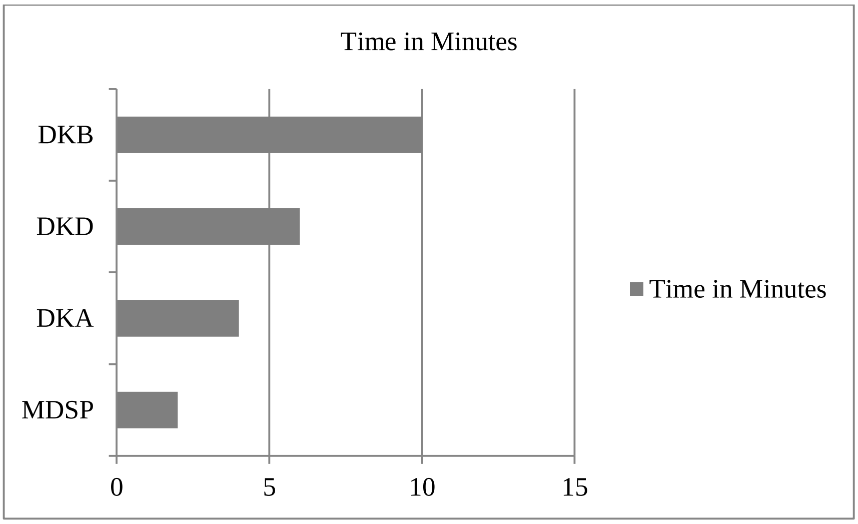


Figure 3.20 Number of Time Comparison

The effectiveness of the MDSP algorithm is proved in terms of nodes (it shows the shortest path) and time factor using the Jaipur city database. To prove the competence of proposed algorithm, compared it with the current modified Dijkstra's algorithms namely, DKB (Dijkstra's Algorithm with Buckets), DKD (Dijkstra's Algorithm with Double Buckets), DKA (Dijkstra's algorithm with Approximate Buckets). The comparison is computing by taking time as shown in Fig 3.19 and the number of nodes to select the efficient shortest path is shown in Fig 3.20. The testing analysis prove that the proposed shortest path algorithm MDSP calculates the effective shortest path by taking the smallest number of nodes and also it takes small amount of calculation time [54].

3.6.3 Dijkstra's Algorithm Improve Version 3

There are many different matrix structures to characterize the graph based on the properties of graph demonstration [27]. Based on the structures of matrix representation, the new algorithm is proposed to determine the candidate subgraphs and clips every subgraph that is either inaccessible from the specified source vertex or does not lead to the particular destination by using the information intrinsic in the matrix and inverse matrix structure illustrations of the graph. The structure of the reverse matrix represents the graph with leaves stored first and the advantage of this structure is that it stores all paths that can reach the node from the source nodes. The proposed method is summarized as follow:

1. Construct the main matrix to represent that includes node, distance, predecessor node, and temporary distance to maintain the minimum distance of the predecessor node in the graph.
2. Construct the reverse matrix to represent the graph rooted with destinations.
3. Mark all candidate nodes in the main matrix by traversing the graph starting from the given destination node. It is possible to use the reverse matrix to mark all the candidate nodes.
4. After marking the candidate node, all neighbors edges are added by visiting all nodes listed in the next column of the current node start form the given source node. Accumulated the subpath path weight by adding the current node weight. When revisiting the using another edge with the new weight, directly jump to coordinates' pointer in the main graph matrix and compare the new weights. And also keep the minimum path distance with updated predecessor nodes. These nodes enable to trace back the shortest path from the present node to the source node.

The proposed algorithm is improved with candidate subgraphs that ignore other irrelevant graph parts looking for the shortest path as illustrated in following the algorithm.

```
Shortest_Path_Candidates(G,  Mark, Rmatrix)
        {
            initialize Mark[i] to 0
            node=Target
            Mark[node]=1
        for every vertex next to node in Rmatrix
            if vertex != coordinates_pointer
                node =vertex
            else
                node =Rmatrix[coordinates_pointer]
                Mark[node]=1
            end if
        end for
            dist= FindShortest(GraphMatrix, Mark, Source)
        }
  FindShortest(Gmatrix,Mark,Source)
        {
            for each vertex v in Gmatrix
                initialize dist[v] = ∞
                pred[v] = undefined
            end for
        dist[Source] := 0
        MarkQ = set of Marked nodes in Gmatrix ordered as of depth first search visits
        while MarkQ is not empty and u != t
        u= vertex in MarkQ with smallest distance in dist[ ];
        remove u from MarkQ
        if (u == Target ||dist[u] == infinity)
            then break
        end if
        for each neighbor v of u and v is in MarkQ
```

```
        if v is coordinate pointer <a,b>
                v=GraphMatrix[a,b]
        end if
p = dist[u] + dist_between(u, v)
        if p<dist[v]
                dist[v] = p
                pred[v] = u
                update v in MarkQ
        end if
end for
end while
return dist
}
```

The experimental result shows the evidence that the proposed method outperforms the conventional original method with performance saving ratio as shown in Table 3.3. The performance of proposed algorithm shows the significant cost saving in random generated graphs with different sizes range from 100 to 500 nodes. In the most of trial, saving performance occur in dense graphs and more in sparse ones.

Table 3.3 Saving Performance of Proposed Method

Nodes	Sparse (%)	Dense (%)
100	0.8907436	0.7525469
150	0.646433	0.6867926
200	0.5183573	0.6371958
250	0.5245072	0.6202232
300	0.5218704	0.6107841
350	0.5804231	0.4876443
400	0.5880742	0.4801536
450	0.5670171	0.4614253
500	0.6200705	0.4900764

3.7 Chapter Summary

This chapter mainly discussed the concepts, nature, and theory related to a route-finding system based on GIS. Firstly, the relation between graph theory and GIS information is discussed in detail and the application areas that applied with graph theory and the recent improved route finding methods by other researchers are described. And then, three types of data models are discussed with related examples to represent the road network and also discussed two types of route-finding problems. The two main factors of the distances calculation and searching strategies for the graph are explained with related examples. Based on the focused issue and the nature of proposed system, the researchers have used the above methods to represent the network as graph, to calculate the distance between two locations.

CHAPTER 4

DIJKSTRA'S ALGORITHM AND MODIFIED DIJKSTRA'S ALGORITHM

This chapter discusses conventional Dijkstra's algorithm and the improved versions of the proposed work.

4.1 Original Dijkstra's Algoirhtm

Edsger Dijkstra, Dutch computer scientist, created Diksjtra's algorithm in 1959 [52]. It is used as the graph searching algorithm and solved the single source path finding problem in the graph that have no negative edge path costs. Dijkstra's algorithm is known as the single-source shortest path algorithm because it calculates the shortest path from the given source node to all the remaining nodes in the graph. Dijkstra's algorithm is one of the most popular methods and it has been usually used by researchers to solve the problems [16]. This approach is frequently used in routing and additional protocol network [18]. Pseudo code of original Dijksta's Algorithm is described as follows:

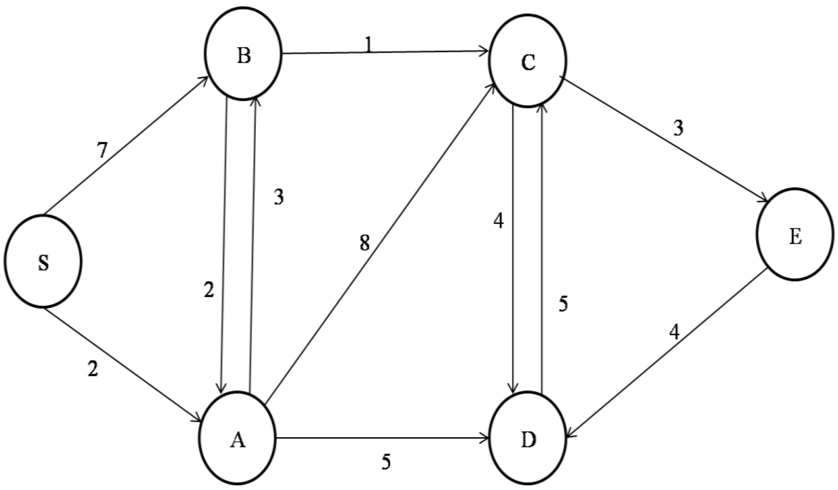
```
function Dijkstra(Graph, s)
            for each vertex v in Graph:
                        d[v] := infinity ;
            end for ;
 d[s] := 0 ;
  while Graph is not empty:
            u := node in Graph with minimum d ;
                         if d[u] = infinity:
                                    break ;
                        end if ;
 remove u from Graph ;
 for each neighbor v of u in Graph:
             temp_d := d[u] + d(v, u) ;
                        if temp_d < d[v]:
                                    d[v] := temp_d ;
                        end if ;
```

end for ;

end while ;

return d[ ] ;

The above Dijkstra's Algorithm can be explained and easy to understand by using the following example.



Initialization

Source is S

Queue => {S,A, B, C, D, E}

i	S	A	B	C	D	E
d[i]	0	∞	∞	∞	∞	∞

Iteration 1

u =S

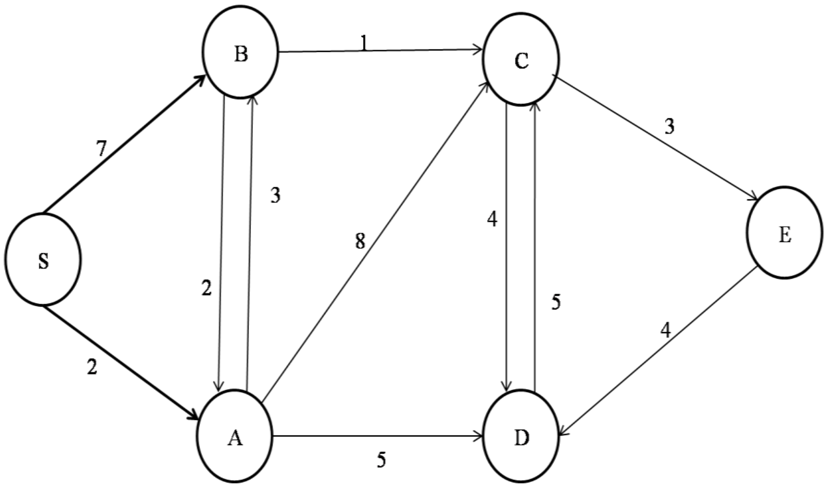
v => A,B

d[A]= min {∞, 0+2} = min {∞,2} =2

d[B]= min {∞, 0+7} = min {∞,7} =7

Queue => {A, B, C, D, E}

i	S	A	B	C	D	E
d[i]	0	2	7	∞	∞	∞



Iteration 2

u =A

v => B,C,D

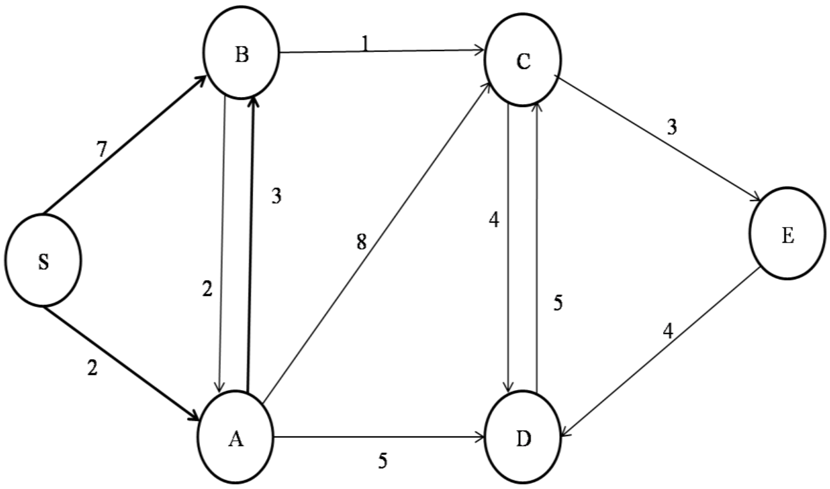
d[B]= min {7, 2+3} = min {7,5} =5

d[C]= min {∞, 2+8} = min {∞,10} =10

d[D]= min {∞, 2+5} = min {∞,7} =7

Queue => {B, C, D, E}

i	S	A	B	C	D	E
d[i]	0	2	5	10	7	∞



Iteration 3

u =B

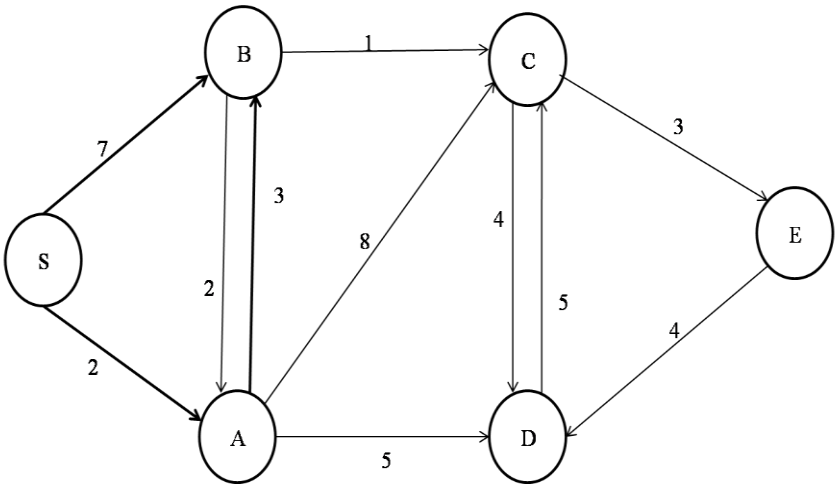
v => A,C

d[A]= min {2, 5+2} = min {2,7} =2

d[C]= min {10, 5+1} = min {10,6} =6

Queue => { C,D,E}

i	S	A	B	C	D	E
d[i]	0	2	7	6	7	∞



Iteration 4

u =C

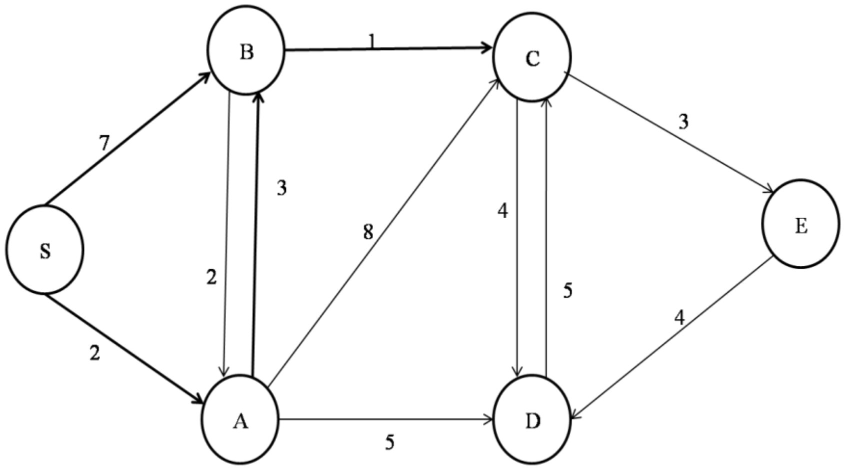
v => D, E

d[D]= min {7, 6+4} = min {7,10} =7

d[E]=min{∞, 6+3}=min{∞,9}=9

Queue => {D,E}

i	S	A	B	C	D	E
d[i]	0	2	7	6	7	9



Iteration 5

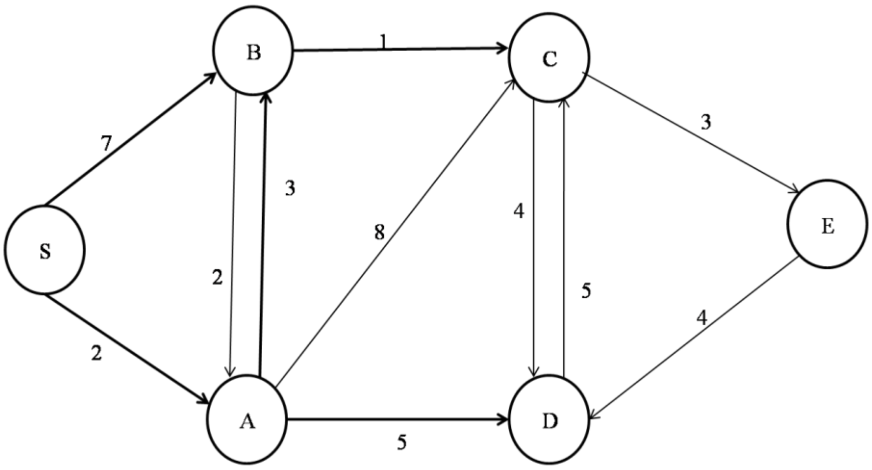
u =D

v => C

d[D]= min {6, 7+5} = min {6,12} =6

Queue => {E}

i	S	A	B	C	D	E
d[i]	0	2	7	6	7	9



Iteration 6

u =E

v =>D

d[D]= min {7, 9+4} = min {7,13} =7

i	S	A	B	C	D	E
d[i]	0	2	7	6	7	9

Queue => {ϕ}

The algorithm is terminated. The final result of original Dijkstra's Algorithm is demonstrated in Fig. 4.1 and described in detail as Table 4.1.

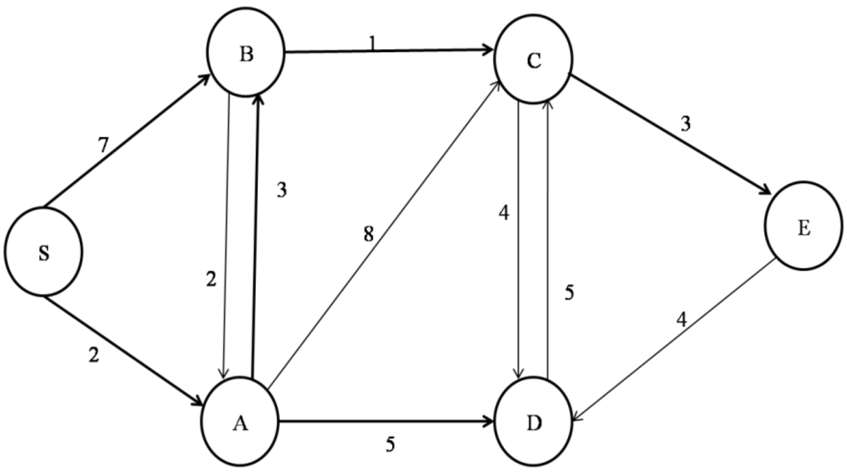


Figure 4.1 Path Result of Given Graph

Table 4.1 Result Given Graph by Using Original Dijkstra's Algoirhtm

Form-To	Path	Total Distance
S to A	S-A	2
S to B	S-B	7
S to C	S-A-B-C	6
S to D	S-A-D	7
S to E	S-A-B-C-E	9

According to this example calculation, Dijkstra's algorithm makes n iterations to get the required shortest path result. If all vertices in the given graph have been visited, then the algorithm finishes [37].

4.1.1 Time Complexity of Original Dijkstra's Algorithm

This variant maintain the priority queue of items, each item have the value. The following operations are performed during the maintaining list [17]. Insert operation: add an item with value to priority queue extract min operation: extract the item in the priority queue that have the minimum value and decrease key operation that replace the value of an item in priority queue with smaller value. Therefore, Dijkstra's Algorithm have the following frequency of calls on these operations $O(1)$ times for insert operation, $O(n)$times for extract min operation and $O(n)$ times for decrease key operation where $n$the number of nodes is and m is the number of edges. Therefore, the total complexity of Dijkstra's Algorithm is $O(n)$[10].

4.1.2 Disadvantages of Dijkstra's Algorithm

1. The first weakness of original Dijkstra's algorithm is that it does a blind search so it consumes a lot of time and waste of memory resources [43].
2. The second weakness is that it doesn't consist and consider the condition of link (edges). For example, the street (edges) is one-ended or close or wide enough to enter.

4.2 Proposed Route Finding Method (Modified Dijkstra's Algorithm)

In order to handle the large amount of road network data, researchers try to advance the original route-finding method or develop their own methods. There are many ways and methods to get the optimal route into a large number of nodes and to speed up the searching process in recent works. Nevertheless, this problem remains open to researchers and it is a real challenge until this time. The bottleneck operation in conventional Dijkstra's Algorithm is finding the minimum temporary label to use in extending the shortest path. The implementation of route finding methods can vary based on using data structure.

4.2.1 Data structure of Proposed Method

The data structure of advanced Dijkstra's Algorithm uses a priority queue to store the node name and its tentative distance. A priority queue can be applied by using various data structures like an array, a linked list, or a binary search tree. In order to make all of the procedures very efficient in our implementation, heap data structure is used to store the nodes and their values. The heap is a type of tree that have the property to store the value of all node is less than or equal or equal to the value of its child nodes, which is in terms of heap order invariant. The tree of nodes is accessed by a distinguished pointer to the node with the smallest distance value. Nodes can be either visited or unvisited but root nodes are never visited. In the proposed route finding method, there are three fundamental operations such as insert operation, decrease_key operation and delete_min operations described as follows:

i. INSERT OPERATION

Insert (H, key)

H_size = H_size+1

H [H_size] = -∞

Increase_key (H, H_size, key)

ii. DECRESE_KEY OPERATION

Decrease_key (H, i, key)

H[i] = key

iii. EXTRACT_MIN OPETATION

Extract_min (H)

min = H[1]

H [1] = H [H_size]

H_size = H_size-1

return min

4.2.2 Processing step and Pseudo code of Proposed Method

The proposed method proceed as the following steps:

1. Initialize the distance of source node with '0' and assign the distance of remaining nodes in the graphs as '∞'.
2. Source node is set as current node and other nodes are mark as unvisited.
3. Calculate the temporary distance of all neighbor nodes of the current node by summing its distance and the weights of the edges of neighbor nodes and also and check the edge status of neighbor nodes.
4. If the calculated distance is smaller than the present distance and the edge status is equal to '1' then overwrites the old distance and mark the node is visited. The visited node will not be checked again and the distance is final and minimal.
5. Set the unvisited neighbor node with minimal temporary distance is marked as current node and continue step 3.
6. If the current node is equal to target or H_Queue is empty, the algorithm will terminate.

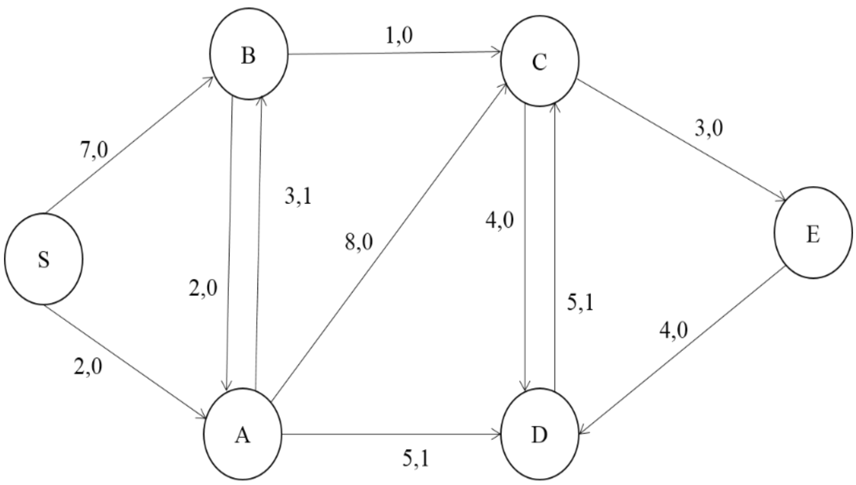
The pseudo code of proposed route finding method is described as follows:

function MDijkstra(V, s)

d[s] ←0

```
target ← t
for  all v ∈ V
        do  d[v] ←infinity
end for
S←∅
H_Queue←V
while H_Queue ≠∅
      u ← ExtractMin (H_Queue,d)
      S←S∪{u}
            if u==target
                        break;
            end if
       for all v ∈ neighbors[u] and status(u,v) != 1
               if   d[v] > d[u] + w(u, v)
                        then      d[v] ←d[u] + w(u, v)
               end if
            end for
            end while
return d
```

The proposed method can be explained and easy to understand by using the following example.



Initialization

Source is S

Destination is C

H_Queue => {S, A, B, C, D, E}

i	S	A	B	C	D	E
d[i]	0	∞	∞	∞	∞	∞

Iteration 1

u =S

v => A,B

status(S,A) ==0

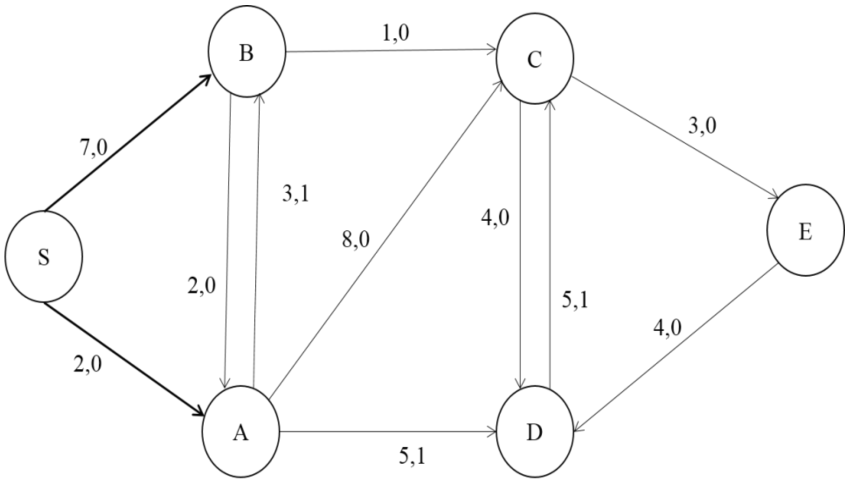
d[A]= min {∞ , 0+2} = min {∞ ,2} =2

status(S,B) ==0

d[B]= min {∞ , 0+7} = min {∞,7} =7

H_Queue => {A, B, C, D, E}

i	S	A	B	C	D	E
d[i]	0	2	7	∞	∞	∞



Iteration 2

u =A

v => B,C,D

status(A,B)==1

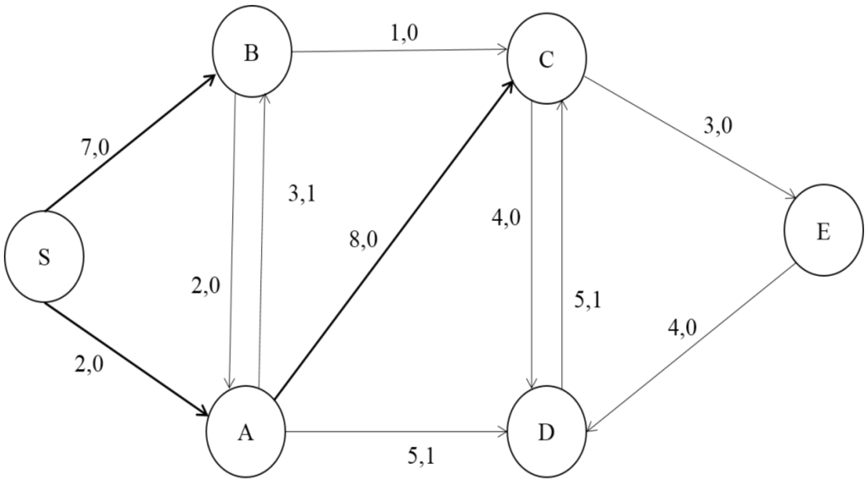
status(A,C)==1

d[C]= min {∞, 2+8} = min {∞, 10} =10

status(A,D)==1

H_Queue => {B, C,D, E}

i	S	A	B	C	D	E
d[i]	0	2	7	10	∞	∞



Iteration 3

u =B

v => A,C

status (B,A) ==0

d[A]= min {2, 7+2} = min {2,9} =2

status (B,C) ==0

d[C]= min {10, 7+1} = min {10, 8} =8

H_Queue => {C,D,E}

i	S	A	B	C	D	E
d[i]	0	2	7	8	∞	∞

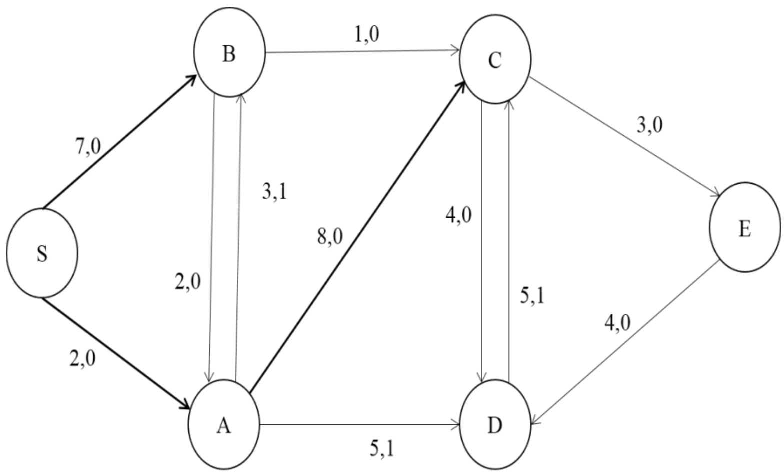


Figure 4.2 Result Path of Graph by Proposed Method

When the process of iteration 4 before start, u = C and target is found. So the algorithm is terminate.

Table 4.2 Total distance and path between selected source and destination

Form-To	Path	Total Distance
S to C	S-A-C	8

The final result of modified Dijkstra's Algorithm is demonstrated as in Fig. 4.4 and described detailed as Table 4.5.

4.2.3 Time Complexity of Proposed Method

There are three basic operations in the proposed route finding algorithm.

i. Insert Operation: Add a node with its values to the priority queue and the time for this operation is O (log n).
ii. Extract_Min Operation: Extract the specified node with smaller value and this operation also needs time O (1).
iii. Decrease_Key Operation: Replace the node which has the minimum value and this operation takes time O (log n).

Consequently, the worst case complexity of proposed method is O (log n).

4.2.4 Advantages of the Proposed Method

1. The first benefit of proposed algorithm is that it terminates when the target node is found so it reduce the consumption of a lot of time and memory resources.
2. The second benefit is that it considers the condition of link (edges). For example, the street (edges) is one-ended or close or wide enough to enter. Therefore, the delay that caused by bad condition of road can avoid.
3. The third benefit is that it provides not only the accurate and best route but also consider the condition of the street in the whole road network which did not include in the previous modified versions.

4.3 Analysis of the Proposed Method and the Original Method

The proposed method and the original method are analyzed by comparing their run time complexity and the number of iterations.

4.3.1 Run Time Analysis of Two Methods

The difference of run time complexity that operates in original method and proposed method is shown in Fig. 4.3. and Table 4.3 show the complexity comparison of each operations in two methods .

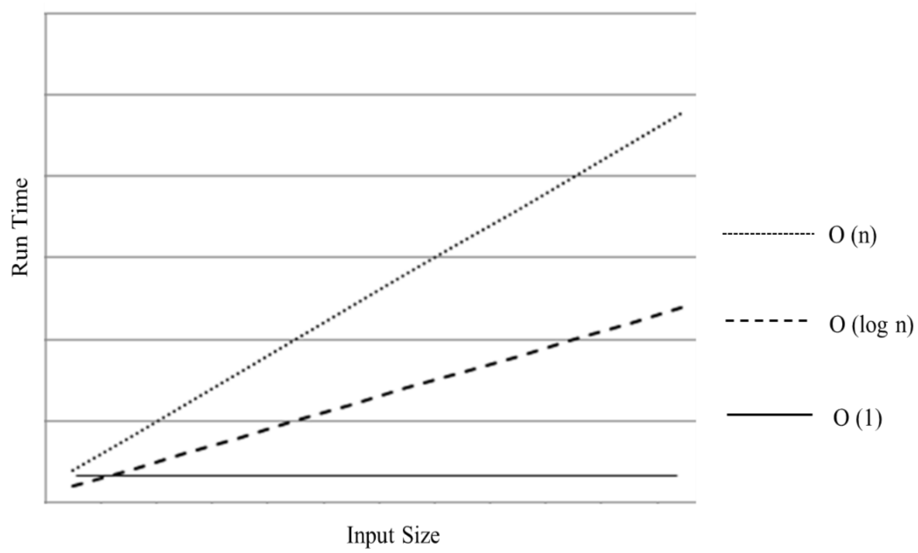


Figure 4.3 Illustration of Runtime Complexity

Table 4.3 Runtime Complexity of Two Methods

Original Dijkstra's Algorithm	Proposed Dijkstra's Algorithm
Use Array Data Structure	Use Heap Data Structure
Insert Operation : O (1)	Insert Operation : O (log n)
Extract_Min Operation : O (n)	Extract_Min Operation : O (1)
Decrease Key Operation : O (n)	Decrease Key Operation : O (log n)

4.3.2 Iteration Analysis of Two Methods

The proposed method and the original method is analyzed and tested with 5 sample tests based on number of iterations as described in the following. These tests have shown the difference between original method and the proposed method when these two methods are applied on the same data based on the number of iterations process. In sample test 1, 2, 3 and 4, the number of iteration times by proposed the method is smaller than the original one and the number of iterations in test 5 and 6 is larger than the original method.

Sample Test 1

Original Method Result

Number of Iteration :533

Result
:1828.0/100:92:90:91:94:95:89:56:79:78:29:30:22:23:21:17:15:14:11:12:399
:398:395:394:391:385:388:802:804:718:719:717:794:700:

Proposed Method Result

Number of Iteration :443

Result
:1975.0/100:101:99:821:820:819:818:817:816:24:18:7:14:11:12:399:400:40
2:397:387:192:803:805:193:804:718:719:717:794:700:

In sample test 1, the source node is 100 and the destination node is 700. In this case, the original method gives the optimal route result with total distance 1828 km and all node id that consisted along the route result by processing 533 number of iterations, but the proposed method can output the optimal route result or different route with 433 iterations by avoiding the condition which can cause delay on the way. The edge status between node id 100 and 92 in the result of original method is 1 because this street is one ended condition in the real road network data and it is the main issue to solve by using proposed method. The proposed method can solve the issue and produce the optimal route result with less iteration.

Sample Test 2

Original Method Result

Number of Iteration: 114

Result:

837.0/50:90:91:93:820:819:818:817:816:24:18:7:10:

Proposed Method Result

Number of Iteration: 75

Result:

1092.0/50:90:92:100:101:99:821:820:819:818:817:816:24:18:7:10:

In sample test 2, the source node is 50 and the destination node is 10. In this case, the original method gives the optimal route result by processing 114 number of iterations and the proposed method can give the optimal route result with 75 iterations. In this sample test, the edge status between node id 90 and 91in the result of original method is 1 because this street is one ended.

Sample Test 3

Original Method Result

Number of Iteration: 413

Result:

2175.0/300:379:380:342:798:750:751:370:752:753:754:755:756:757:367:366:38:36:395:398:399:400:

Proposed Method Result

Number of Iteration: 357

Result:

2175.0/300:379:380:342:798:750:751:370:752:753:754:755:756:757:367:366:38:36:395:398:399:400:

Sample Test 4

Original Method Result

Number of Iteration :132

Result :

952.0/100:104:105:106:109:113:112:801:800:799:747:61:60:

Proposed Method Result

Number of Iteration :89

Result :

952.0/100:104:105:106:109:113:112:801:800:799:747:61:60:

In sample 3 and 4, the two methods give the same result but the proposed method can produce the result by processing less iteration times.

Sample Test 5

Original Method Result

Number of Iteration: 69

Result:

615.0/50:90:91:94:95:89:56:79:78:29:30:

Proposed Method Result

Number of Iteration: 81

Result:

1133.0/50:90:92:100:101:99:821:820:819:818:817:816:24:34:31:30:

Sample Test 6

Original Method Result

Number of Iteration: 144

Result:

930.0/5:9:289:288:290:813:814:815:816:817:71:72:75:77:79:78:80:

Proposed Method Result

Number of Iteration: 166

Result:

1040.0/5:9:289:288:290:813:814:7:18:24:34:31:33:29:78:80:

In sample test 5 and 6, the proposed method produces the optimal route result with large number of iterations than the original method. However, the proposed method must provide accurate and effective optimal route result. Table 4.4 shows the number of iteration comparison by using two methods in each sample test.

Table 4.4 Comparison of Sample Tests

Sample Test	Source Node and Destination Node	Number of Iterations by Original Method	Number of Iterations by Proposed Method
Test 1	100 and 700	533	443
Test 2	50 and 10	114	75
Test 3	300 and 400	413	357
Test 4	100 and 60	132	89
Test 5	50 and 30	69	81
Test 6	5 and 80	144	166

4.4 Chapter Summary

This chapter is mainly focused to compare the nature, working process, iteration steps and time complexity of proposed route finding method namely as Modified Dijkstra's Algorithm and original Dijkstra's algorithm. By adding additional variables and conditional statements, the proposed algorithm has improved the original one to decrease the time complexity and memory consumption. The original method processes from the source node to all other remaining nodes in the graph, so the processing time and memory usage are larger than the proposed method that terminates the algorithm when the target is found. According to the result from the detailed calculation example of two methods, the iteration steps of the proposed method are smaller than the original method. Lastly, based on the result of sample tests the proposed method can perform faster in route finding process by reducing iteration time and processing time in most cases.

CHAPTER 5

GENERAL ARCHITECTURE AND DATA PREPARATION

The main objective of the proposed work is to find out the optimal route between the incident location and the emergency rescue teams location within Yangon Region. It is very essential to save lives by giving care instantaneously when the emergency case occurred. To move the injury people from the hazard location to the hospitals, the work of medical service centers, fire stations and emergency rescue teams, within a short time, is a critical task. During this situation, the address information of phone call could not conclude the accident location precisely. And it is also difficult to determine the close emergency services to give the required facilities immediately. It is also vital to give the effective evacuation process and recovery actions, the optimal route to go to the victim location. Therefore, the optimal route finding system is proposed to solve the above problems.

5.1 General Architecture of Proposed System

Fig. 5.1 shows the general architecture of the proposed system. The proposed system is intended to identify the exact location of the emergency area and find the nearest fire stations. And then the proposed system finds the optimal path between nearest fire station and accident case location.

Figure 5.1 General Architecture of Proposed System

5.2 Overview of the Proposed System

The overview of the system processing steps is shown in Fig. 5.2. By using a regular phone call or a message, people can report the address information of the location of victim to the emergency rescue teams. After receiving this information, the system verifies the location on Google Map. The system will match the address of incident place with coordinate information to determine the precise place. And then, the system will offer the nearest emergency service locations to get a quick response and effective services. Finally, the system calculates the optimal route between the emergency service teams and the accident location by using the proposed advanced Dijkstra's Algorithm. Once knowing the optimal route to go to the victim location, the emergency trucks can reach there in a short time without delaying condition. So, the damage level can be decreased and the valued lives and properties can also be saved.

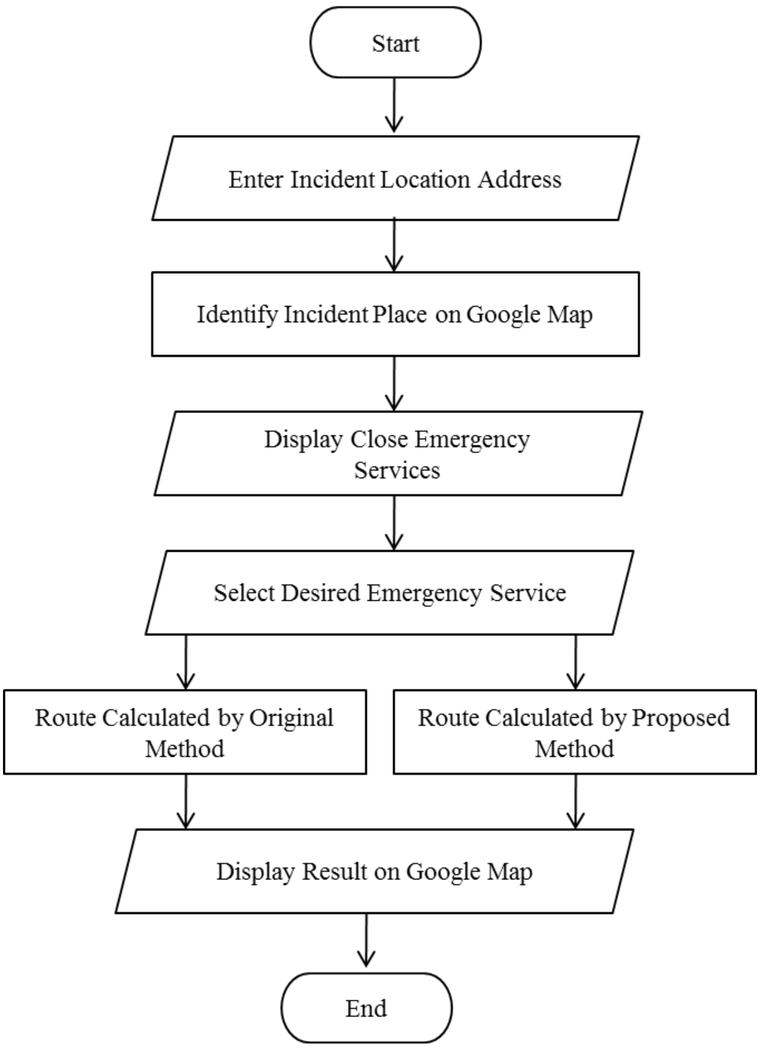


Figure 5.2 Overview of Proposed System

5.3 Study Area

The proposed system is mainly developed for Yangon Region which is the large city and constructed with the complicated unstructured road network. It is situated at latitude 16.80528 and longitude 96.15611 with 10,171 km^2.

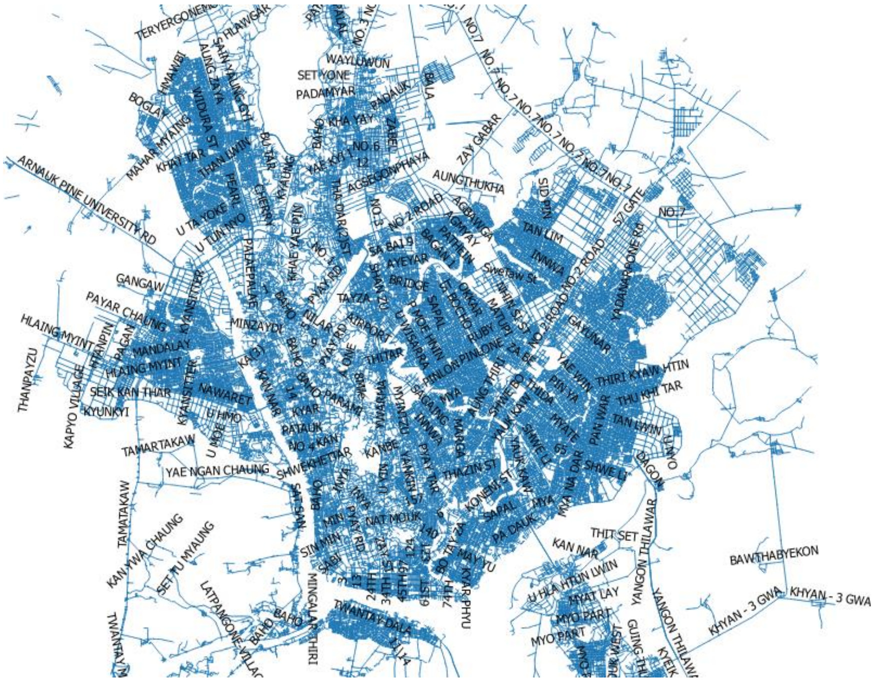


Figure 5.3 Road Network of Yangon Region

Although Yangon Region comprises 45 townships [67], this proposed system encompasses on 33 townships administrated by Yangon City Development Committee. In this proposed work, the road network is created with total number of edges 96780 and the number of vertices 32885. Fig. 5.3 shows the road network of the Yangon Region.

5.4 Data Creation for Proposed System

To develop the well-organized database of the proposed system, data are collected from related emergency departments and checked by using Google Map and GPS devices. The location of emergency services such as fire stations, hospitals, and police stations are marked as emergency service points on Google Map and used to find the close emergency services for hazard location.

5.4.1 Data Creation Tool

For creating the vector map of the Yangon region road network, QGIS (Quantum Geographical Information System) is used as data creation tool. QGIS functions as a geographic information system software that allows users to analyze and edit spatial information and to compose and export maps. QGIS incorporates with other open-source GIS packages, including PostGIS, GRASS GIS, and Map-Server. QGIS supports vector data that is warehoused as either point, line, or polygon features and multiple formats of raster images are supported and it can also geo-reference the images.

5.4.2 Data Collection for Tested Region

In general, there are 34 fire stations in Yangon Region. Some sample data for fire emergency service locations are shown in Table 5.1 with related geo-location and contact information. There are many ways to identify the hazard location by using the residential address, the nearest landmark, telephone number, etc. In this proposed work, the incident location is identified by the street name as address information. Table 5.2 shows some sample data of the road network. To compute the optimal route, the road network table is generated with suitable attributes and applied. Table 5.3 describes the data creation of a road network to evaluate the optimal route. In the status column, '1' represents the streets are narrow, one-ended or one way and '0' represents that street can be used.

Table 5.1 Sample Geo-location and Contact Information of Fire Stations

No	Name	Latitude	Longitude	Contact Info
1	Central Fire Station	16.779591	96.152645	01-252011
2	Hmawbi Fire Station	17.106246	96.060307	01-620030
3	Kyauktada Fire Station	16.776082	96.158498	01-252022,01-252011
4	Ahone Fire Station	16.8366687	96.0762595	01-220802
5	Kyeemyindaing Fire Station	16.812479	96.122159	01-534825,09-73164126
6	Sanchaung Fire Station	16.804094	96.133036	01-527099,01-536689
7	Tarmway Fire Station	16.803593	96.173997	01-554778,01-54841
8	Tarmway_B Fire Station	16.810659	96.174664	01-554893
9	Dawbon Fire Station	16.782553	96.187315	01-553021

10	Thaketa Fire Station	16.793132	96.203339	01-547040
11	South Okkalapa Fire Station	16.841023	96.183119	01-699149
12	Thingangyun Fire Station	16.832601	96.197814	01-562677
13	North Okkalapa A Fire Station	16.895225	96.157143	01-688149
14	Shwpaukkan Fire Station	16.928111	96.184485	09-421080164,01-699151
15	North Okkalapa Fire Station	16.917345	96.159221	09 -425356252
16	Waibargi Fire Station	16.916289	96.14914	01-699378
17	Insein Fire Station	16.886802	96.101259	01-640070,01-40987
18	Hlaing Fire Station	16.850763	96.124874	01-519578
19	Mayangon Fire Station	16.864666	96.120911	01-661501
20	Bayin Naung Fire Station	16.863489	96.107613	01-681648
21	Municipal Fire Brigade	16.871611	96.159921	01-667281
22	FSD-HQ	16.869806	96.154904	01-584060
23	Dagon Seikkan Fire Station	16.845701	96.265748	09-32046428,09-3120294
24	Hlaingtharyar_B Fire Station	16.875918	96.068839	01-645017
25	Mingalardon Fire Station	17.046965	96.140114	09-448011101,01-600178
26	South Dagon Fire Station	16.854688	96.223212	01-590071
27	North Dagon Fire Station	16.959331	96.295907	01-599467
28	Shwepyithar Fire Station	16.97397	96.076451	01-611014
29	Shwepyithar_B Fire Station	16.959878	96.076919	09-4208544
30	Thaketa_B Fire Station	16.807593	96.21824	09 -5154137,09- 556834
31	Hlaingtharyar Fire Station	16.87198	96.043671	01-707550,09 73206341
33	Mingalar Taung Nyunt Fire Station	16.8199146	96.1615491	01-8610702
34	East Dagon Fire Station	16.89895	96.2020095	01-585460

Table 5.2 Sample Data of Road Network

Node-ID	Street Name	Latitude	Longitude
100	PARAMI	16.85749	96.12123
101	PARAMI	16.8575	96.12079

102	PARAMI	16.85746	96.11966
103	OAK KYIN BUTER YONE	16.85566	96.12186
104	OAK KYIN BUTER YONE	16.85546	96.12062
105	OAK KYIN BUTER YONE	16.85567	96.12311
108	NO 2 MARLAR MYAING	16.85584	96.11758
111	NO 2 MARLAR MYAING	16.85623	96.12115
113	BAYINTNAUNG	16.85459	96.10823
114	BAYINTNAUNG	16.85421	96.10818
116	BAYINTNAUNG	16.85549	96.10833
119	THIRI MYAING	16.85247	96.11934
120	THIRI MYAING	16.85286	96.11926
121	THIRI MYAING	16.85313	96.11921
122	THIRI MYAING	16.85318	96.1192
123	MAHAR SWE	16.85319	96.11958
124	MAHAR SWE	16.8532	96.12004
125	MAHAR SWE	16.85315	96.12062

Table 5.3 Sample Data for Route Calculation

From Node	To Node	Distance	Status
0	32499	82.69307031	0
1	32502	22.74712508	0
2	32502	5.880056487	0
2	3	16.69534908	0
3	2	16.69534908	0
3	11116	62.835	0
4	32276	55.0181062	0
5	12023	475.442	0
5	6	169.9567521	0
6	8	97.78493555	0
6	5	169.9567521	0

7	8	384.665	0
8	7	384.665	0
8	12136	86.28250284	0
8	6	97.78493555	0
9	22563	16.54258404	0
10	11	48.85064811	0
10	22563	115.044	0
11	12312	30.88405041	0
11	10	48.85064811	0
15	16	6.06549	0
15	23983	301.546	0
83	655	46.361	0
83	84	42.869	1
83	87	42.952	0
86	85	60.2185	1
86	107	17.2074	0
86	91	21.6633	0
89	649	44.6517	0
89	85	30.9629	1
89	88	36.9421	1
89	111	135.486	0

5.5 Chapter Summary

This chapter mainly describes the general architecture and system overview of the proposed work. Data creation tool, the study area of the proposed work and some sample data such as road network data, emergency services location data and route calculation data used in the system are also described.

CHAPTER 6

IMPLEMENTATION AND PERFORMANCE EVALUATION

The proposed optimal route finding system for emergency vehicles is mainly intended to save lives and properties and reduce the damage level by providing the exact location of the emergency case, the emergency service places which locate near the incident site and the minimum delay route along the way to go to the emergency location. The system is implemented as a web application system and the overview of the proposed web application structure is shown in Fig. 6.1.

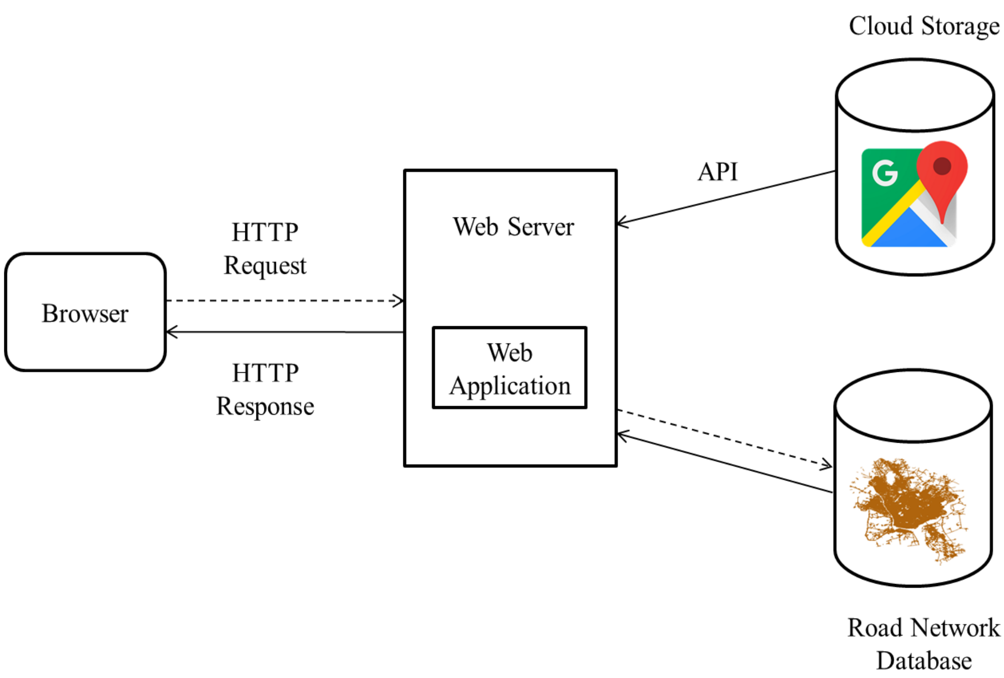


Figure 6.1 Overview of Proposed Web Application Structure

6.1 Proposed System Demonstration

The proposed system is implemented by using client-server architecture where a request having two locations as source and destination is sent from client to server and the optimal route is returned to the client as a reply from server. For that reason, the functional design of the proposed work includes the significant components as shown in Fig. 6.2. The user interface consists of client end to take input for processing. Web application and database consists of the server site. The system process basically takes incident location as a start point and emergency service location as a target point, and then computes the optimal route by using conventional Dijkstra’s Algorithm and Advanced Dijkstra’s Algorithm with detailed intermediated junctions and roads within the particular area. And also the system provides close emergency services which locate near the incident location. The database of the system contains the whole road network information of Yangon Region in terms of junctions and roads and emergency service locations. The client site sent the input containing source and destination points for the requested optimal route to the server site. When the server received the request data as input and computes the request optimal route by interacting with the database by using SQL queries to acquire the necessary information for optimal route calculation. After calculating the request from the client end, the results are responded to display the user interface.

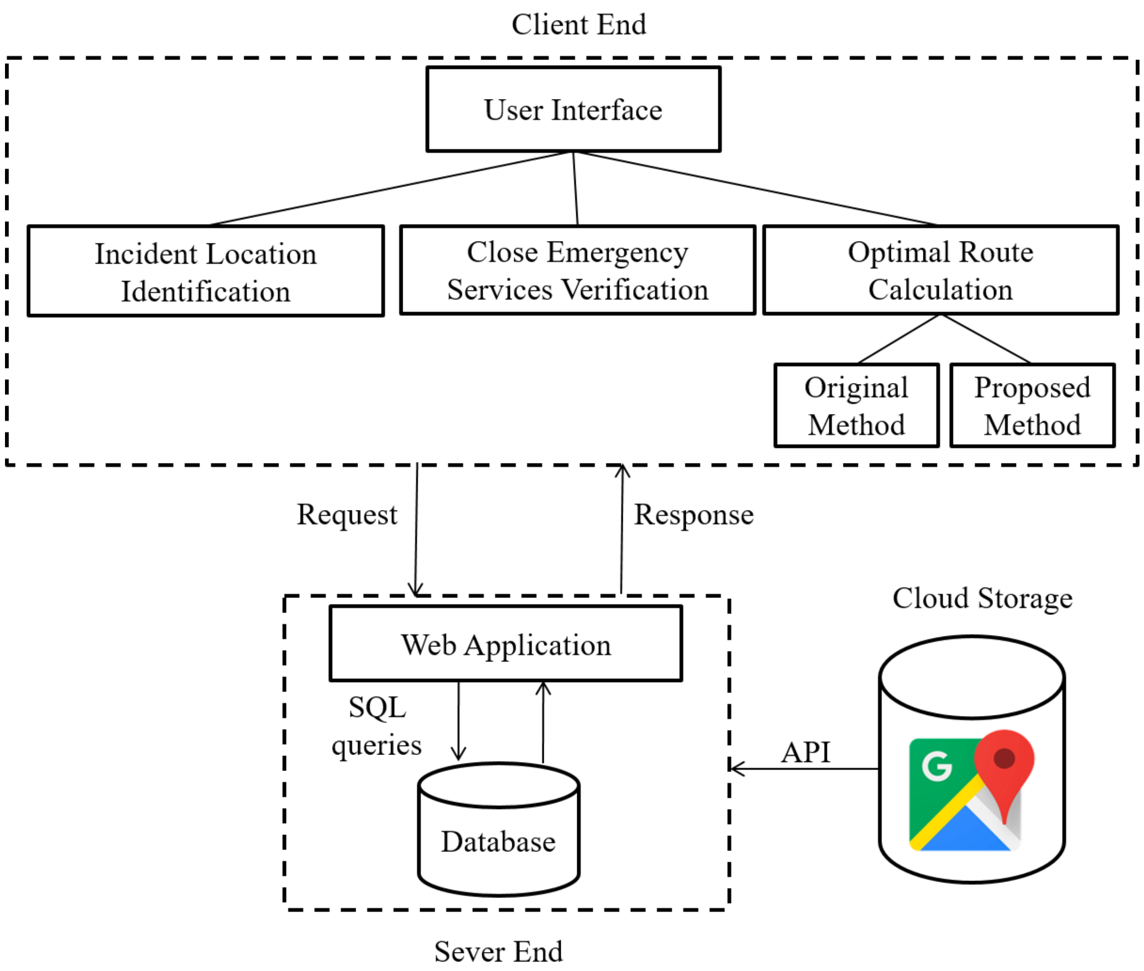


Figure 6.2 Functional Design of Proposed System

The proposed system is organized with three main processes as follows:

i. Incident location identification
ii. Close emergency services verification
iii. Optimal route calculation

The system will accept the township and the residential address or street name as emergency place information to identify the exact location. After receiving the incident address information, the system can verify the location exactly with related latitude and longitude and also import this geo-location on the Google Map to view user. Then the system provides three close emergency services which locate near the incident location and also imports to Google Map to show the user in GUI. After knowing the close emergency services, the user can choose the desired emergency service to find the optimal route and the system will calculate and display the result of the optimal route on Google Map. In the case of optimal route calculation, the system calculated the route by using original Dijkstra's Algorithm and advanced Dijkstra's Algorithm. In the main page of the proposed system is demonstrated with the location of emergency services points such as hospitals, fire stations and police stations in Yangon Region as shown in Fig. 6.3.The emergency event which demonstrates in this implementation is mainly focused on fire.

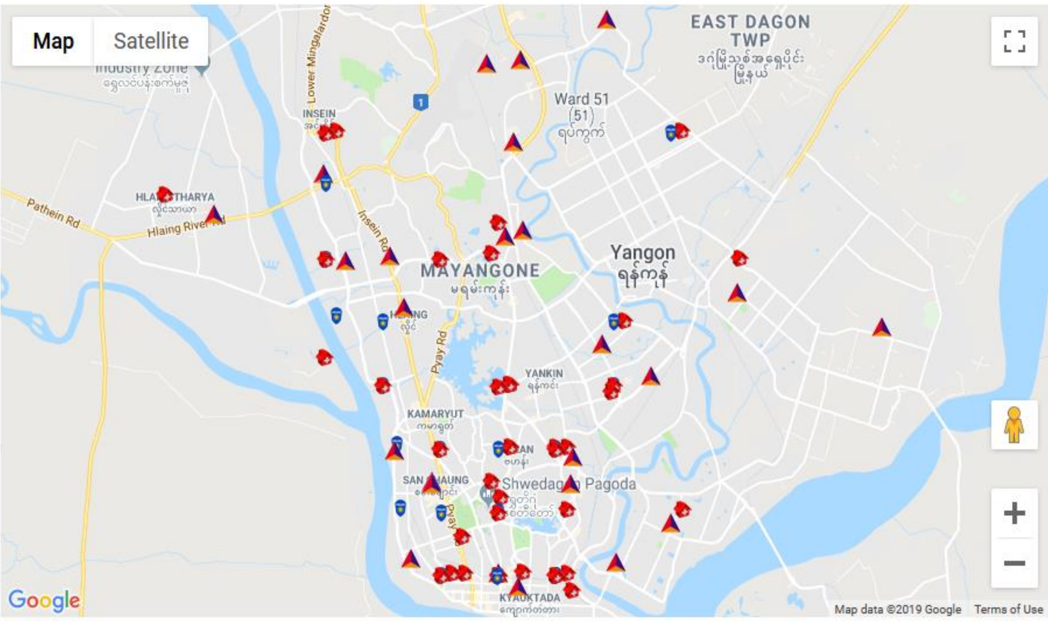


Figure 6.3 Locations of Emergency Service Points in Yangon Region

6.1.1 Incident location identification

In this step, the user can easily select the address of the fire event in drop down box. In this sample testing, fire event location is Kanbawza Lann Thwe, Tarmwe Township and the server replies to the location of the requested address on the Google Map to view user. User can verify the incident location by choosing the township and street name as illustrated in Fig. 6.4. and the location of incident place can be identified as shown in Fig. 6.5.

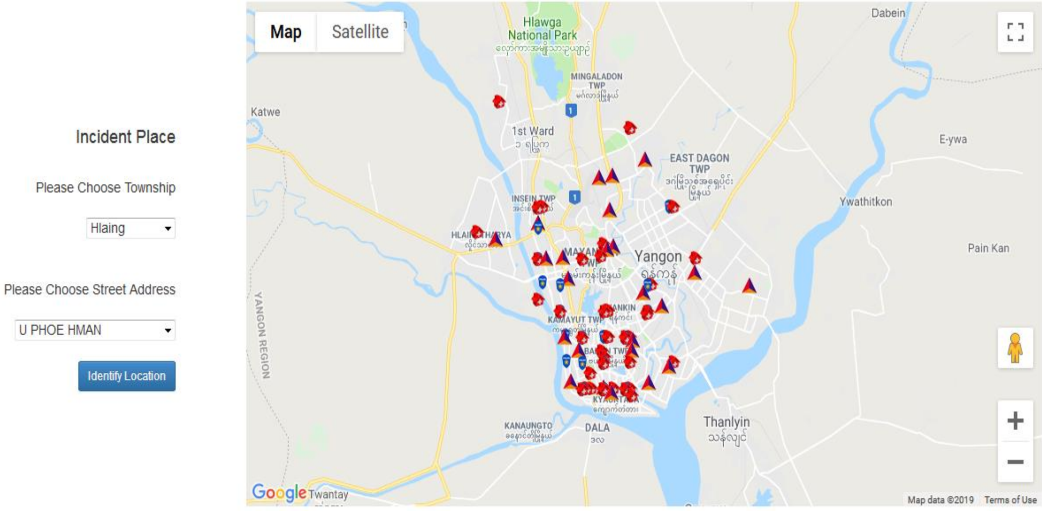


Figure 6.4 Incident Location Verification

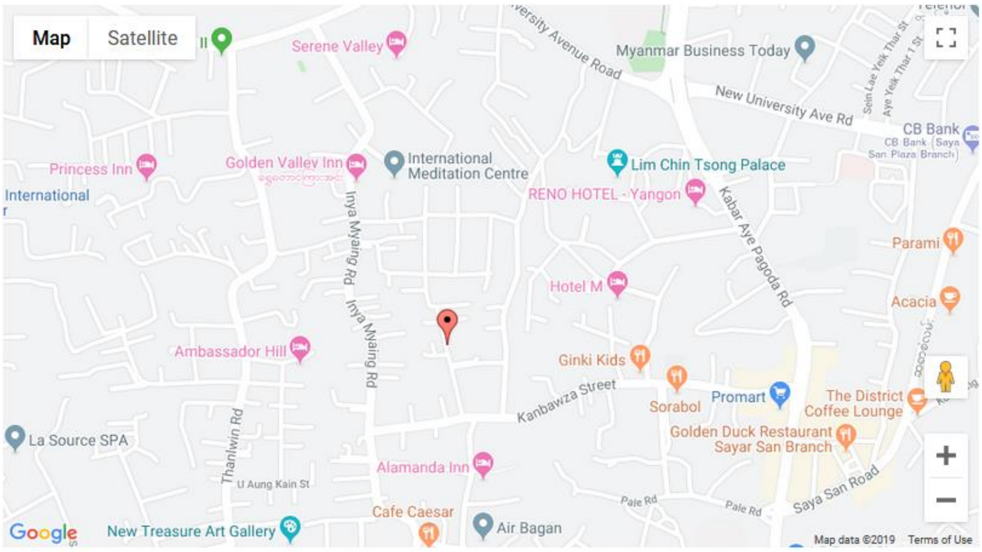


Figure 6.5 Location of Incident Place

6.1.2 Close emergency services verification

After identifying the incident location on Map, the system calculates the three emergency services which locate near the fire emergency case and the result is import to the Google Map to display user as described in Fig. 6.6. The close fire stations of incident location are displayed as ascending order by distance. The closest fire stations for incident location, Kanbawza Lann Thwe, Tarmwe Township are

i. State and Region FSD

ii. San Chaung Fire Station

iii. Tarmway_B Fire Station

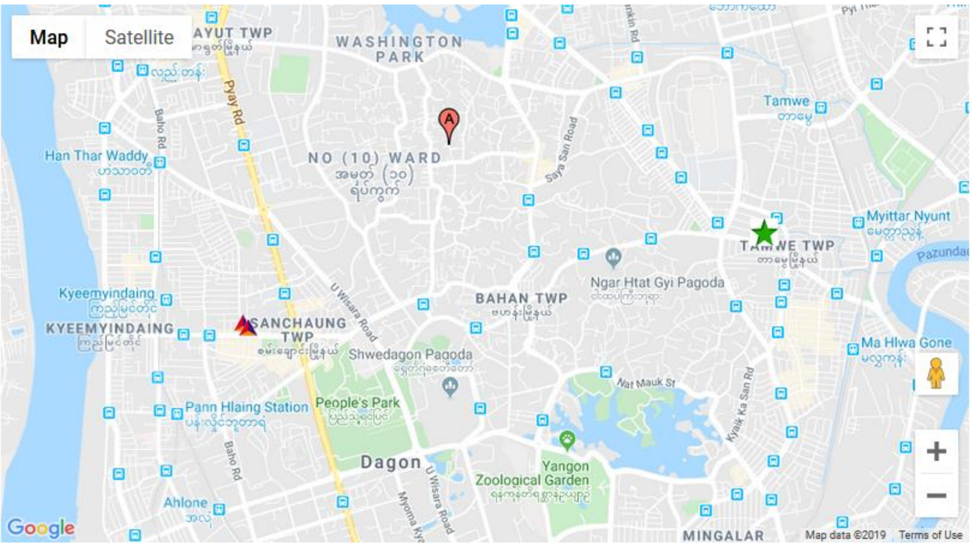


Figure 6.6 The Three Nearest Fire Stations

6.1.3 Optimal route calculation

After calculating the close fire stations, the user can select the desired fire station to know the optimal route to go the fire incident location. In this stage, the route result is calculated by using the conventional method and the proposed routing method. Therefore, there are two route results between each fire station and incident location. In this explanation, the source place is represented for Fire Station and destination place is represented for Fire Event Location with related latitude and longitude. The proposed system generates the route result by using original and proposed methods to go to the incident location and also supports the total distance and time that consume along the way. The result between State and Region FSD and Kanbawza Lann Thwe by using original method is shown in Fig. 6.7.

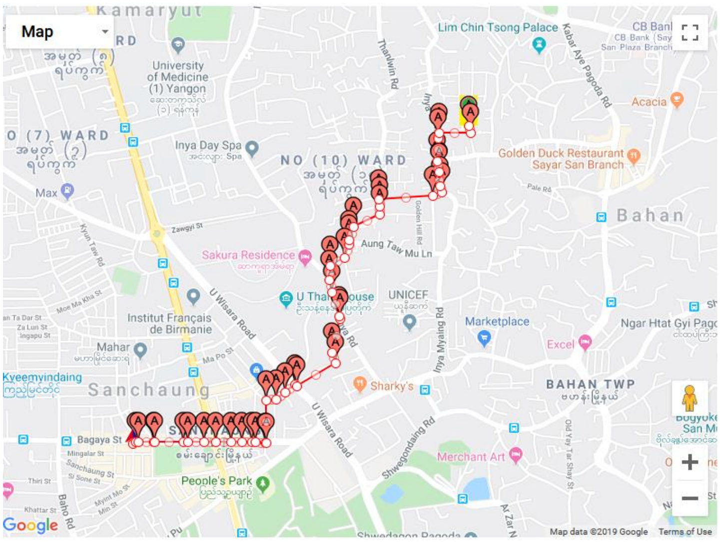


Figure 6.7 Optimal Route between State and Region FSD and Kanbawza Lann Thwe by Using Original Method

FireStation-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA--> BAGAYA>BAGAYA-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->U WISARA RD.-->U WISARA RD-->SHWE LI-->SHWE LI-->INYA-->INYA-->INYA-->INYA-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->GOLDEN HILL-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYAMYAING-->INYAMYAING-->KANBAWZA LAN THWE--> Incident Place

Total Distance To Go Incident Place : 2.98060 km

Time Taken To Go Incident Place : 3.72575 min

In the optimal route results by using original method, there are one-ended streets on the way to go incident place. The emergency vehicles cannot pass through from Than Lwin Street to Golden Hill Street and Inya Myaing Street to Kanbawza Lann Thwe because the buildings are situated between these streets and cannot use to go to the incident location. Fig. 6.8 shows the road condition between Inya Myaing Street to Kanbawza Lann Thwe and Fig. 6.9 shows the road condition of Than Lwin Street to Golden Hill Street. These are the issues of original method in route finding process. Therefore, the proposed method is applied to fix the problem and the proposed method provides the satisfying result. The optimal route result by using proposed method is shown in Fig. 6.10.

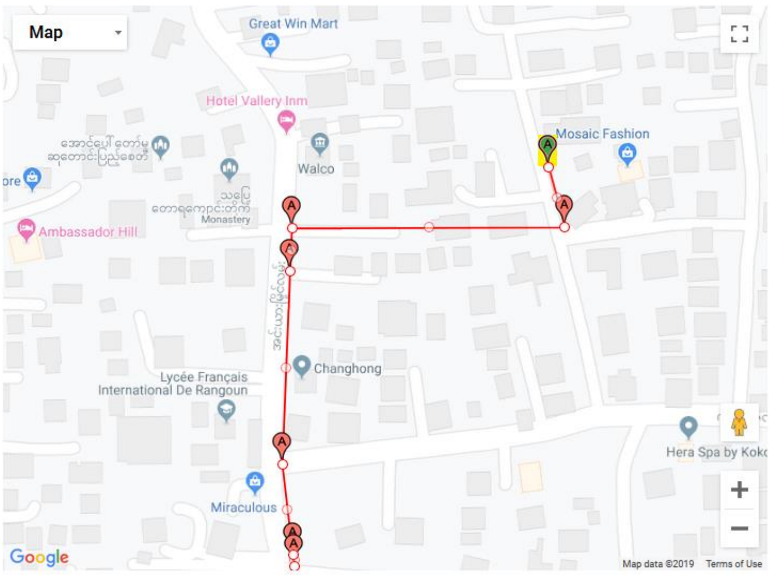


Figure 6.8 Road Condition between Inya Myaing Street and Kanbawza Lann Thwe

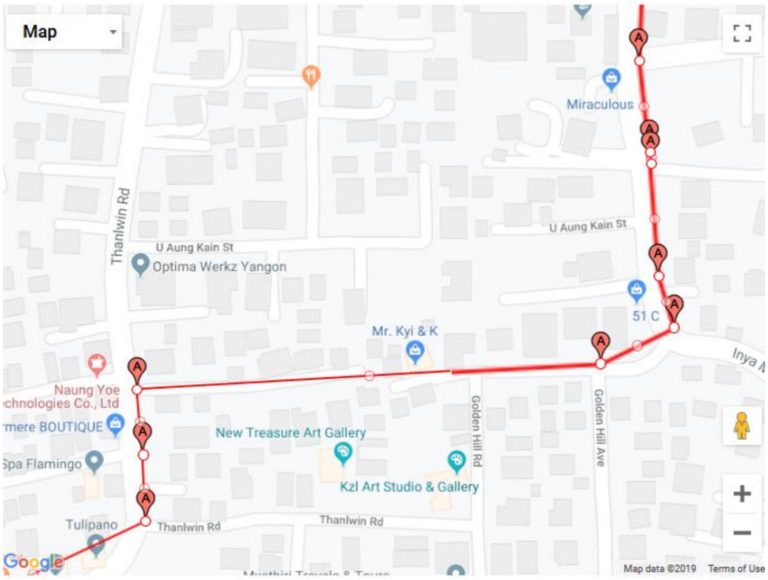


Figure 6.9 Road Condition between Than Lwin Street and Golden Hill Street

The optimal route results between Sanchaung Fire Station and Kanbawza Lann Thwe by using original method is illustrated in Fig 6.11. In the optimal route finding by original method, it also chooses Than Lwin Street to Golden Hill Street and Inya Myaing Street to Kanbawza Lann Thwe to go to the incident location and the proposed method give the result by avoiding the one-ended streets. Fig. 6.12 shows the route result by using proposed method.

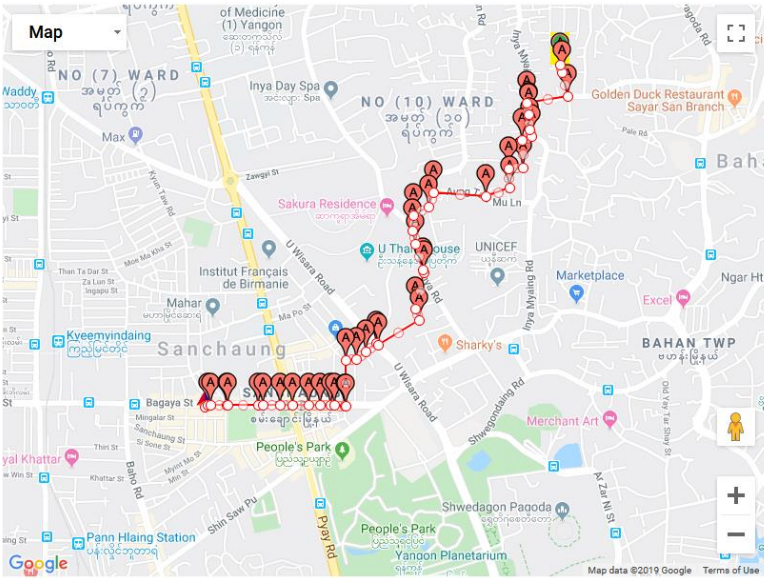


Figure 6.10 Optimal Route between State and Region FSD and Kanbawza Lann Thwe by Using Proposed Method

FireStation-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->U WISARA RD.-->U WISARA RD-->SHWE LI-->SHWE LI-

->INYA-->INYA-->INYA-->INYA-->THAN LWIN-->THAN LWIN-->THAN LWIN-->AUNG TAW MU-->LAN THWE-->LAN THWE-->GOLDEN HILL-->GOLDEN HILL-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->KANBAWZA-->KANBAWZA LAN THWE--> Incident Place

Total Distance To Go Incident Place : 3.05594 km

Time Taken To Go Incident Place : 3.81992 min

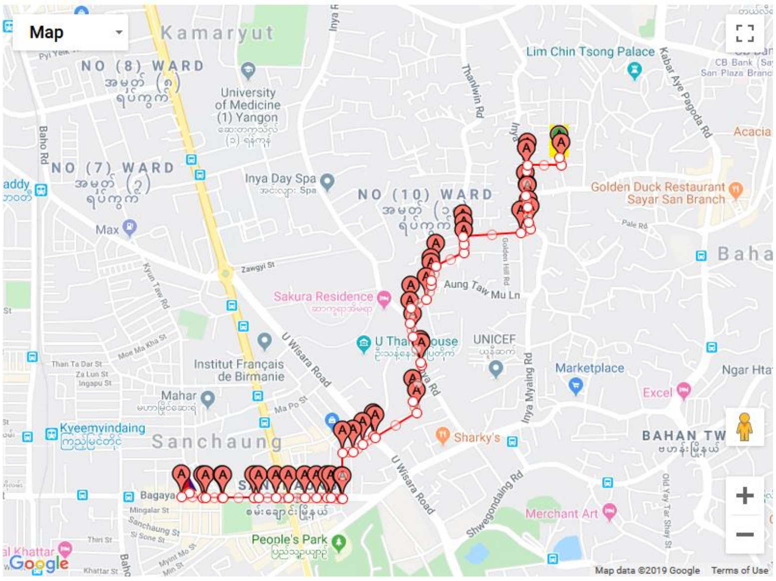


Figure 6.11 Optimal Route Between Sanchaung Fire Station and Kanbawza Lann Thwe by Using Original Method

FireStation-->ZABURIT-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->U WISARA RD.-->U WISARA RD-->SHWE LI-->SHWE LI-->INYA-->INYA-->INYA-->INYA-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->THAN LWIN-->GOLDEN HILL-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYAMYAING-->INYAMYAING-->KANBAWZA LAN THWE--> Incident Place

Total Distance To Go Incident Place : 3.07989 km

Time Taken To Go Incident Place : 3.84987 min

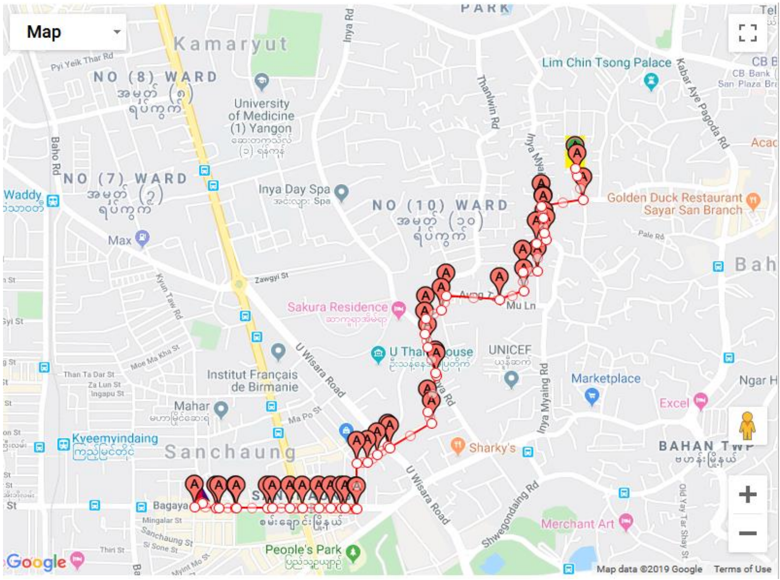


Figure 6.12 Optimal Route Between Sanchaung Fire Station and Kanbawza Lann Thwe by Using Proposed Method

FireStation-->ZABURIT-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->BAGAYA-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->MYAYNI GONE ZAY ST-->U WISARA RD.-->U WISARA RD-->SHWE LI-->SHWE LI-->INYA-->INYA-->INYA-->INYA-->THAN LWIN-->THAN LWIN-->THAN LWIN-->AUNG TAW MU-->LAN THWE-->LAN THWE-->GOLDEN HILL-->GOLDEN HILL-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->INYA MYAING-->KANBAWZA-->KANBAWZA LAN THWE--> Incident Place

Total Distance To Go Incident Place : 3.15523

Time Taken To Go Incident Place : 3.94404

Fig. 6.13 shows the route result between Tarmway_B Fire Station and Kanbawza Lann Thwe by using original method and proposed method. In this case, the two methods provide the same result and there is no one-ended or narrow streets along the way.

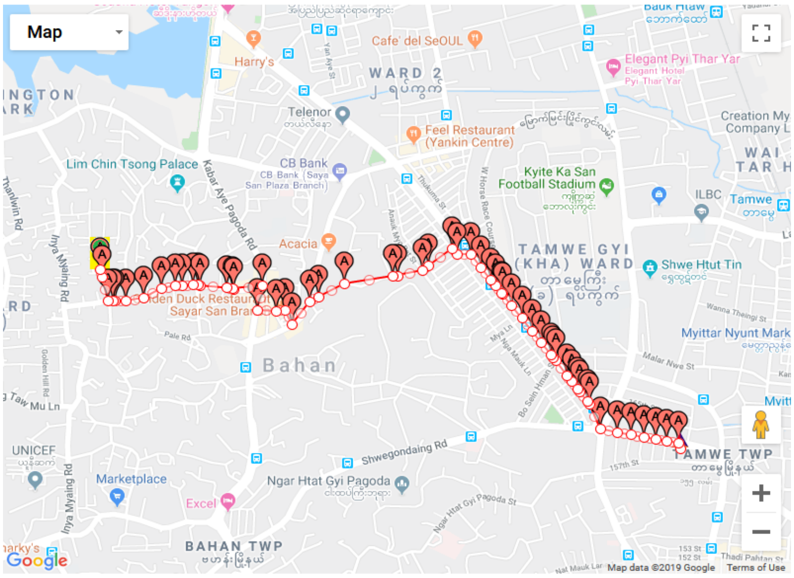


Figure 6.13 Optimal Route Between Tarmway_B Fire Station and Kanbawza Lann Thwe by Using Original Method and Proposed Method

Fire Station-->THAMAIN BAYAN-->THAMAIN BAYAN-->THAMAIN BAYAN-->THAMAIN BAYAN-->THAMAIN BAYAN-->THAMAIN BAYAN-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->NO.1 INDUSTRIAL RD-->U CHIT MAUNG-->U CHIT MAUNG-->U CHIT MAUNG-->U CHIT MAUNG-->U CHIT MAUNG-->U CHIT MAUNG-->LAN THWE-->MIN STREET-->MIN STREET-->ANAUK MYO PAT ST-->KOK KINE SWIMMING POOL-->KOK KINE SWIMMING POOL-->KOK KINE SWIMMING POOL-->KOK KINE SWIMMING POOL-->SAYAR SAN NO.1-->SAYAR SAN NO.1-->LAN THWE-->LAN THWE-->KABARAYE PAGODA-->KABARAYE PAGODA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA-->KANBAWZA LAN THWE--> Incident Place

Total Distance To Go Incident Place: 3.56210 km

Time Taken To Go Incident Place: 4.45263 min

According to the route results mentioned above, we can compare the street name and the condition of street which passed through along the way and can also compare the total distance to go to the incident location and time consumption to go to there. The main purpose of the proposed method is to avoid the streets which are not wide enough to enter fire trucks, one-ended or narrow. Therefore, it calculates the route for emergency vehicles by avoiding these conditions. That is why the proposed work can support the optimal route result. According to the real data of road conditions between Than Lwin Street to Golden Hill Street and Inya Myaing Street to Kanbawza Lan Thwe are one-ended. If the driver enters this street it will consume time to get to the right street. It can cause delay on the way and in the rescue processes for emergency case. Therefore, the result of the proposed method can support the drivers to go to the incident location without delaying caused by wrong street choice along the way.

6.2 Result Analysis

The results of the proposed system are verified and validated by comparing the result of original Dijkstra's Algorithm. The proposed work is tested on the road network with the number of edges 96780 and the number of nodes 32885.

6.2.1 Discussion of Route Result

In order to prove that the proposed method can efficiently provide the optimal route, the original Dijkstra's Algorithm is used to compare. The route result with total distance and time by applying two methods are described with three sample tests.

Test 1: Route Result between Mayangon Fire Station and OK Kyaung Ln, Hlaing Township

Result by using original method:

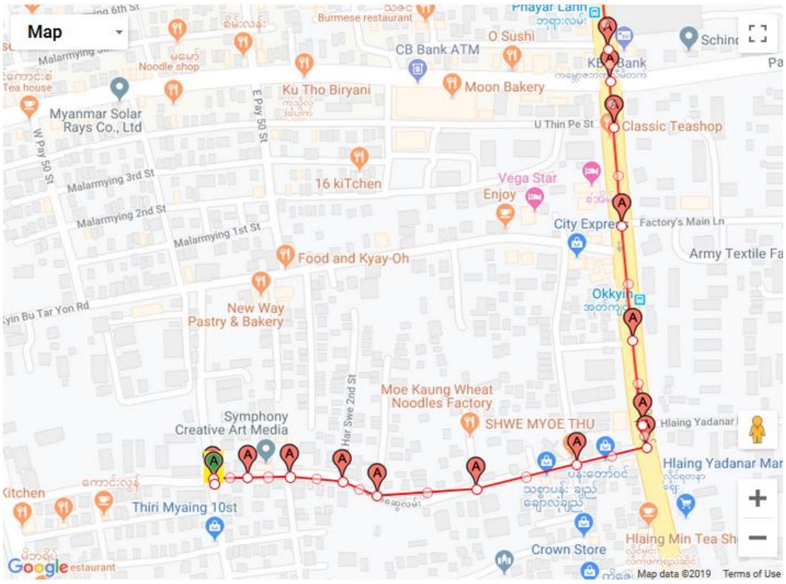


Figure 6.14 Optimal Route between Mayangon Fire Station and OK Kyaung Ln by Using Original Method

Fire Station-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN-->MAHAR SWE-->MAHAR SWE-->MAHAR SWE-->MAHAR SWE-->MAHAR SWE-->MAHAR SWE-->OK Kyaung Ln-->OK Kyaung Ln--> Incident Place

Total Distance: 1.81066 km and Total Time: 2.26333 min

Route result by proposed method:

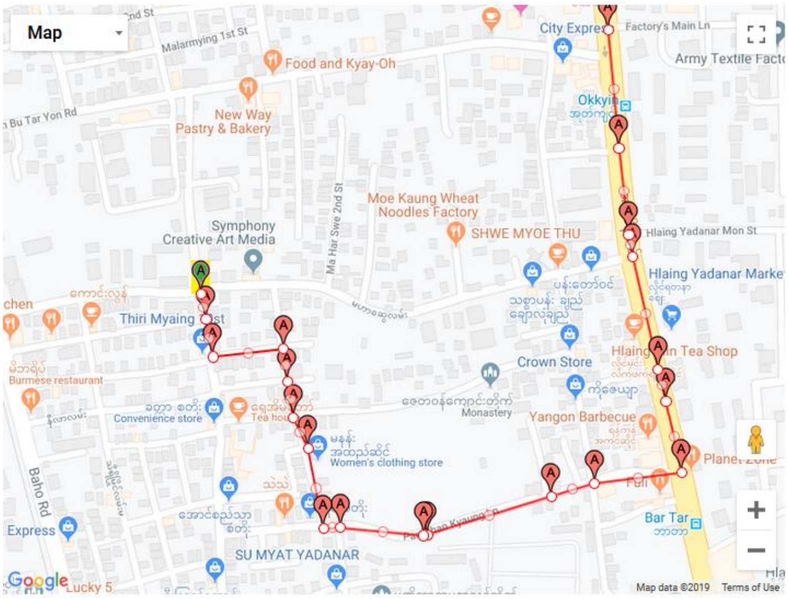


Figure 6.15 Optimal Route between Mayangon Fire Station and OK Kyaung Ln by Using Proposed Method

Fire Station-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->YANGON-INSEIN--> YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN--> YANGON -INSEIN--> YANGON-INSEIN-->YANGON-INSEIN-->YANGON-INSEIN-->PANCHANKYAUNG-->PANCHANKYAUNG-->PANCHANKYAUNG-->PANCHANKYAUNG-->PANCHANKYAUNG-->PANCHANKYAUNG-->PANCHANKYAUNG--> ZAY TA WON KYAUNG-->NO 9 THIRI MYAING-->NO 10 THIRI MYAING-->OK Kyaung Ln-->OK Kyaung Ln-->OK Kyaung Ln--> Incident Place

Total Distance: 2.35087 km and Total Time: 2.93859 min

In sample test 1, Mahar Swe Street which the fire truck cannot enter, but the original method uses this street and the proposed method can avoid the street as described in the route result. The proposed method takes slightly more distance and time of the route than the original one but the route result is optimal and can reduce delay caused by road condition.

Test 2: Route Result between Kyauktada Fire Station and Set Yone Street, Mingalar Taung Nyunt Township

Route result by original method:

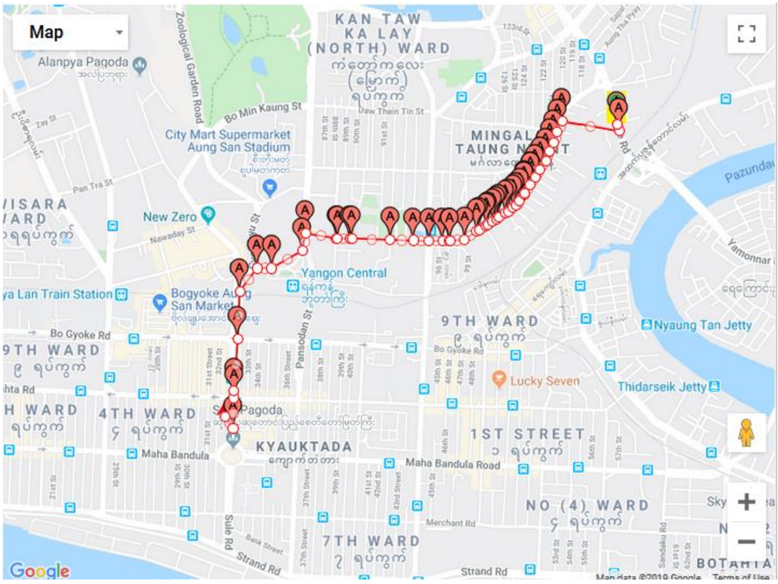


Figure 6.16 Optimal Route between Kyauktada Fire Station and Set Yone Street by Using Original Method

Fire Station-->SULE PAGODA-->SULE PAGODA-->SULE PAGODA-->SULE PAGODA-->SULE PAGODA-->KUN CHAN St-->KUN CHAN St-->KUN CHAN St-->KUN CHAN St-->BO MIN YAUNG-->SET YONE-->SET YONE--> Incident Place

Total Distance To Go Incident Place : 2.93755 km

Time Taken To Go Incident Place : 3.67194 min

Route result by proposed method:

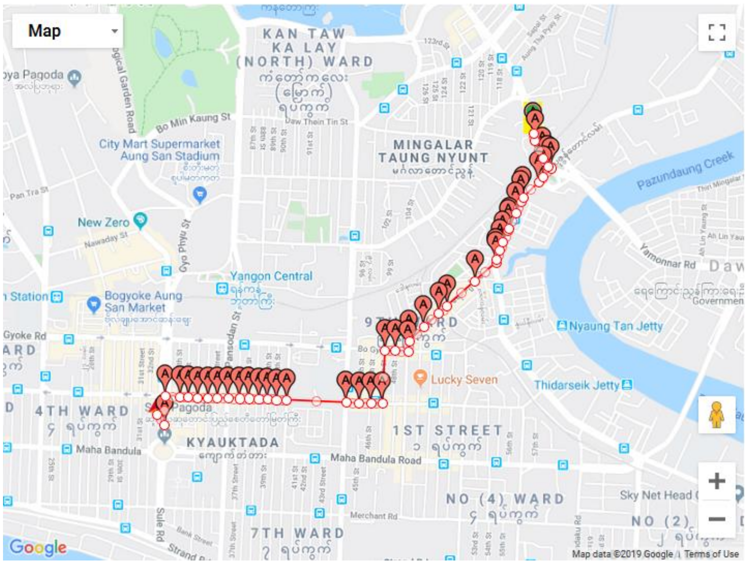


Figure 6.17 Optimal Route between Kyauktada Fire Station and Set Yone Street by Using Proposed Method

Fire Station-->SULE PAGODA-->SULE PAGODA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->ANAWRAHTA-->BOGYOKE AUNG SAN-->BOGYOKE AUNG SAN-->BOGYOKE AUNG SAN-->YAE KYAW-->YAE KYAW-->YAE KYAW-->YAE KYAW-->YAE KYAW-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->UPPER PAZUNDAUNG-->SET YONE-->SET YONE-->SET YONE-->SET YONE-->SET YONE--> Incident Place

Total Distance To Go Incident Place : 2.96819 km

Time Taken To Go Incident Place : 3.71024 min

In this test, the emergency vehicles cannot pass through from Bo Min Yaung Street to Set Yone Street. However, the original method chooses the one-ended street from Bo Min Yaung Street to Set Yone Street to go to the incident location, but the proposed method can select the right street to get the desired location by avoiding the one-ended street.

Test 3: Route result between Mayangone Fire Station and Zayya Thu Kha Street, Mayangone Township

Route result by original method:

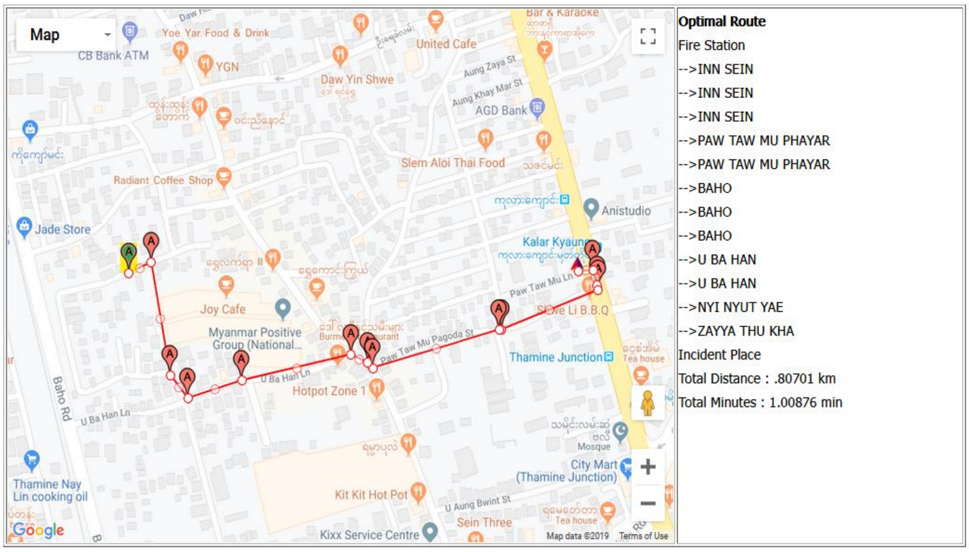


Figure 6.18 Optimal Route between Mayangone Fire Station and Zayya Thu Kha Street by Using Original Method

Fire Station-->INN SEIN-->INN SEIN-->INN SEIN-->PAW TAW MU PHAYAR-->PAW TAW MU PHAYAR-->BAHO-->BAHO-->BAHO-->U BA HAN-->U BA HAN-->NYI NYUT YAE-->ZAYYA THU KHA--> Incident Place

Total Distance To Go Incident Place : 0.80701 km

Time Taken To Go Incident Place : 1.00876 min

Route result by proposed method:

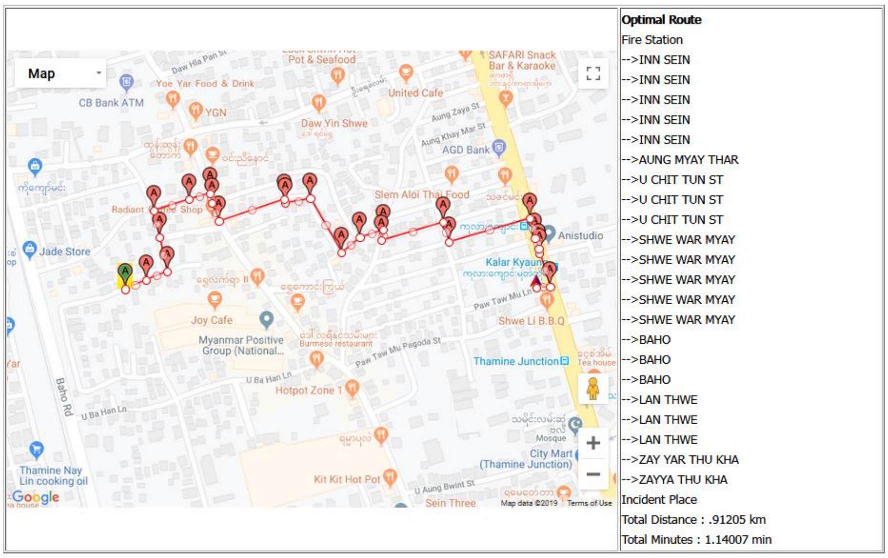


Figure 6.19 Optimal Route between Mayangone Fire Station and Zayya Thu Kha Street by Using Proposed Method

Fire Station-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->INN SEIN-->AUNG MYAY THAR -->U CHIT TUN ST-->U CHIT TUN ST-->U CHIT TUN ST-->SHWE WAR MYAY-->SHWE WAR MYAY-->SHWE WAR MYAY-->SHWE WAR MYAY-->SHWE

WAR MYAY-->BAHO-->BAHO-->BAHO-->LAN THWE-->LAN THWE-->LAN THWE-->ZAY YAR THU KHA-->ZAYYA THU KHA--> Incident Place

Total Distance To Go Incident Place : 0.91205 km

Time Taken To Go Incident Place : 1.14007 min

In sample test 3, the fire truck cannot enter from Nyi Nyut Yae road to Zayya Thu Kha Street because Nyi Nyut Yae Street is one-ended. The original method give the route result passed through Nyi Nyut Yae Street to Zayya Thu Kha Street and it is the wrong way to go the incident place. The proposed method can provide the effective optimal route for emergency vehicles by avoiding the one-ended street with slightly different in distance and time.

6.2.2 Performance Evaluation of Two Methods in Number of Iterations

Fig. 6.20 shows the comparison for the number of iterations by using two methods. By applying two methods, this evaluation provides the different route result within 100-time experiments. In most experiments, the number of iterations by using proposed method is smaller than the original method and in some cases, the number of iteration by using proposed method is larger than the original method, but it gives optimal route result. Some example of iteration distinction between two methods is explained in section 4.3.2. with sample tests.

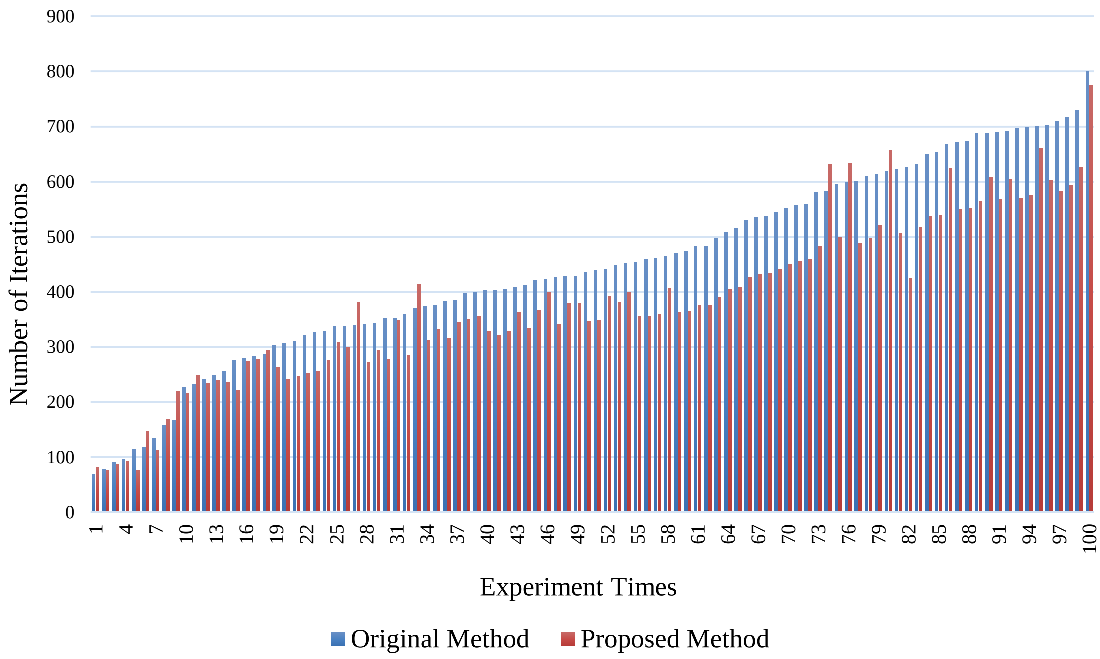


Figure 6.20 Number of Iteration Comparison

6.2.3 Performance Evaluation of Two Methods in Processing Time

The comparison of the performance two methods with the number of visited nodes and processing time is shown in Fig. 6.21. As the following result graph, the processing time of the original Dijkstra's Algorithm is significantly increased when the numbers of nodes are huge.

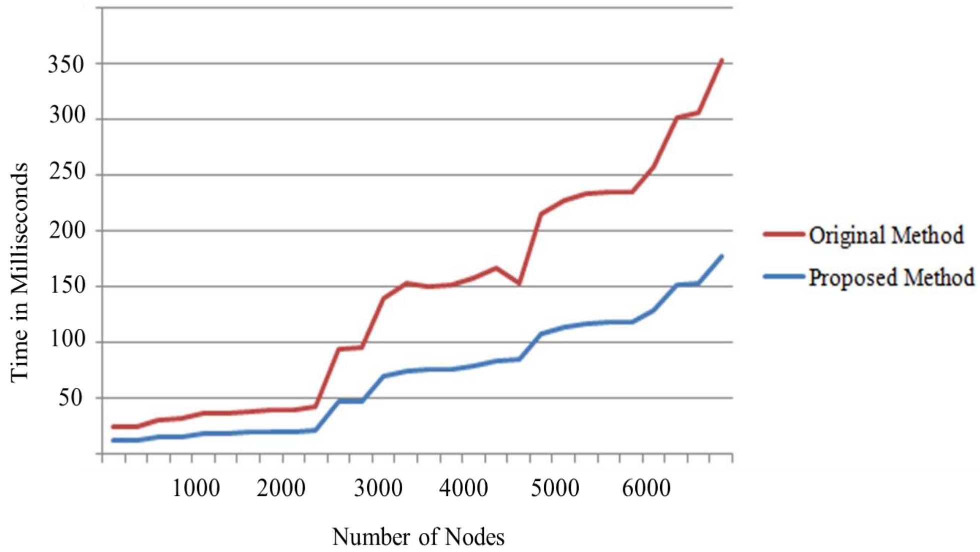


Figure 6.21 Execution Time Comparisons of Two Methods

6.2.4 Evaluation of Runtime Complexity

Table 5.3 shows the required computation time for number of nodes 10,100,1000,10000, and 10000 in each operation that process in two methods.

Table 6.1 Evaluation of Runtime Complexity Based on Number of Nodes

Run Time	Computation for 10 nodes	Computation for 100 nodes	Computation for 1000 nodes	Computation for 10000 nodes	Computation for 100000 nodes
O (1)	1	1	1	1	1
O (n)	10	100	1000	10000	100000
O (log n)	3	7	10	13	17

6.3 Chapter Summary

This chapter mainly discussed the web application structure, functional design, and detailed processing steps and tested results of the proposed system. This chapter also briefly discussed and described the nature and difference between the original method and the proposed method in the route-finding step with detailed result explanations. In addition, the performance of the system is implemented and tested by comparing the original route-finding method and proposed one. In the performance evaluation of proposed system, the execution time of two methods, the experiment 100 times are analyzed and tested and to compare the number of iterations which process by two methods and the runtime complexity are compared with detailed results and explanations. The outcomes of the proposed work can be helpful for emergency vehicles which always use the unstructured or complex road network in an emergency situation.

CHAPTER 7

CONCLUSION AND FUTURE WORK

The developing rate of country could be measured by minimizing loss of lives and damage of property. Fire incident as a hazard ought to be controlled at all cost. Optimal route identification system for emergency vehicles (e.g. fire trucks) plays the significant role in solving the route finding problem on the Yangon Region road network when the fire incident happened. The combination of advanced technologies and GIS technology has been verified to be a tool for improving ability in fire management services. The proposed system is implemented as a web-based application to solve the problems faced by the fire trucks and emergency vehicles. This system significantly solves the problem of complex road network, and also analyzes the interruption caused by road condition such as one-ended and narrow streets. It provides the useful decision to determine the closest fire stations and the optimal route between fire stations and incident place in terms of distance and time. The proposed system can handle multi accident situation. The proposed system is developed with simple and easy to understand for the user and will help the emergency management transportation by providing the optimal path which will decrease the damage level, valuable lives, and properties by effective and efficient evacuation processes. Moreover, in this proposed work, different improved methods of the route-finding system have also been analyzed and one of the route finding methods has been proposed to improve the emergency service transportation. The proposed route-finding method is able to calculate the optimal route efficiently in the poorly structured and large road network. The proposed method is more superior to the basic route finding method, as it requires some new algorithmic ingredients such as road status variables. The optimal route results of proposed algorithm achieve more accurate and speed up than the original method. And also the results of the proposed work show that technologies are very efficient for emergency transportation which use the unstructured road network. This study has mainly focused on the use of the limited statistical data that was difficult to acquire from the emergency service departments. According to the route result of the proposed system, it is conducive to the rescue workers and the drivers, and emergency teams much easier, and makes sure to get to the accident site without any delay.

7.1 Advantages of the Proposed System

The advantages of the proposed system are described as follows:

i. This system is simple and easy to understand for users.
ii. It helps the emergency car drivers who are unfamiliar with the unstructured road network during an emergency case.
iii. It supports the optimal route result for emergency services without delay along the way to save lives and properties, and also reduces the damage level of emergency cases.
iv. It can provide three or more emergency services which locate near the emergency location.
v. The system has experienced on the large road network and obtained satisfying results.
vi. The proposed method reduces the number of iteration that need to traverse the route thus reducing the time complexity than the original method.
vii. The proposed path finding algorithm reduces number of iteration required to traverse the path, thus reducing time complexity
viii. The proposed method is terminated when the target node is found. Therefore, the memory consumption is also less than the original method.
ix. Different kinds of emergency applications can be applied by using the proposed approach.
x. The proposed system can be implemented in all other cities of the road network in Myanmar if the status of streets and details of the geospatial database of the applied city are available.

7.2 Limitations of the Proposed System

i. The proposed system depends on internet connectivity and telephone network.
ii. It is needed to add or remove the status of road condition as block or unblock because these conditions lead to an uncertain route result.
iii. It does not use sensors and GPS devices between drivers and emergency service organization and does not consider the real-time condition of traffic congestion.
iv. It is majorly focused on the use of the limited statistical data which is difficult to obtain from the emergency service organizations.

7.3 Future Works

An emergency service organization can take the maximum advantage by advancing the proposed web application that improves with other needed factors in the further. Many advanced wireless technologies like that sensors, GPS, and mobile GIS application can better communication between the drivers and emergency service organizations. The future work is focused on integrating with real time on-road traffic condition to provide more dynamic, reliable and accurate route result.

AUTHOR’S PUBLICATIONS

1. K-zin Phyo and Myint Myint Sein, “Optimal Route Finding to Support Fire Emergency Service," 14th International Conference on Computer Applications, Yangon, Myanmar, 25th - 26th February, 2016,pp.58-62.

2. K-zin Phyo and Myint Myint Sein, “Optimal Path Finding for Emergency Cases on Android," 14th Annual International Conference on Mobile Systems, Applications and Services, Singapore, 23rd - 25th June, 2016, pp.71.

3. K-zin Phyo and Myint Myint Sein, "Optimal Route Finding for Weak Infrastructure Road Network," 10th International Conference on Genetic and Evolutionary Computing , Fuzhou City, Fujian Province, China, November 7-9, 2016, pp. 230-237.

4. K-zin Phyo and Myint Myint Sein,” Effective Emergency Response System by Using Improved Dijkstra’s Algorithm”, 15th International Conference on Computer Applications , Yangon, Myanmar, February 16-17, 2017, pp.257-261.

5. K-zin Phyo and Myint Myint Sein,” ,” Effective Evacuation Route System During Natural Disaster ”, Asia Pacific Advanced Network Research Workshop, Dalian, China, August 26 – September 1, 2017, pp.70-75.

6. K-zin Phyo and Myint Myint Sein,” ,” Effective Evacuation Route System During Natural Disaster ”, Open Journal System, Volume 44.

7. K-zin Phyo and Myint Myint Sein, “Optimal Route Assessment for Emergency Vehicles Travelling on Complex Road Network”, 11th Multi-disciplinary International Workshop on Artificial Intelligence, LNCS and LNAI 10607, Brunei, November 18 - 19,2017,pp. 380-390.

8. K-zin Phyo and Myint Myint Sein, “Optimal Route Assessment for Emergency Vehicles Travelling on Complex Road Network”, Lecture Note In Computer Science, Volume 10607.

9. K-zin Phyo and Myint Myint Sein ,”Investigation of Optimum Rescue Itinerary by Using Advanced Routing Method”, IEEE 7th Global Conference on Consumer Electronics, Nara, Japan, October 4-13,2018,pp.521-522.

BIBLIOGRAPHY

[1] M. Alivand, A.A. Alesheikh and M.R. Malek,” New Method for Finding Optimal Path in Dynamic Networks”, World Applied Sciences Journal 3 (Supple 1): 25-33, 2008.

[2] M.O.Almumaiz,” Improvement of Transportation Network of Al-Muwaffaqiyah Town Using Gis”, Journal of Engineering and Development, Volume 18, Issue 5, September, 2014, pp. 231-245.

[3] E.O.Amoako,” Shortest Route Optimization for Emergency Service: Case Study in Kumasi,Ghana”, International Journal of Innovative Research and Development, Volume 6 ,Issue 9,pp 147-166.

[4] C. Bagchi, K. Chopra and M. Yamuna,” Ambulance Service Using Modified Dijkstra’s Algorithm”, International Journal of Pharmacy & Technology, Sep-2016 ,Vol. 8 , Issue No.3 ,pp. 17627-17633.

[5] M.Basyir, M.Nasir, Suryati and W. Mellyssa,” Determination of Nearest Emergency Service Office using Haversine Formula Based on Android Platform”, EMITTER International Journal of Engineering Technology ,Volume 5, Issue 2, December 2017,pp. 270-278.

[6] A.Belan, R.Mudliar, S. Muley, C. Darade and R.A. Kudale, ” Location Based Emergency Services”, International Journal of Engineering Research and Technology (IJERT), Volume 3, Issue 2, February ,2014,pp. 2516-2520.

[7] N. A. Bipu,” Geographic Information System and Spatial Analysis”, International Journal for Empirical Education and Research.

[8] A.Chandak , R. Bodhale , R.Burad,” Optimal shortest path using HAS, A star and Dijkstra algorithm”,Imperial Journal of Interdisciplinary Research (IJIR) ,Volume 2, Issue 4, 2016,pp. 978-980.

[9] T. Chondrogiannis, P. Bouros, J. Gamper and U.Leser,” Exact and Approximate Algorithms for Finding k-Shortest Paths with Limited Overlap”, 20th International Conference on Extending Database Technology (EDBT), March 21-24, 2017,Venice, Italy,pp. 414-425.

[10] D. Schultes ,“Route Planning in Road Networks”, February 2008.

[11] A. R. Dasgupta, Distinguished Professor BISAG and Honorary Advisor, GIS Development,”Introduction to Enterprise GIS”, June 2010.

[12] P. Dhanvani , “Difference between BFS and DFS | BFS vs. DFS”, April 18, 2014.

[13] P.Dong, D. Li, J. Xing, H. Duanand and Y. Wu," A Method of Bus Network Optimization Based on Complex Network and Beidou Vehicle Location", Journal of future internet, April 2019.

[14] P.V.Dooren," Graph Theory and Applications", Inspired from the course notes of V. Blondel and L. Wolsey (UCL), Dublin, August 2009.

[15] F. Escobar, G. Hunter, I. Bishop and A. Zerger ," Introduction to GIS", Department of Geomatics, The University of Melbourne.

[16] A.Fitro, O. S. Bachri*, A. I. S. Purnomo, I. Frendianata,"Shortest Path Finding in Geographical Information System Using Node Combination And Dijkstra's Algorithm", International Journal of Mechanical Engineering and Technology (IJMET), Volume 9, Issue 2, February 2018, pp. 755–760.

[17] A. V. Goldberg, R. E. Tarjan," Expected Performance of Dijkstra's Shortest Path Algorithm", Princeton University, June 1996.

[18] A.Goyal, P.Mogha, R.Luthra and N.Sangwan," Path Finding: A* or Dijkstra'?", International Journal in IT and Engineering, Volume 02, Issue 1, January, 2014.

[19] N.Gupta, K.Mangla,A. K. Jha and M.Umar," Applying Dijkstra's Algorithm in Routing Process", International Journal of New Technology and Research (IJNTR), Volume 2, Issue 5, May, 2016, pp. 122-124.

[20] S. Hartanto, A. P. U. Siahaan, W.Fitriani," Haversine Method in Looking for the Nearest Masjid", International Journal of Recent Trends in Engineering and Research (IJRTER) Volume 3, Issue 8, August 2017, pp. 187-195.

[21] F. Ivis, "Calculating Geographic Distance: Concepts and Methods", NESUG 2006.

[22] G.R.Jagadeesh and T.Srikanthan," Location Based Emergency Services Networks: A Hierarchical Approach", School of Computer Engineering Nanyang Technological University, Singapore.

[23] A.Jain, U.Datta and N.Johsi, "Implemented Modification in Dijkstra's Algorithm to Find the Shortest Path for 'N' Nodes with Constraint", International Journal of Scientific Engineering and Applied Science, Volume-1, Issue 2, February 2016, pp. 421-426.

[24] S. Jaiswal, "Representation of Graphs"

[25] N.Kai, Z.Y.Ting and M. Y.Peng ." Shortest Path Anal Modified Dijkstra's Shortest Path (MSDP) algorithm ysis Based on Dijkstra's Algorithm in Emergency Response System", Vol.12, No.5, May 2014, pp. 3476 -3482.

[26] A. Karduni , A. Kermanshah and S. Derrible, “A protocol to convert spatial polyline data to network formats and applications to world urban road networks ”, Scientific Data, 2016, Volume 3, Article number 160046 .

[27] F. Khamayseh, N.Arman , “Improvement of Shortest Path Algorithm Using Subgraphs’ Heuristics”, Journal of Theoretical and Applied Information Technology, June 2015, Volume 76, Issue 1, pp. 109-117.

[28] S. Kukadapwar , D. Parbat,” Estimation of Optimal Path on Urbab Road Networks Using AHP Algorithm”, International Journal for Traffic and Transport Engineering, 2016, volume 6 Issue 1,pp. 13–24.

[29] N. Kumar, M. Kumar and S. K. Srivastva,” Geospatial Path optimization for Hospital: a case study of Allahabad city, Uttar Pradesh”, International Journal of Modern Engineering Research, Volume 4, Issue10, October 2014, pp. 9-14.

[30] N.Kumar, M.Kumar, D.M.Denis , S. K.Srivastava and O.S.Srivastva,” Geospatial School Bus Routing”, International Journal Of Engineering And Science (IJES), 2014, Volume3, Issue 11 , pp. 80-84.

[31] D.Lippman,”Graph Theory and Network Flows” (http://www.flickr.com/photos/sbeebe/2850476641/).

[32] B. Lu, C. Brunsdon, M. Charlton and P. Harris,” The Minkowski approach for choosing the distance metric in geographically weighted regression”, International Journal of Geographical Information Science, September 2015.

[33] A. Madkour, W.G. Aref, Faizan Ur Rehman, Mohamed Abdur Rahman, Saleh Basalamah, A Survey of Shortest-Path Algorithms, May 8, 2017.

[34] P.V. Matheus, E.Strano , P. Bordin and M. Barthelemy,” The simplicity of planar networks”, Scientific Reports.

[35] C. Monga and Richa, “Graph Traversals and Its Application In Graph Theory” International Journal of Computer Science and Mobile Applications, Vol.6 Issue. 1, January- 2018, pg. 38-42.

[36] D.Munoz and R. Enriquez, “Heuristic Approach to Position Location Problem”, Position Location Technique and Application, 2009.

[37] M. Muthulakshmi and M.M. Shanmugapriya, “Shortest Path Algorithm and Its Implementation”, International Journal of Mathematics Trends and Technology (IJMTT), Volume 36, Issue 2, August 2016, pp. 82-85.

[38] V.T. N. Nha, S. Djahel and J. M. Lero ,"A Comparative Study of Vehicles' Routing Algorithms for Route Planning in Smart Cities", UCD School of Computer Science and Informatics, Ireland.

[39] D. E. Nyaung and K. Yamaguchi, "Smartphone based Emergency Reporting and Response System in Myanmar", Rikkyo University Repositry, Feburary 2018.

[40] D.D.Prasad,A. S. Muttipati, M. Snehadivya and S. Kavitha "Applications of Computer Science Based on Graph theory", International Journal of Engineering, Science and Mathematics , Volume 6, Issue 8, December 2017, pp 1117-1222.

[41] K. Pritee, R.D. Garg," Identification of Optimum Shortest Path using Multipath Dijkstra's Algorithm Approach", International Journal of Advanced Remote Sensing and GIS 2017, Volume 6, Issue 1, pp. 2442-2448.

[42] R.Puente, L.Cortes," Algorithm for Shortest Path search in Geographic Information Systems by Using Reduced Graphs", Springer Plus, 2013, Volume 2, Issue 1.

[43] M.Puthuparampil, "Report Dijkstra's Algoritm".

[44] M. Rouse, "Definition of Google Map", February, 2013.

[45] .A.M.Sabri, A. S. H. Basari, B.Husin and K. A. F.A. Samah," Simulation Method of Shortest and Safest Path Algorithm for Evacuation in High Rise Building", Applied Mathematical Sciences, Vol. 8, 2014, no. 104, pp. 5163 – 5172.

[46] M.N.Shah," Implementation of Graph Theory in Computer Networking To Find an Efficient Routing Algorithm", International Journal of Innovative Research in Computer and Communication Engineering, Volume 4, Issue 1, January, 2016,pp. 12-20.

[47] R.Shahid , S.Bertazzon , M.L. Knudtson and W.A.Ghali , "Comparison of distance measures in spatial analytical modeling for health service planning", BMC Health Services Research, November 6,2009.

[48] M.Sharma, J.K.Gupta and A.Lala," Survey of Route Choice Modelsin Transportation Networks", Advances in Intelligent Systems and Computing, pp. 1285-1289.

[49] L. Shen, H. Shao, L.Zhang and J. Zhao," The Global Optimal Algorithm of Reliable Path Finding Problem Based on Backtracking Method", Mathematical Problems in Engineering ,Volume 2017, Article ID 4586471.

[50] S.G.Shirinivas, S.Vetrivel and N.M.Elango," Applications of Graph Theory in Computer Science Overview", International Journal of Engineering Science and Technology, Vol. 2, Issue 9, 2010, pp. 4610-4621.

[51] A.K.Singh and P.Singh "An approach for web based GIS Route finder system", International Journal of Advanced Research in Computer Science and Software Engineering, Volume 2, Issue 5, May 2012.

[52] R.Singhal and K.Malhotra, ," Graph and its representations", Geeksforgeeks.

[53] S. Sivakumar and C.Chandrasekar," Modified Dijkstra's Shortest Path Algorithm", International Journal of Innovative Research in Computer and Communication Engineering, Volume 2, Issue 11, November 2014, pp. 6459-6456.

[54] S. Skiena,"Lecture 10: Graph Data Structures", Department of Computer Science State University of New York, Stony Brook, NY 11794–440.

[55] Q.Song,M.Li and X.Li," Accurate and fast path computation on large urban road networks", PLoSONE13(2), University of Jinan,February,2018,pp. 1-13.

[56] S.Surbhi ," Difference Between Latitude and Longitude", February 15, 2017.

[57] R. J.Venkat, "Path Finding - Dijkstra's Algorithm", Indiana State University, December 13, 2014.

[58] L. Wang, X. Xue, Z. Zhao and Z. Wang, "The Impacts of Transportation infrastructure on Sustainable Development: Emerging Trends and Challenges ", International Journal of Environmental Research and Public Health 2018.

[59] Q.Wu ,"Incremental Routing Algorithms For Dynamic Transportation Networks", Department of Geomatics Engineering, January 2006, Universityof Galgary

[60] W.S.Xi," The Improved Dijkstra's Shortest Path Algorithm and Its Application", 2012 International Workshop on Information and Electronics Engineering, Proceeding Engineering, pp. 1186 – 1190.

[61] Y.Yang ,Z. Li ,X. Wang ,and Q. Hu," Finding the Shortest Path with Vertex Constraint over Large Graphs", Volume 2019, Article ID 8728245.

[62] M.T.Zar and M.M.Sein,"Public Transportation System in Yangon Region", 13th International Conference on Computer Applications (ICCA2015), 2015, pp. 335-339.

[63] M.Zaveri, "Exploring Data Structures: Graphs and its traversal algorithms", Jul 24, 2018.

[64] F. Zhan, "Three fastest shortest path algorithms on real road networks: Data structures and procedures, Journal of Geographic Information and Decision Analysis", vol. 1, no. 1, pp. 70-82.

[65] Carnegie Mellon University, School of Computer Science, Chapter 9,"Graphs: Definition, Applications and Representation".

[66] Myanmar Fire Service Department
(https://www.fsd.gov.mm)
[67] Myanmar General Administrative Department
(http://www.gad.gov.mm/en/content/total-list-districts-townships-sub-townships-towns-wards-village-tracts-and-villages-regions)
[68] Neo4j Graph Algorithms library
(https://neo4j.com/docs/graph-algorithms/current/algorithms/single-source-shortest-path/index.html)
[69] Tech Differences, "Difference between BFS and DFS", October 13, 2017.
[70] U.S. Department of Transportation Federal Highway Administration," Hazard Mitigation R&D Series: Article 1: Taking a Key Role in Reducing Disaster Risks", October 16, 2014.
[71] "Development of Model Intelligent Transport System Deployments for the ASIAN Highway Network", Bangkok, December 2017.
[72] "Euclidean distance", Technical White paper, September, 2005.
[73] Google Earth
(https://www.google.com/earth/)
[74] "Spherical Trigonometry and Navigation", Stony Brook University, MAT 336, History of Mathematics, Dec 6, 2006.
[75] "Three famous metrics", Machine Learning Weblog, 2013

LIST OF ACRONYMS

2D	Two Dimensional
3D	Three Dimensional
APSP	All Pairs Shortest-Path
ACO	Ant Colony Optimization
PSO	Particle Swarm Optimization
API	Application Programming Interfaces
ArcGIS	Aeronautical Reconnaissance Coverage Geographic Information System
BFS	Breadth First Search
DFS	Depth First Search
DKB	Dijkstra's Algorithm with Buckets
DKA	Dijkstra's algorithm with Approximate Buckets
DKD	Dijkstra's Algorithm with Double Buckets
GA	Genetic Algorithm
GIS	Geographic Information System
GPS	Global Positioning System
GUI	Graphical User Interface
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ID	Identification
JDK	Java Development Kit
JSP	JavaServer Pages
k-SPwLO	k-Shortest Paths with Limited Overlap
MATLAB	Matrix Laboratory

MMK	Myanmar Kyat
MSDP	Multicast Source Discovery Protocol
OSM	Open Street Map
PDF	Portable Document Format
SQL	Structured Query Language
WPS	Web Processing Service
QGIS	Quantum Geographic Information System
SSSP	Single-Source Shortest-Path
SMS	Short Message Service
SVG	Scalable Vector Graphics
SWT	Standard Widget Toolkit
XAMPP	Cross Platform of Apache + Maria DB + PHP + Perl
DB	Database
PHP	Hypertext Preprocessor

APPENDIX A

Software Installation and Configuration

In order to develop the proposed system, the required installation and configuration are described as follow.

1. Installing JDK

Install JDK on window and create system variable "Path" and set the location of the bin folder of the JDK installation.

C:\Program Files\Java\jdk1.7.0\bin

2. Installing Eclipse

Download and install the Eclipse\jee-mars4

3. Installing Apache Tomcat

i. Download and install Apache Tomcat
ii. Creates environmental variable "JAVA_HOME" and set it to JDK installed directory.
   C:\Program Files\Java\jdk1.8.0_131
iii. Create environmental variable "CATALINA_HOME" and set it to Apache Tomcat installed directory.
   C:\Program Files\apache-tomcat-7.0.37

4. Setting Up Tomcat Server in Eclipse

Create Tomcat Server in eclipse with

Server's hostname – localhost

Server name –Tomcat v7.0 Server at localhost

5. Installing XAMPP Control Panel for MySQL Server

XAMPP control panel version 3.2.2 is downloaded and installed to implement and configure MySQL database for the proposed system. It is free and easy to install and create databases using web browser**.**

6. Adding Required Jar Files

The required external jar files are added to implement the system. Table 6.1 shows the list of jar files which used to implement the system.

Jar Files	Explanation
mysql-connector-java-5.1.17-bin.jar	Enable to connect SQL database and server
gson-2.8.2.jar	Enable to display the stored data using java script object notation for map
junit-4.11.jar	Enable to write repeatable tests
orsoncharts-1.4-eval-nofx.jar	Orson Charts is a 3D chart library for the Java platform
orsonpdf-1.6-eval.jar	Enable to provide a Graphics2D implementation that generates PDF output
swtgraphics2d.jar	Enable to draw Graphics2D stuff on a swt composite
Jfreesvg 2.0.jar	Enable to generate graphical output in SVG format
hamcrest-core-1.3.jar	Enable to allow checking for conditions in the code via existing matchers classes

7. Generating Google Map API Key

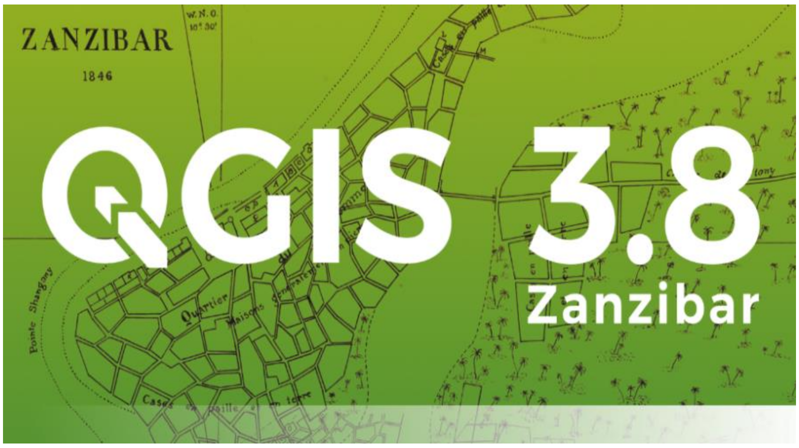
In order to use Maps JavaScript API in the proposed system a special Google Maps browser key is generated and used as follow.

```
<script src="https://maps.googleapis.com/maps/api/js?
Key=AIzaSyCWVdaIVl7M6wj4EP1BCBqbx6JWmxs9c0M
&callback=myMap"> </script>
```

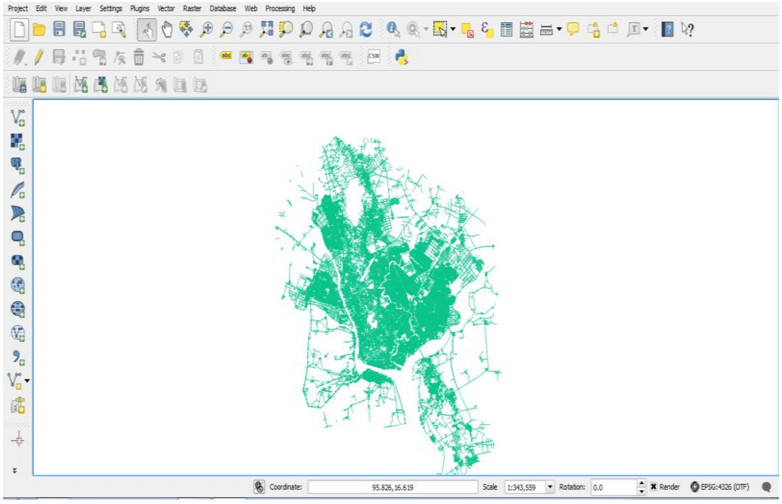
APPENDIX B

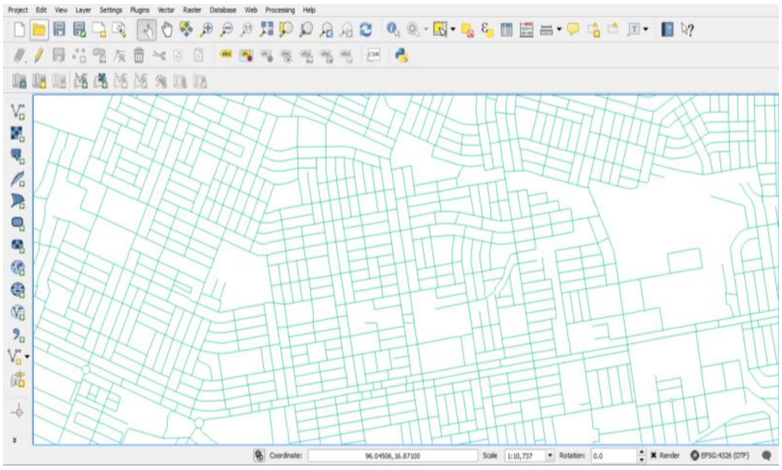
QGIS Software Installation and Data Creation

1. Navigate to https://qgis.org/en/site/forusers/download.html in web browser and download QGIS software and install.



2. Create road network line shape file of Yangon region.





3. Create Intersection Point of Road Network