Implementation of dynamic cost based routing for navigation under real road conditions using FOSS4G and OpenStreetMap

Sittichai Choosumrong[1], Venkatesh Raghavan[1] and Eugenio Realini[1]

[1]Graduate School for Creative Cities, Osaka City University
3-3-138 Sugimoto, Sumiyoshi-ku, Osaka 558-8585, Japan
Email: sittichai.ocu@gmail.com

ABSTRACT

On complicated road networks drivers need a map and navigation tools to efficiently compute routes to their destination points. Nevertheless, these navigation tools should take into account factors that would slow down or even stop the transit in specific roads, such as car accidents or natural disasters such as flooding or landslides. The happening of such events should be reported as soon as possible through the Internet, in order to update the road databases in each navigation device, so that it can compute alternative routes to let drivers reach their destinations as quickly as possible.

In this paper, an approach for finding shortest path, shortest time and safest path in a dynamic situation under real-time road conditions using pgRouting algorithm is proposed. The objective of this study is to enhance open source routing algorithms by modifying the functioning of pgRouting software. The entire system implements the dynamic routing using open source geospatial software (using the PostGIS and pgRouting extensions for the PostgreSQL DBMS). A number of variables, including traffic status and vehicle specification data are manipulated by the user to achieve shortest path, shortest time and safest path results. The system developed will be useful in traffic management and also emergency relief and rescue operations.

1. INTRODUCTION

Current routing systems in GIS software mostly provide routes that allow users to navigate between source and destination points, but sometimes the resulting routes do not take into account recent changes in the network data. For example, a road can be temporarily blocked because of car accidents or road works, blocking the driver from going through it. This problem often provokes significant loss of time and fuel, and especially in case of emergencies the real road network conditions should be taken into account for the computation, by representing not only the relationships between the transportation elements, but also the real-time and dynamic traffic restrictions in the road network.

The topic of this work is the implementation of a dynamic route planning system that depends on real-time road conditions by using exclusively Free and Open Source Software for Geospatial (FOSS4G). In particular, the shortest path algorithms is implemented by using pgRouting software. The main concern is to propose a study which provides an easy interface for the server and client, enabling the system to propose route planning as per user requirements. Therefore the system needs to respond to state variables that must be kept updated in order to represent the current situation of the road network in real-time.

2. METHODOLOGY AND DATA STRUCTURE

2.1 Method

The entire system completely relies on the FOSS4G stack. Figure 1 shows the components of a dynamic routing system framework. pgRouting software queries the route from the road database and the road database receives the real-time road condition from GIS clients. When the real-time data is received, the values in the database are updated and then a new route is computed.

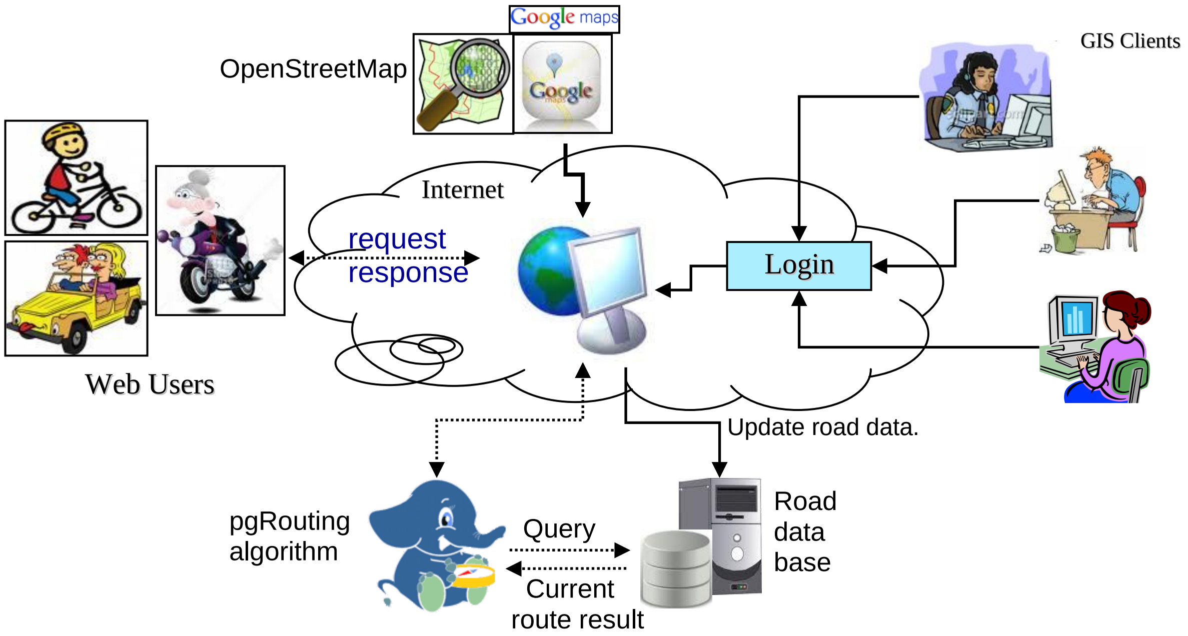

Figure 1. The components of the dynamic routing application.

2.2 Data

The computation of the routing in a dynamic way was obtained by implementing a "dynamic cost" value, depending on both dynamic and static features of each road segment. As for the dynamically changing component, a "road condition" column was added in order to enable or disable roads depending on their current state. As for the shortest time and safest route search component, three columns representing static features of the roads were added in the table of the road attribute data: the "road length", the "road type" and the "limited speed" columns were added to each network segment. The "road condition" attribute column is employed in order to keep such dynamically changing data apart from the main (static) spatial data (Figure 2 shows the structure of cost values). Figure 3 shows the system routing framework.

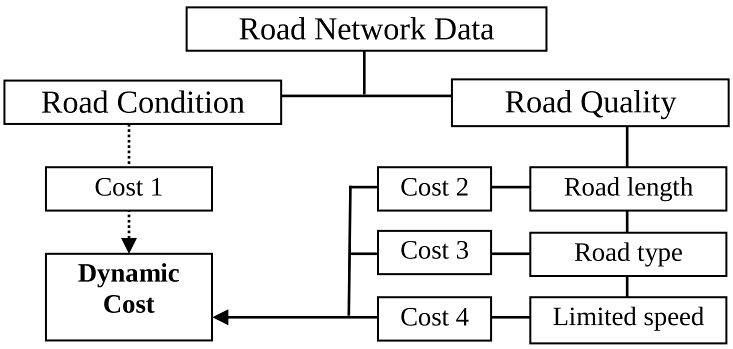

Figure 2. The structure of the cost attribute for roads.

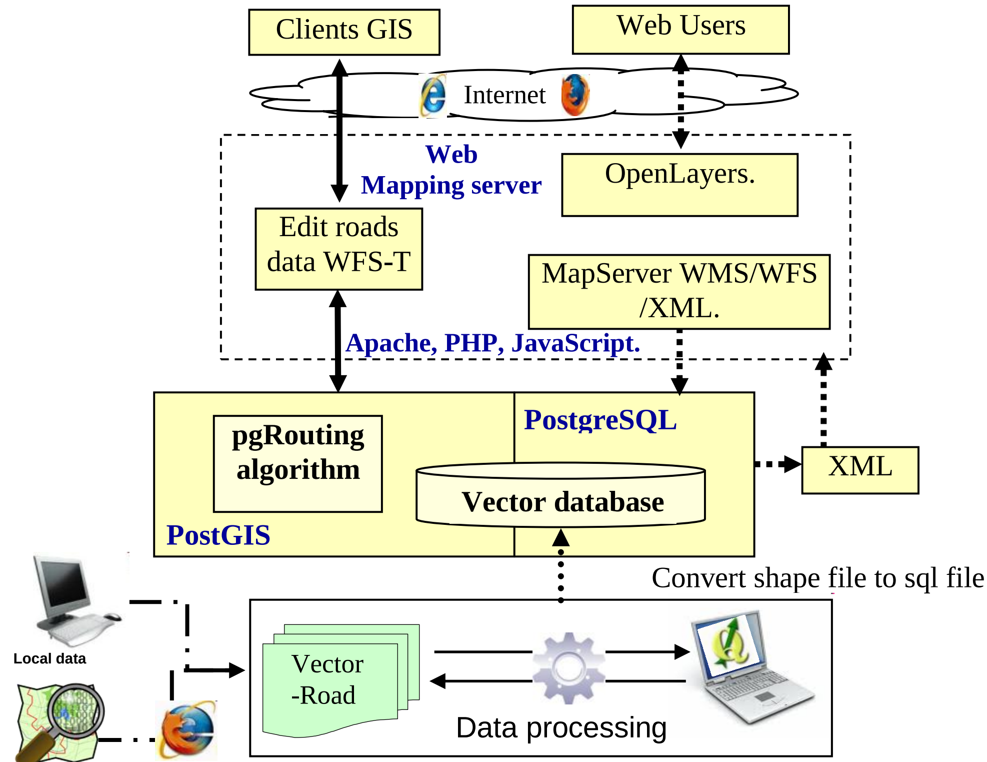

Figure 3. System Overview.

2.3 Shortest Path Search

2.3.1 Static Model

Algorithmic model of shortest path search.

The algorithm to find the shortest path is based on a graph, G = ({V}, {E}) where V are the nodes and E are the edges, defined by couples of nodes. Each edge E(i,j) has a weight (or cost) W(i,j), which in its basic form represents the distance between the two nodes of the edge. Each node is assigned a value which is the sum of the weights from the start node to the node itself. Given a network with known edge weights, the shortest path problem consists in finding the shortest distance from a source node A to a specific node in the node set V. In particular, this path is selected out of all the possible paths by choosing the one for which the sum of its weights is minimum. Figure 4 shows an example of finding shortest path in a weighted network.

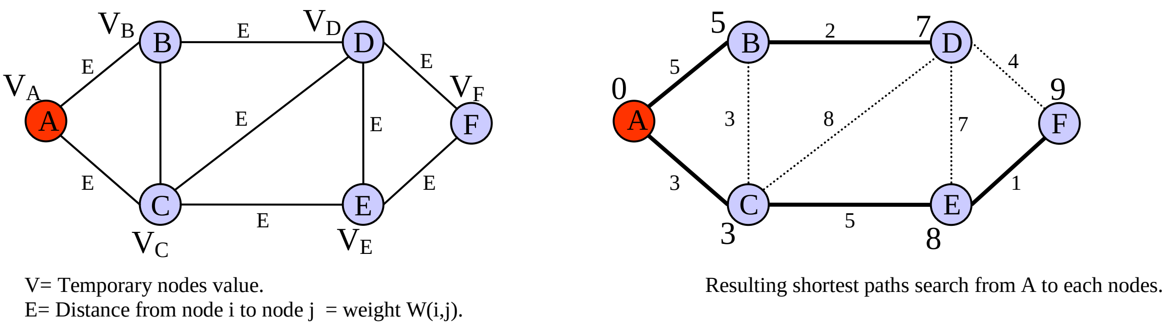

Figure 4. Basic shortest paths search.

Table 1. Resulting shortest paths from A to each node.

Node	A	B	C	D	E	F
Distance	0	5	3	7	8	9

2.3.2 Dynamic Model

Increasing costs value of an edge.

To recalculate the shortest paths we update the graph by increasing the cost W(i,j) of an edge E(i,j) by a value C(i,j) where the roads are blocked or in difficult conditions. For example, if the roads is blocked the road condition column value in the database will be increased. The pgRouting algorithm needs to be modified to perform queries including the road condition column to compute the new weight. Figure 5 shows the example graph if the edge E(V_A, V_B) is the blocked road. The cost of edge E(V_A, V_B) will be increased to infinity. When the algorithm computes again the affected vertices of the second computation are V_B and V_D because the cost of those nodes will change to the new weights of node B=6 and node D=8. Non-affected vertices are V_C,V_E,V_F, because in this case from the start point A to nodes V_C,V_E,V_F there is no need to go through edge (V_A, V_B). For other dynamic events which do not block completely a road, such as traffic jams, the cost will be changed in the "limited speed" column.

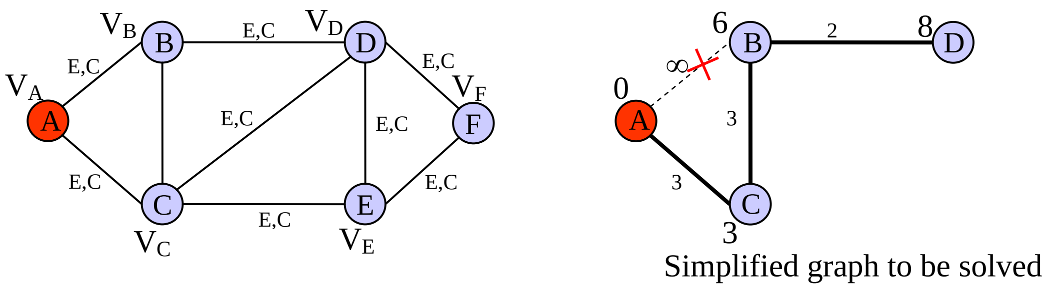

Figure 5. Shortest path search with the dynamic cost.

3. RESULTS

A web interface was developed using JavaScript, PHP, and OpenLayers (Figures 6, 7, 8). Figure 6 illustrates the shortest path from A to B by simply by computing the shortest path algorithm. Figure 7 shows a new route if a dynamic event occurred on one of edges along the original shortest path from A to B. In this example the black line (edge number 1025) is a blocked road, which blocks traffic from passing through it. The cost will be increased to infinity. The result of the experiment shows that the efficiency of route finding is improved since with the new method the result avoids this road and give a new (dynamically updated) route to the user. Figure 8 shows the result obtained by avoiding small roads as much as possible and by not caring about the distance from A to B. This algorithm combines the shortest path as explained before (also taking into account blocked roads) with the values of the "road type" (and eventually also "limited speed") column to get the safest and as short as possible path.

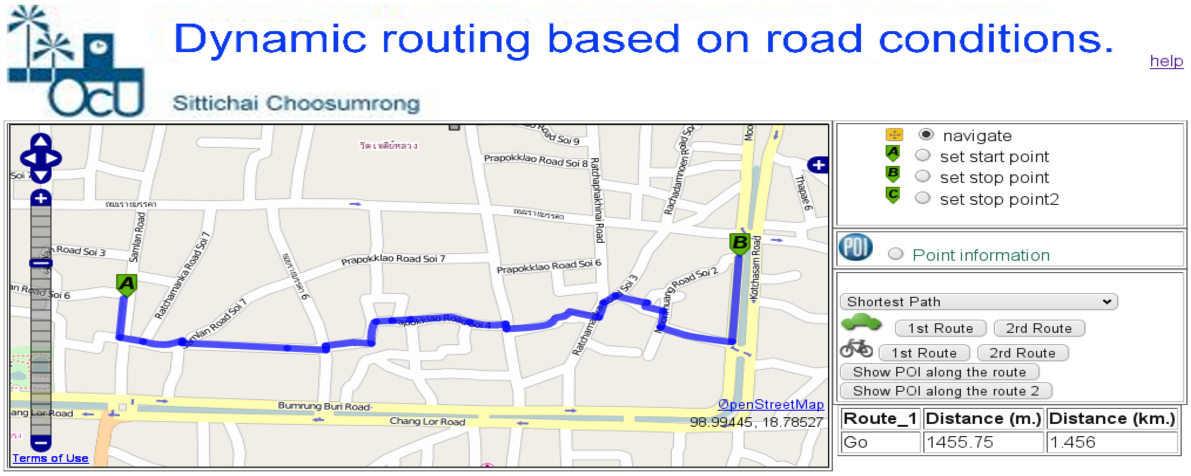

Figure 6. Shortest path from location A to B.

Figure 7. Shortest path from location A to B after dynamic change in road conditions.

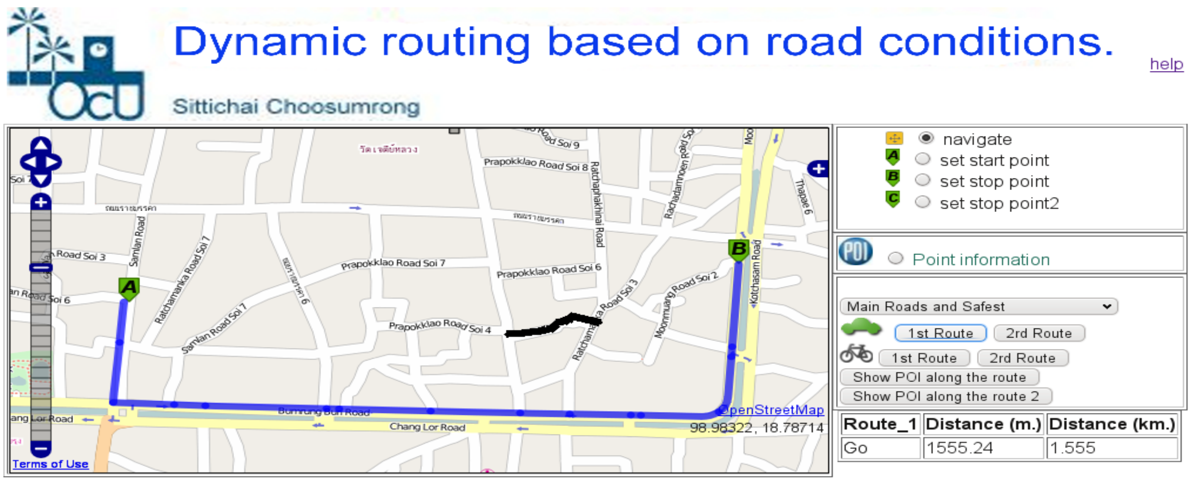

Figure 8. Path computed from A to B with safest path algorithm.

4. DISCUSSION AND CONCLUSIONS

pgRouting algorithms, such as Dijkstra algorithm, were enhanced, by modifying them for taking into account dynamic changes in the road conditions. This paper suggests a new concept of calculation of shortest path, shortest time and safest path routes that supports dynamic changes information with pgRouting to develop a routing algorithm for solving the real road network condition problem. The initial results are showed in section 2. On the basis of a weighted network representation, the dynamic cost change will act directly on edge weights by increasing them in real-time of an amount defined in the road condition column.

An alternative way could be to use True/False Boolean value to set whether a road is blocked or not. The normal roads condition value would be set to False value, but if the roads is blocked the values is changed to True. If the value is False the cost is 0 and if the value is True the cost becomes infinity.

The system has been developed using FOSS4G. Moreover, this system also provides POI (Point of Interest) management functionality which can store POI in DBMS and show on the map when the user got the route result and asked for the POI along the route result.

Further studies will focus on other problems related to data modeling for vehicle navigation, such as turning restriction modeling. The system could be useful for managing and regulation traffic flow and also, with further enhancements, for emergency relief and rescue.

5. ACKNOWLEDGEMENTS

The first author would like to thanks the Ministry of Education, Culture, Sport, Science and Technology (MEXT) of Japan for providing the Japanese Government (Monbukagakusho) Scholarship under which this work was carried out. The authors would like to acknowledge to Dr.Sarawut Ninsawat for his help in preparing this paper.

6. REFERENCES

An Introduction to pgRouting with OpenLayers. Retrieved October 2010 from the World Wide Web: http://pgrouting.postlbs.org/wiki/WorkshopFOSS4G2007/

Choosumrong S., Raghavan V., Realini E., (2010), Implementation of dynamic cost based routing for navigation under real road conditions using FOSS4G and OpenStreetMap. Proceedings of Geoinforum 2010, Tokyo, Japan, 22-23 June: Geoinformatics 21(2), pp. 108-109 (ISSN 0388-502X)

IA Musliman, AA Rahman, V Coors, (2007), Modeling Dynamic Weight For 3D Navigation Routing. Retrieved October 2010 from the World Wide Web http://www.gisdevelopment.net/technology/lbs/ma07223.htm

Matthew P. (2008), The development of a Geographic Information System for traffic route planning using Location Based Services, pp.69-116