REAL-TIME COST UPDATES BASED ON CURRENT TRAFFIC CONDITION FOR OPTIMAL ROUTING PLANNING

Sittichai Choosumrong[1] and Venkatesh Raghavan[1]

[1]Graduate School for Creative Cities, Osaka City University3-3-138 Sugimoto,
Sumiyoshi-ku, Osaka 558-8585, Japan
Email: sittichai.ocu@gmail.com

KEY WORDS: Dynamic Cost, Shortest Path, Optimal Routing, FOSS4G

ABSTRACT: 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 the recent changes in the network data. For example, a road can be temporarily blocked because of car accidents or road works and the driver cannot go through that road. 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 because it represents not only the relationships between the transportation elements, but also the real-time and dynamic traffic restrictions in the traffic network. 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. The dynamic cost for the road navigation routing system are needed for route selection and driver guidance. That is, when the driver defines two points on a given road network, the system should provide an optimum path connecting these two points based on the multiple impedances, such as the geometric distance, time (speed limit), as well as the traffic conditions.

The objective of this paper looks at enhancing pgRouting algorithm to searches of shortest path, time and safety route that supports dynamic changes information with pgRouting to develop a routing algorithm for solving of the real road network condition. Based on the concepts described in this paper, implementation of the routing based on dynamic costs (current traffic condition) was demonstrated. The system implemented on FOSS4G stack and can be customized for a wide variety of needs.

INTRODUNTION

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, because in a traffic network a road can be temporarily blocked because of car accidents or road work and drivers cannot go through that road. If the navigation system did not update the information and take into the database, the routing result might be not given up-to-date traffic condition routing result.

The objective of this work is the implementation of a dynamic route planning system that depends on real-time road conditions by using Free and Open Source Software for Geospatial (FOSS4G). In particular, the shortest path algorithm is implemented by using pgRouting software. pgRouting is a library which support for shortest algorithms like Dijkstra, a-star, shooting-star. These algorithms assume that the edge weights are static and their values do not change with the current road condition. pgRouting extends the PostGIS/PostgreSQL geospatial database to provide geospatial routing functionality.

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. Moreover, currently network enabled mobile devices become mostly used. Fortunately, it should be useful for a driver if we could update the real road condition via such devices to give the current road condition because it is an expedient to update a traffic information such as a taxi driver if they meet some accident or some problem they can update traffic road condition via android devices in the real time.

This system uses underlying map (road connectivity) data from OpenStreetMap (OSM). OSM creates and provides free geographic data such as street maps to anyone who wants them. Moreover, OSM offers a good data source to use for pgRouting, because it's freely available and has no restrictions in terms of processing the data. Data availability still varies from country to country, but the worldwide coverage is improving day by day. Data structure used in OSM are including "nodes" that a points with a geographic position and "ways" that lists of nodes, representing a polyline or polygon.

METHODOLOGY AND DATA STRUCTURE

Method

The entire system completely relies on the FOSS4G stack. Figure 1 shows the system framework. In this framework;

- Web Mapping Server was implemented using Open Source Software such as OpenLayers, Apache, PHP, MapServer.
    - Openlayers, PHP, and JavaScript are used for creating a web mapping.
    - QGIS used for download and modified the road network data.
    - pgRouting algorithms used and enhanced to allow the new routing algorithms.
    - PostGIS used for a spatial data extension for PostgresSQL
    - PostgreSQL used for storing of all spatial and non-spatial data. Moreover, PostgresSQL integrating with pgRouting function to query the routing result.

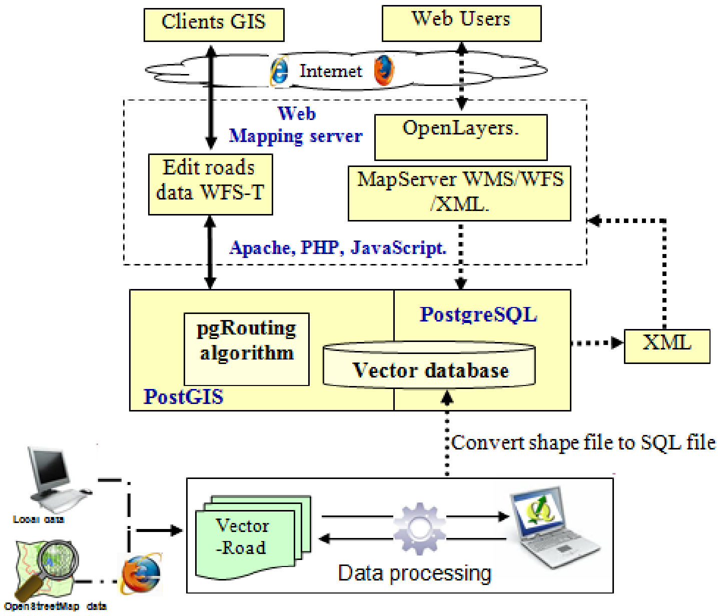

Figure 1 System framework

Figure 2 shows the components of a dynamic routing application. 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.

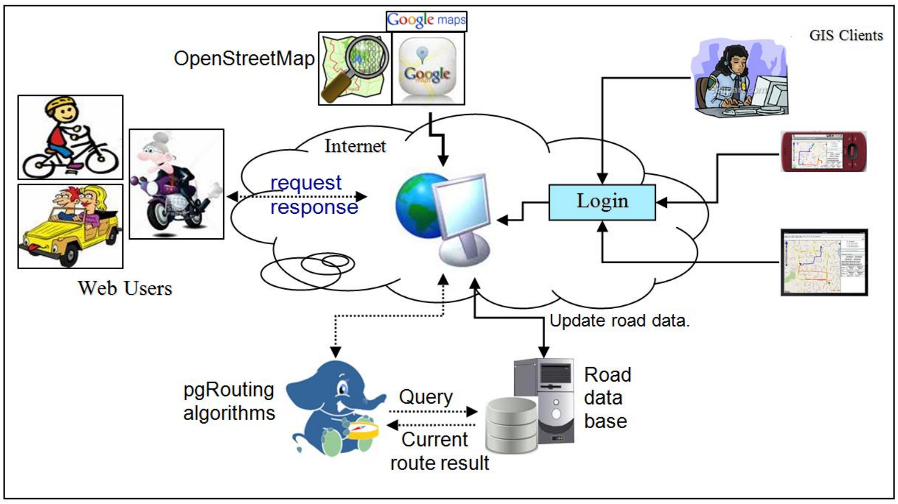

Figure 2 The components of the dynamic routing application

Modify the Road Network Data

Basic data structure that pgRouting functions request are gid, source, target, cost, x1, y1, x2, y2, reverse_cost, the_geom and pgRouting compute the shortest path from cost which define the length of each road segments. Thus, the cost values needs to modify to give the dynamic route data. The computation of the routing in a dynamic route was obtained by preparing a "dynamic cost" value, depending on both dynamic and static features of each road segment. As for the dynamically changing component, a "traffic condition" column (Cost B) was added in order to enable or disable roads depending on their current state. As for the travel time and safety route search component, two columns features of the roads were added in the table of the road attribute data, the "speed limit" (Cost C) and the "road types" (Cost D) columns were added to each network segment. The "traffic 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 dynamic cost values).

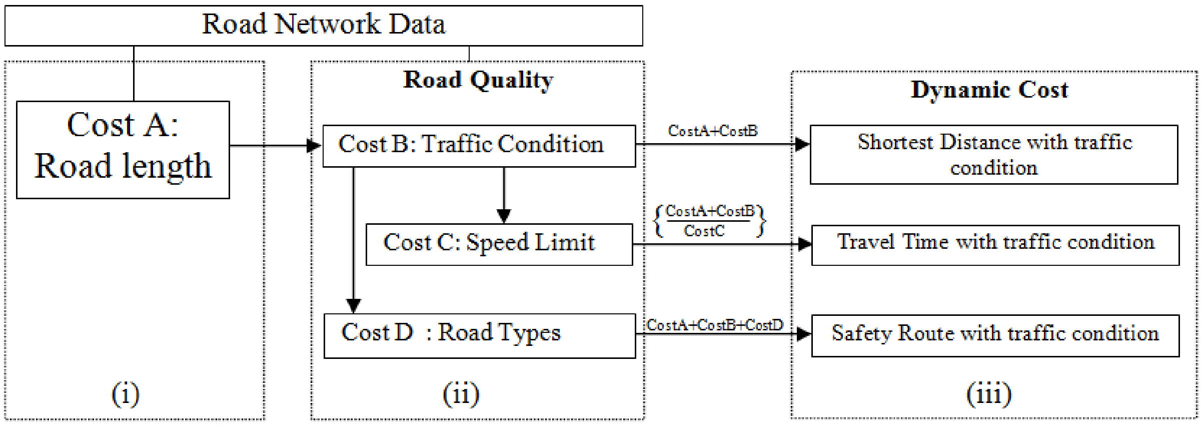

Figure 3 Structure of the Dynamic Cost Values
i) CostA=Distance of the road, ii)=More variables added into the road data to compute the dynamic result, iii)=New functions with dynamic traffic condition on the road network.

Shortest Path Search in a Real Road Network

As mentioned before the routing graph might change over time due to dynamic traffic, dynamic conditions on the roads network. As we will see, the two most important changes are increasing and decreasing the cost. Other events can be modeled based on these two operations. As we known the pgRouting algorithms compute the graph by searching the minimum cost of all the road segment connecting between start node to end node. This algorithm finds the length of a shortest distance from node **a** to node **z** in a connected weighted (Cost) graph:

- Let w(i, j) > 0 denote the cost of edge (i, j) (edge is a road segment on the real road network);
- Let L(x) denote the label of node x;
- At termination, L(z) is the length of a shortest path from a to z.

Table 1 Shortest Path Search Algorithm Processes

Input:	A connected weighted graph in which all weights are positive Nodes a to z
Output:	L(z), the length of a shortest path from a to z

Algorithm:	procedure dijkstra (w, a, z, L)
	    L(a) = 0
	    for all nodes x ≠a
	        L(x) = ∞ (infinity)
	    V = set of all nodes
	    // V is the set of nodes whose shortest distance from a has not been found
	    while z ∈V
	        begin
	            choose v ∈V with minimum L(v)
	            V = V – {v}
	            for each x ∈V adjacent to v do
	                L(v) = min {L(x), L(v) + w(v, x)}
	        end
	end

Static Model

The algorithm to find the shortest path (shortest distance on the road network) is based on a graph that contains set of nodes V and set of edges E (edge on a graph denotes a road segment in the real-road network), defined by pair of nodes. Each road 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 road segment. Each node is assigned a value which is the sum of the cost from the start node to the node itself. Given a network with known road cost, the shortest path problem comprises of 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 cost is minimum. Figure 4 shows an example shortest path searches model in a cost network and the shortest path result had shown in Table 1.

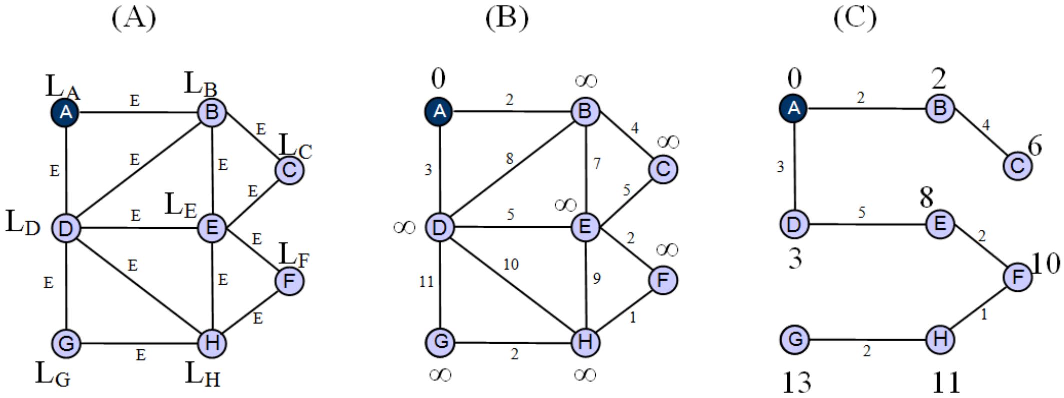

Figure 4 Basic Shortest Path Searches Model
A) Sample network graph, B) Initial edge and nodes value,
C) Minimum cost values of each node from start nodeA.

Table 2 Result of Shortest Paths from Node Start (A) to each Node from Figure 4

Node	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**
Cost	0	2	6	3	8	10	13	11
Pred(v)	0	A	B	A	D	D,E	D,E,F,H	D,E,F

Note: Pred(v) = previous vertex

Dynamic Model

To recalculate the shortest path (shortest distance), we update the graph by increasing the Cost W(i,j) of and edge E(i,j) (a road segment) by a value C(i,j) where the roads are blocked or in difficult conditions. For example, if the road is blocked the road condition column value in the database will be increased. The pgRouting functions needs to customized to perform queries including the road condition column to compute the new cost. Figure 5 shows the example graph if the road E(VA, VB) is the blocked road. The cost of road E(VA, VB) will be increased to infinity. When the function computes again the affected nodes of the second computation are V_B and Vc because the cost of those nodes will change to the new cost of node B and node C. Non-affected vertices are VD, E, F, H, G because in this case from the start point A to nodes VD, E, F, H, G there is no need to go through road (VA, VB). For other dynamic events which do not block completely a road, such as traffic jams, the cost will be changed in the “speed limit” column.

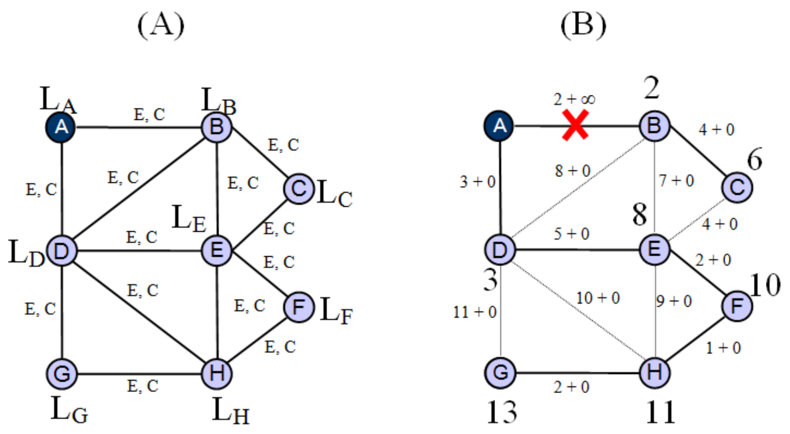

Figure 5 Shortest Path Searches Model with the Dynamic Cost
A) Compute values including the dynamic cost, B) The old result before dynamic cost change,
C) Result changed with effected nodes.

To accesses the system via an portable devices, the Android SDK and VDK Manager need to implemented. Before getting started with the Android SDK, you might need to install the JDK, if you don't have it already. Unfortunately, if we would like to accesses the system via portable devices we need to edit some JavaScript code to make the system can be used with such devices.

RESULTS

A Web Interfaces was developed using JavaScript, PHP, and OpenLayers. The clients GIS interfaces for this system are designed taking into account of cost values. Interfaces are designed with easy navigation and edits cost values. Figure 6 and Figure 7 shows the system interface from Android and iPad emulator devices.

Figure 6 Andriod Emulator showing the route from A to B and B to C

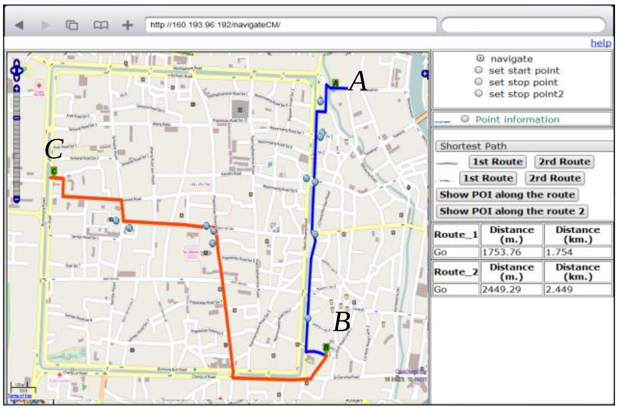

Figure 7 iPad Simulator showing the route result and cost Point of Interest from A to B and B to C

Figure 8 shows the shortest distance from A to B simply by computing the shortest distance function. This function considered only the shortest distance from A to B and do not consider about the road type and speed limits. Figure 9 shows the Client GIS interface update road data if the dynamic event occurred on the real road condition. In this example the road segment number 661 is a blocked road because of car accident, which blocks traffic from passing through it. To update the cost value of road segment number 661. Client GIS needed to login verification before being able to access the client GIS interface. After update the road data the cost will be increased. 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 gives a new (dynamically updated) route to the user. Figure 10 shows a new route after computed again if a dynamic event occurred on one of road segment along the original shortest distance from A to B (road segment number 661) by avoiding the road segment number 661 where temporarily blocked. The black line is shows the road blocked and cannot pass through that segment on the road network.

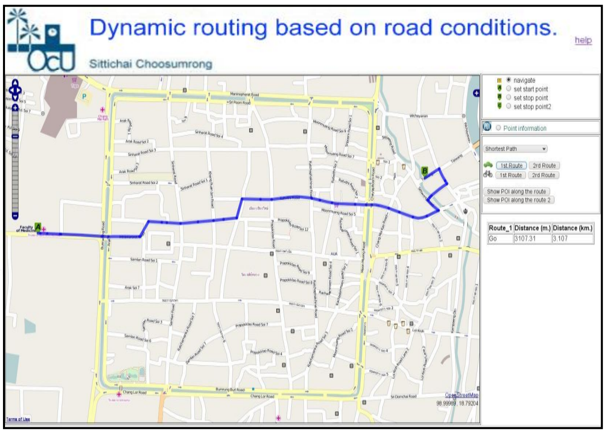

Figure 8 Shortest Distance Result from Location A to B

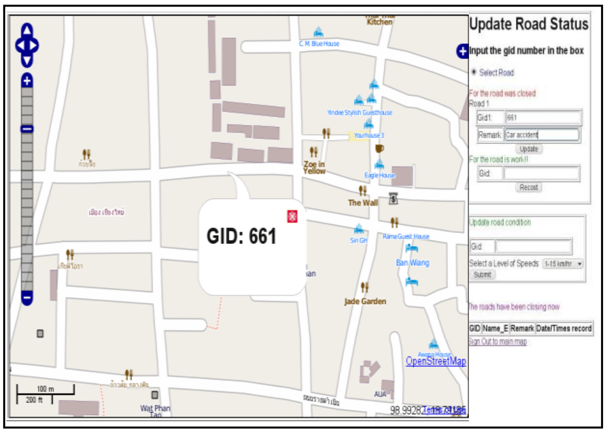

Figure 9 Sample Update Cost Values on Web Interface

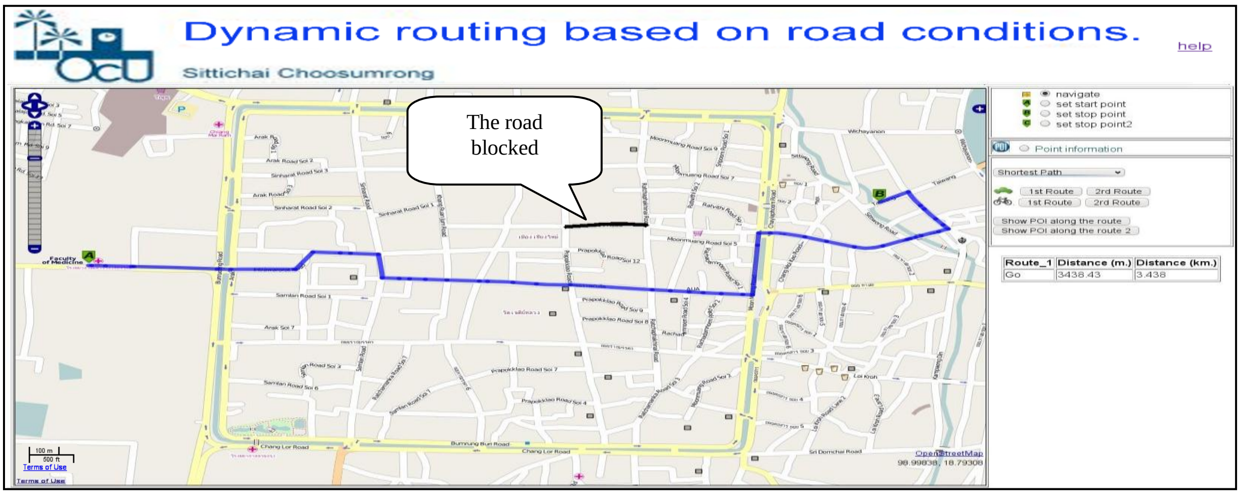

Figure 10 Result Change after Dynamic Changed on the Road Condition

CONCLUSIONS

pgRouting algorithms, such as Dijkstra algorithm, were enhanced, by modifying them for taking into account dynamic changes in the road conditions. This paper proposes a new method of calculation of shortest path, shortest time and safety path routes that supports dynamic changes information with pgRouting to develop a routing algorithm for solving the real road network condition problem. 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. The system also integrated with Android SDK that can access the system via network enabled portable devices. Further studies would be focus on other problems related to data modeling for vehicle navigation. pgRouting could be implemented as Web Processing Service (WPS) using ZOO platform (http://zoo-project.org/).

REFERENCES

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 GIS-IDEAS 2010, Hanoi,Vietnam, 9-11 December 2010, JVGC Technical Document No. 5, pp. 53-58 (ISBN 978-4-901668-76-7).

Choosumrong, S. and Raghavan, V., 2011. Optimal Traffic Routing Based on Real-time Cost Updates for Current Road Conditions. Proceedings of Geoinforum 2011, Osaka, Japan, 23-24 June: Geoinformatics 22(2), pp. 66-67 (ISSN 0388-502X).

Raghavan, V., Masumoto, S., Ninsawat, S., Yoshida, D., Nonogaki, S., Yonezawa, G. and Binh, N.H. (2007) Training Note on Spatial Data Sharing using Free and Open Source Software 2006.