IMPLEMENTING 3D NETWORK ANALYSIS IN 3D-GIS

Ivin Amri Musliman[1],Alias Abdul Rahman[1],Volker Coors[2]

[1]Dept. of Geoinformatics, Faculty of Geoinformation Science & Engineering, Universiti Teknologi Malaysia, 81310 Skudai, Johor, Malaysia. - (ivinamri, alias)@fksg.utm.my

[2]Faculty of Geomatics, Computer Science and Mathematics, Hochschule für Technik, Schellingstr.24, 70174 Stuttgart, Germany. - volker.coors@hft-stuttgart.de

Commission ICWG II/IV

KEY WORDS: 3D Network Analysis, 3D Navigation, Dynamic Networks, 3D-GIS

ABSTRACT:

Way finding or routing has been always used by common people in navigating from one place (of origin) to another (destination). And most of them are in two dimensional routing or network. In orthogonal concept, two-dimensional (2D) or three-dimensional (3D) routing uses planar or non-planar graph and spatial extend, where the third dimension is used to calculate the weights for the edges in the graph. Usually routing is done in a planar graph embedded in 2D space. Sometimes it is extended to non-planar graphs to model bridges, etc. but still embedded in 2D space. Currently, most of shortest path algorithm used in GIS application is often not sufficient for efficient management in time-critical applications such as emergency response applications. It doesn't take into account dynamic emergency information changes at node/vertex level especially when applying in emergency situations such as large fires (in cities or even in buildings), floods, chemical releases, terrorist attacks, road accidents, etc. This study is based on finding shortest path route in a dynamic situation for indoors and outdoors using Single Sink Shortest Path (SSSP) routing algorithm. The main focus of the research is to implement 3D shortest path analysis in 3D data set. A program is developed to calculate the shortest path in 3D data set with the support of dynamic events that might or will occur at each of calculated routes.

3 INTRODUCTION

Network analysis function in GIS is still in 2 dimensional (2D) environment. It uses 2D or 2.5D data (e.g. road layer) to find and simulate the shortest path route. In some other situation, road networks layer were overlaid with Triangular Irregular Networks (TIN) for best visualization scene, but the shortest path results are still using 2D horizontal plane for the distance calculation. Shortest path algorithm such as Dijkstra is the most common algorithm used in finding shortest path which has a lower computational complexity (Zhan & Noon 1998). Over the recent years, many researchers (Demetrescu and Italiano 2004 & 2005, King 1999, Ramalingam and Reps 1996) have devoted to developed and improving current Dijkstra algorithm for various purposes (e.g. static and dynamic shortest path problems) which it is still under fundamental algorithmic techniques that are the kernel of shortest paths. GIS software developers used these algorithms (mainly static shortest path algorithm) for their 2D network analysis engine. The approach still uses 2D data model and the information integrated into the routing algorithm is limited and therefore it cannot be used in 3D network analysis practice, thus it is impossible to do navigation in 3D-GIS environment.

There are a few works done in 1998 by Coors and Flick and Köninger and Bartel and Zlatanova on 2000 related towards 3D GIS for web-based and urban development, but these researchers' only focuses on concepts, establishing framework and its application from a bigger scope of view. While in 2003, Altmaier and Kolbe, Kolbe and Gröger, Kolbe et. al. 2005, Döllner et. al. 2006, and Döllner and Hagedorn 2007 were engaging pretty much on the development of virtual 3D city model using different approaches (such as CityGML) and its application for managing and integrating urban information, and visualizing and navigating (in 2D) the city models. Since data management and visualization is the focus, there is a need for a real 3D routing from the research idea which it can be extended towards 3D navigation for 3D-GIS.

A few researches on navigation in 3D have been initiated recently. Research of 3D pedestrian navigation model by Lee (2005), Kwan & Lee (2005) shows the intention of using 3D technology for indoor emergency response in a 3D building environment. Due to the lacks of the effective and comprehensive consideration of multi-dimensional and dynamic emergency information in vehicle emergency routing, Zhu et al. (2006) has proposed an algorithm which based on the functional requirement analysis of 3D vehicle emergency routing. It only discusses the issues on 3D dynamic network and emergency-related information discovery and integration in near 3D-GIS environment. The research combines transportation data model and using Multi-Criteria Evaluation (MCE) approach to form a table of matrix for scoring and weighting the three major physical factors, namely, slope gradient, lane number and traffic light number as 'best' alternative shortest path route.

In 2006, Ivin et al. had discussed the anticipated and initial requirements for making 3D navigation in 3D-GIS environment possible. Some research gaps were identified. Later in 2007, Ivin et al. again try to model dynamic weight of routes (road networks) and implemented an incremental single sink shortest path (SSSP) algorithm that is used for finding shortest path in dynamic routes. Although the proposed algorithm are in 2D planar graph which is usable for indoor routing especially for building evacuation in case of disaster occurs, the approach can be extended use for 3D navigation in 3D-GIS. A dynamic graph algorithm maintains a given property P on a graph subject to dynamic changes, such as edge insertions, edge deletions and edge weight updates. A dynamic graph algorithm should process queries on property P quickly, and perform update operations faster than recomputing from scratch, as carried out by the fastest static algorithm. An algorithm is fully dynamic if it can handle both edge insertions and edge deletions. A partially dynamic algorithm can handle either edge insertions or edge deletions, but not both. It is incremental if it supports insertions only, and decremental if it supports deletions only.

2 CURRENT STATUS ON 3D NAVIGATION

Way finding or routing has been always used by common people in navigating from one place (of origin) to another (destination). And most of them are in two dimensional routing or network. In orthogonal concept, 2D or 3D routing uses planar or non-planar graph and spatial extend, where the third dimension is used to calculate the weights for the edges in the graph (Ivin et al, 2007). Usually routing is done in a planar graph embedded in 2D space. Sometimes it is extended to non-planar graphs to model bridges, etc. but still embedded in 2D space. For example, if the network model type is planar (directed) graph, where the node as 0D coordinate (x, y) in 2D space, therefore objects such as bridges, flyover, etc. cannot be modeled. While for non-planar (directed) graph, where node as 0D coordinate (x, y) in 2D space, objects such as bridges, flyover, etc. can be modeled but street length cannot be derived from the model (directly). And as for 3D space (which is the target of this study), objects such as bridges, flyover, etc. can be modeled in a non-planar (directed) graph, with node as 0D coordinate (x, y, z). Street length can only be derived from the model, if arcs are close to original street geometry, which is not necessary for routing. For routing, a node is necessary at each junction etc. A slope road with no junction still can be modeled with two nodes and one arc for routing purposes even if it is long and has a lot of serpentines as usual in the mountains but this is the geometry part of the network.

Routing in 3D is almost impossible at the beginning, but with the emergence of new models and concepts (Sharkawi and Abdul Rahman 2007, Ujang and Abdul Rahman 2007, Zhu et. al. 2006, Ivin et. al. 2006, Döllner et. al. 2006, Kolbe et. al. 2005, Shi and Zlatanova 2005, Zlatanova et. al. 2005, Liu et. al. 2005, Zlatanova and Holweg 2004, Nebiker 2003, Altmaier and Kolbe 2003, Zipf and Schilling 2003, Coors and Jung 1998) and improved algorithms for way finding and navigation (Ivin et. al. 2007, Shi and Zlatanova 2005, Demetrescu and Italiano 2004 & 2005, Ramalingam and Reps 1996) together with the development of communication and positioning technologies (Worboys and Duckham 2004), it has made researchers expand the focus of this research area. In finding a shortest path and doing navigation in 3D network, one should consider dynamic changes that might and/or could occur on the road network, i.e. road accidents, traffic jams, floods, etc. These dynamic changes also do appear in pedestrian navigation mode (which defines as navigation across terrain allowing searching to cross boundary of road networks).

The concept of traditional navigation data model (in 2D) described as relation among entities related to navigation (Liu et. al 2005) which proposed methods to design and manage navigation database. It was commonly known and represented as node-arc model by a set of nodes and a set of arcs. Whereas 3D navigation data model incorporates the height values, z, in each of every node neither on surface nor subsurface thus allows more complicated representation and network analysis. Most navigation data models are developed based on general data models of GIS, which mainly refers to discrete entity model and network model (Goodchild 1998). According to the different characteristics, network model is generally classified as three types: planar, non-planar and 3D network model. Traditional 2D navigation uses routing algorithm which is characterized by the use of planar or non-planar network model while 3D navigation uses routing algorithm bases on 3D network model. As a result, 3D networks overcome the problems of 2D networks such as 3D structures e.g. overpasses or underpasses are better represented and the true distance is measured across sloping or hilly terrain. It is difficult for traditional navigation data models to handle several problems encountered in navigation applications, such as process of dynamic attributes (Goodchild 1998), complex feature representation, consistent representation of multi-scale topological relations, highly effective data storage and non-planar feature representation (Liu et. al 2005). Researchers over the world has been working on the problems faced and produce new data models and technologies for the navigation data model construction such as Liner Reference Systems (LRS), dynamic segmentation, feature-based data model, network data model of ArcGIS and hyper graph data model. However, none of them completely solves the problems of traditional navigation data models. More data models or standards that can be used for navigation services come forth. Here lists some of them: ISO-GDF (Geographical Data Files) (see GDF4.0 manual), SDAL (Shared Data Access Library) of NavTech Co. (NavTech), GIS-T Enterprise (Dueker and Butler 1997), and lane-based data model (Fohl et. al. 1996).

2.1 Concept of Dynamic Weight

In dynamic graph with time dependent G{V(t), E(t)} a node/vertex may be removed or added, depends on the dynamic changes that occurred real-time on the graph especially in buildings. Weight of the edges is also changes over time. For example, in an emergency situation, e.g. fire in a specific floor in a building, elevators will be assigned as a dynamic node/vertex. In case of a fire incident, it is advised to use the stairs instead of elevators. Considering the complexity of modern buildings and the great numbers of people that can be inside the buildings, it is rather difficult to organize a quick evacuation.

Vehicle routing problem. In this case, destination node is fixed, whereas the start node would be dynamically located anywhere on the map. For example, call for a taxi; a person calls a taxi company and gives his/her location to the operator. Then the operator will searched and dispatched the nearest taxi to the given location and informed the caller the estimated time of arrival of the taxi. With a dynamic factor such as road traffic taking into account, the assigned taxi might not be able to arrive at the specific location in time. Other available taxi (second nearest) will then be notified, and the process goes on until a taxi reached the caller. Algorithm that will be used for this type of problem will have one-to-many type Therefore this type of case is much easier than the following problem.

Rescue operation planning. Would the model be able to assign each rescuer to survivor(s) in a rescue operation and evacuation process with known or unknown number of survivor in case of a fire situation? The model of a building can either be an abstraction of a building is represented with polygons in 3D space (Zlatanova et. al 2004) which is likely geometry or topology model, or a logical model, which represents the connections between the rooms. The rooms and important crossings are represented with nodes; the paths are represented as links between nodes

The requirements of real-time, more accuracy and individuation for dynamic navigation can be met in the situation. There are at least two elements to realize dynamic navigation in vehicle navigation. Firstly, is the data that reflects the real-time traffic information. Secondly, the effectiveness and highly precise algorithms of short time traffic prediction. Dynamic navigation for vehicle navigation relies on the real-time traffic data (Liu et. al. 2005). Zhu et al. (2006) have addressed a model for emergency routing for escape plan. In this model, it emphasis on multi-dimensional and dynamic routing algorithm for vehicle emergency. The algorithm is based on the functional requirement analysis of 3D vehicle emergency routing. Detailed discussion about the efficiency analysis of shortest path algorithms can be found in Zhan and Noon's (1998) research.

3 3D DYNAMIC NETWORK MODEL

In short, the overall process of building a 3D dynamic network can be shown briefly in three steps. Firstly, from planar and non-planar networks to 3D networks, the ambiguous situation of under/overpass and network overlay in the 2D graph can be clarified. It improves the abilities of visualization and effective and comprehensive data integration. Secondly, by moving from a static network to a dynamic network, real-time information about the vehicle and other events can be directly integrated into the routing process. Lastly, after optimization, the unnecessary vertices are eliminated and the total number of vertices is greatly reduced, thus improve the ability of rapid response. For indoors, distances can be calculated between 2 node coordinates. Elevators have to be models as vertex per floor with connecting edges. Stairs have to be models as edges. It can happen that two parallel stairs connect the same nodes therefore parallel edges are allowed. Next is describing the 3D road network data model. It is based on point features derived from Digital Surface Model (DSM) and road networks layer (line feature). Both data is modeled to suit for 3D network analysis function where a network optimization process, i.e., dangle points will be detected and then optimized, its network connectivity changed and some impassible segments have been eliminated. Distances can be calculated between 2 node coordinates. Elevators have to be models as vertex per floor with connecting edges. Stairs have to be models as edges. It can happen that two parallel stairs connect the same nodes. This means that E is no set any more. In a set, each element is unique! But in this case two nodes are connected by two distinguishable edges E1 = (Vi,Vj) and E2 = (Vi,Vj); this is similar to a model with lanes. Parallel edges are allowed.

3.1 Anticipated 3D Routing Algorithm Implementation

As mentioned before the routing graph might change over time due to specific events. As we will see, the two most important changes are increasing and decreasing the costs of an edge and inserting a new edge into the graph. Other events can be modelled based on these two operations. Inserting a new edge e=(v,w) to a graph can be considered as decreasing the costs c(e) from ∞ to a value c. Deleting an edge e=(v,w) is similar to increase the costs of an edge to ∞. Deleting a vertex v can be done by deleting all edges that are incident to v. Inserting a vertex is trivial as long as no edge is connecting it with the rest of the graph. These connecting edges will be inserted using the inserting edge algorithm. Table 1 below described when it is required to delete, insert or modifying an edge or vertex.

Task	Example (Dynamic Events)
Deleting an edge	An edge might be blocked due to a disaster (ceiling collapse in buildings or road accidents) and can not be used any more.
Inserting an edge	In case of fire, more time is taken when rescuer uses ladder to rescue people from first floor, etc.
Modifying costs of an edge	More difficult to take this way out (due to smoke, etc.) or more easy to take this way out due to rescue team appearance.
Delete vertex	In case of fire, elevators can not be used.

Table 1: Description of dynamic events for routing algorithm.

In order to find the shortest way out of a building in case of an emergency from a given location or in traffic jams, the Single-Sink Shortest Path problem (SSSP) has to be solved for the dynamic graph G. Under the assumption that every edge e=(v,w) in the graph has positive costs c(e)>0, the problem can be solved by modified Dijkstra algorithm (adding the 3rd dimension, z into existing calculation). If the structure of the graph changes due to some event, the whole algorithm has to be run again even if the changes do not have any effect on the result. The Dijkstra algorithm can be considered as batch processing on a given input graph G and sink Vertex s. If the data input is changed, the algorithm as to be run again. In this chapter we propose an incremental approach to deal with the changes of the graph structure. Changes in the graph structure are usually local changes. Once the SSSP problem is solved for the given input (G, v), only a small part of the solution has be recalculated due to the event. This incremental approach is usually much more efficient. For a detailed analysis of complexity of this kind of incremental algorithms can be found in Ramalingam and Reps's (1996) research.

In this research, distance factor is added into the implemented algorithm used by Ivin et al. (2007) to enable the calculation of 3D shortest path in 3D dynamic network model.

$$\textit{Distance} = \sqrt{\Delta x^2 + \Delta y^2 + \Delta z^2}$$

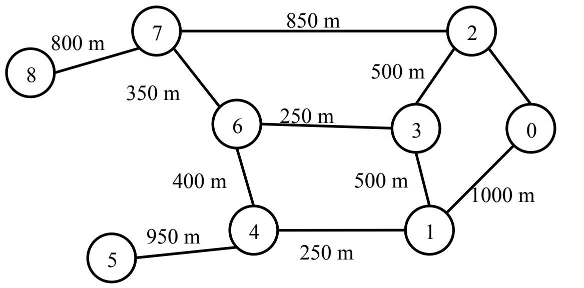
where *Distance* = the 3D distance of two nodes in real world.

Right after solving the dynamic event(s) routing algorithm, the shortest path distance is recalculated again with the implementation of proposed distance algorithm.

3.2 Incremental SSSP Algorithm

The input of the SSSP problem is a graph G (V, E), a cost function c: E → R+ and a sink vertex v. The shortest distance dist(w) from v for every vertex w $\in$ V should be computed. Dijkstra algorithm will solve the problem. However, G changes over time as discussed before. We are interest in an incremental algorithm that handles these changes without solving the SSSP problem for the whole graph again.

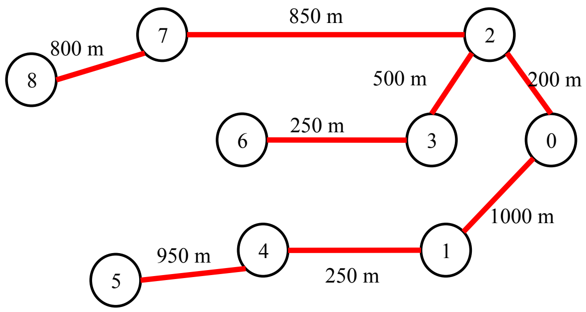
Under the assumption that the shortest path from v to w is unique for every vertex w $\in$ V, the resulting shortest paths build a spanning tree of G. This spanning tree ST consists of edge e $\in$ E that is used in at least on shortest path. Thus, e=(t,u) $\in$ ST if and only if dist(u) = dist(t) + c(e). In general, a shortest path might not be unique. It might be that a vertex could be reached on two different paths with the same costs. In this case, the resulting shortest paths do not build a spanning tree any more but a directed acyclic graph (DAG).



SSSP sink : 0

Vertex	0	1	2	3	4	5	6	7	8
Distance to Vertex 0	-	1000	200	700	1250	2100	950	1050	1850

Resulting shortest paths:

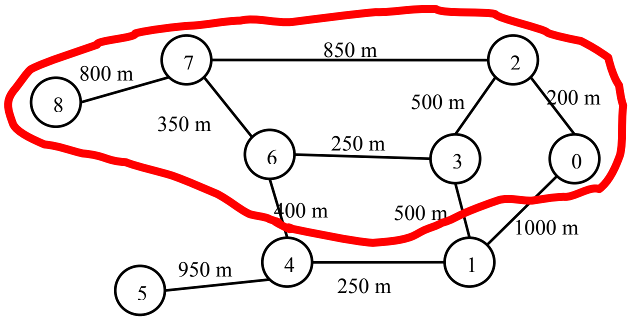


3.2.1 Deleting / increasing costs of an edge

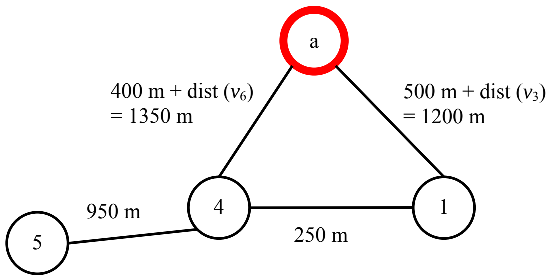
Increasing costs c(e) of an edge e=(t,u) or deleting this edge only has an effect to the solution of the SSSP problem if the edge e is part of an existing shortest path, thus $e \in ST$. In that case, all nodes that use the modified edge in their shortest path might be affected by this event. These nodes are the successors of node t in ST. Other nodes can not be affected because the costs for e are increasing. To recalculate the shortest paths form the sink to the affected vertices we simplify the graph as follows. All vertices that are not affected by the event will be treated as a single vertex. Each edge e=(x, y) of the affected vertices x to one vertex y that is not affected will get a new weight: cs'(e)= c(e) + dist(y). The SSSP problem is solved again with this simplified graph. For example;

Delete Edge e=(v0, v1)
Affected vertices: v1, v4, v5.
Not affected vertices: v0, v2, v3, v6, v7, v8.

Resulting graph :



Simplified graph to be solved:



3.2.2 Inserting / decreasing costs of an edge

Decreasing costs c(e) of an edge e=(t,u) or inserting a new edge. A vertex w is affected by this event if the new edge e enables a shorter path from v to w with distnew(w) < distold(w). The new edge e has to be part of this new shortest path. The length of the new shortest path is given by distnew (w) = dist(t) + c(e) + dist(u,w) where dist(u,w) is the length of the shortest path between u and w. The algorithm works similar to Dijkstra algorithm. In a batch implementation of Dijkstra algorithm all adjacent vertices vi of a vertex x are adjusted if dist(x) + c(x, vi) < dist(vi) where dist(vi) is the shortest distance so far. In the incremental version of the algorithm if edge e=(t,u) is inserted into G or the costs of the (already existing) edge is decreased, it has to be checked if dist(t) + c(t,u) < dist(u) or dist(u) + c(t,u) < dist(t). If this is the case, the algorithm continues with the affected vertex similar to the batch implementation of Dijkstra algorithm. Otherwise, the inserted edge does not change the shortest paths.

3.3 Implementation of the Incremental SSSP

This section describes the implementation of the incremental SSSP algorithm. The interface was developed using new 'classes' within Visual Basic 6.0 environment. Figure 1 illustrates the shortest path from A (vertex 1) to B (vertex 57) using modified Dijkstra algorithm. On the other hand, Figure 2 shows a new route derived from the implemented algorithm based on a dynamic event occurred in one of the edges along the original shortest path route from A to B. The dynamic event occurred at one edge that consists of vertex 18 as source and vertex 29 as destination. While the dotted line represents the original shortest path route.

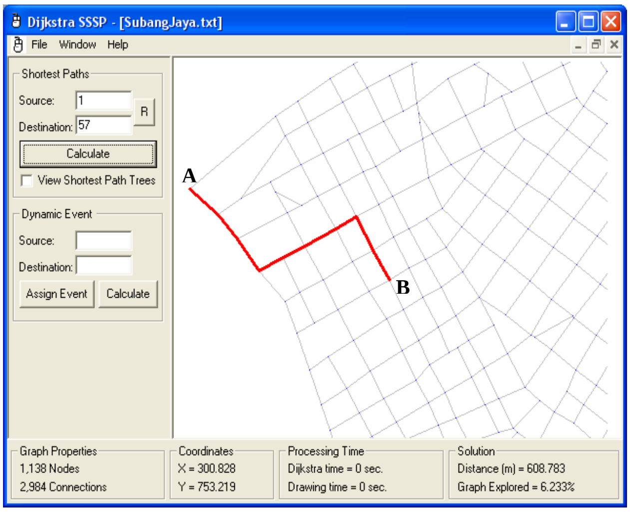


Figure 1. Shortest path from location A (vertex 1) to B (vertex 57) using the Dijkstra algorithm

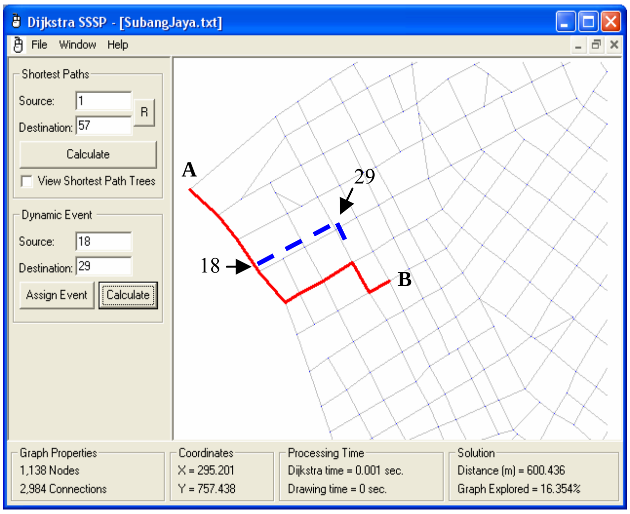


Figure 2. Shortest path from location A (vertex 1) to B (vertex 57) using Incremental SSSP Dijkstra algorithm after assigning a dynamic event at vertex 18 (source) and vertex 29 (destination). The dotted line was the original shortest path route from A to B before the dynamic event.

4 CONCLUSION

This study suggested a new spatial analysis function in 3D-GIS; the 3D network analysis for calculating shortest path routes in 3D network data model which supports dynamic changes information on the 3D data. This function not only can be used for finding shortest path on road networks or across terrain (surface) but also able to visualize significant landmarks to improved map presentation. Although this study is only focusing on shortest distance as the impedance factor for shortest path, other factors such as fastest (time) routes, minimum cost routes, and etc. can be forwarded as future study and development.

REFERENCES

Angela Altmaier and Thomas H. Kolbe, 2003. Applications and Solutions for Interoperable 3D Geo-Visualization [Electronic Version] from http://www.citygml.org/docs/Altmaier_und_Kolbe_PhoWo2003_englisch.pdf

C. Demetrescu and G. F. Italiano, 2004. A new approach to dynamic all pairs shortest paths. Journal of the Association for Computing Machinery (JACM), 51(6):968–992

C. Demetrescu and G. F. Italiano, 2005. Experimental analysis of dynamic all pairs shortest path algorithms. To appear in ACM Transactions on Algorithms. Special issue devoted to the best papers selected from the 15th Annual ACMSIAM Symposium on Discrete Algorithms (SODA'04)

Coors, V. and Flick S., 1998. Integrating Levels of Detail in a Web-based 3D GIS. 7th, ACM Workshop on Geographic Information Systems, Washington, USA, Nov. 6-7, 1998

Coors, V. and V. Jung, 1998. Using VRML as an Interface to the 3D data warehouse, in: Proceedings of VRML'98, New York (Monterey CA, Feb. 16-19), ACM Press, New York, 1998.

Döllner, Jürgen, Kolbe, Thomas H., Liecke, Falko, Sgouros, Takis and Teichmann, Karin, 2006. The Virtual 3D City Model Of Berlin - Managing, Integrating And Communicating Complex Urban Information. In proceedings of the 25th International Symposium on Urban Data Management (UDMS) 2006, Aalborg, Denmark, 15-17 May 2006.

Fischer, M. M. 2004. GIS and Network Analysis. [Electronic Version] from http://www.ersa.org/ersaconfs/ersa03/cdrom/papers/433.pdf

Fohl, P., Curtin, K. M., Goodchild, M. F. and Church, R. L., 1996. A non-planar, lane-based navigable data model for ITS. In M.J. Kraak and M. Molenaar (eds.) Proceedings, Seventh International Symposium on Spatial Data Handling, Delft, August 12-16, pp. 7B.17-7B.29

Goodchild, M.F., 1998. Geographic information systems and disaggregate transportation modeling. Geographical Systems 5(1–2):19–44

G. Ramalingam and T. Reps, 1996. An incremental algorithm for a generalization of the shortest path problem. Journal of Algorithms, 21:267–305

G. Ramalingam and T. Reps, 1996. On the Computational Complexity of Dynamic Graph Problems, Theoretical Computer Science, Vol 158 / 1&2. [Electronic Version] from http://citeseer.ist.psu.edu/cache/papers/cs/32474/http:zSzzSzwww.cs.wisc.eduzSzwpiszSzpaperszSztcs96a.pdf/ramalingam96computational.pdf

Ivin Amri Musliman, Alias Abdul Rahman and Volker Coors, 2006. 3D Navigation for 3D-GIS – Initial Requirements. Innovations in 3D Geo Information Systems, Springer: pp. 125-134

Ivin Amri Musliman, Alias Abdul Rahman and Volker Coors, 2007. Modeling Dynamic Weight for 3D Navigation Routing. Proceedings of Map Asia 2007, 14-16 August, Kuala Lumpur
Jayaperiyah Subramaniam, 2005. Integration of GIS and CAAD. Master Thesis, Stuttgart University of Applied Sciences, Germany.

Jürgen Döllner and Benjamin Hagedorn, 2007. Integrating Urban GIS, CAD and BIM data by service based virtual 3D city models. In proceedings of 26th International Symposium on Urban Data Management (UDMS) 2007, Stuttgart, Germany, 10-12 October 2007.

Khairul Hafiz Sharkawi and Alias Abdul Rahman 2007. Indoor and Outdoor Navigation System based on 3D Game Engine. In proceedings of Joint International Symposium & Exhibition on Geoinformation 2007 & International Symposium on GPS/GNSS 2007(ISG-GNSS2007), Persada Johor, Malaysia, 5-7 November 2007.

Kolbe, T. H. and Gröger, G., 2003. Towards Unified 3D City Models. [Electronic Version] from http://www.citygml.org/docs/CGIAV2003_Kolbe_Groeger.pdf

Köninger, A. and Bartel, S., 1998. 3D-GIS for Urban Purposes, Geoinformatica, 2(1), March 1998.

Kwan, M. P. & Lee, J. 2005. Emergency response after 9/11: The potential of real-time 3D GIS for quick emergency response in micro-spatial environments. Computers, Environment and Urban Systems 29: 93-113.

Lee, J., 2005. 3D GIS in Support of Disaster Management in Urban Areas [Electronic Version] from http://www.directionsmag.com/article.php?article_id=2049.

Liu Yuefeng, Zheng Jianghua, Yan Lei, Xu Yiqin, 2005. Study on the real time navigation data model for dynamic navigation. IGARSS '05 Proceedings. Geoscience and Remote Sensing Symposium, 2005. [Electronic Version] from http://ieeexplore.ieee.org/iel5/10226/32596/01525224.pdf

M. Abdul Mannan and Juergen Bogdahn, 2004. Virtual Environments in Planning Affairs. In 20th ISPRS Proceedings, Istanbul, Turkey. [Electronic Version] from http://www.isprs.org/istanbul2004/yf/papers/925.pdf

Michael Worboys and Matt Duckham, 2004. GIS: A Computing Perspective. Boca Raton, Florida: CRC Press LLC, 2nd edition.

Muhamad Uznir Ujang and Alias Abdul Rahman, 2007. Incorporating 3D Game Engine for 3D Network Analysis. In proceedings of Joint International Symposium & Exhibition on Geoinformation 2007 & International Symposium on GPS/GNSS 2007(ISG-GNSS2007), Persada Johor, Malaysia, 5-7 November 2007.

Nebiker, S., 2003. Support for Visualization and Animation in a Scalable 3D GIS Environment – Motivation, Concepts and Implementation. [Electronic Version] from http://www.photogrammetry.ethz.ch/tarasp_workshop/papers/nebiker.Pdf

Shi Pu and Sisi Zlatanova, 2005. Evacuation Route Calculation of Inner Buildings. Geo-information for Disaster Management, Springer: pp. 1143-1161

Thomas H. Kolbe, Gerhard Gröger and Lutz Plümer, 2005. CityGML – Interoperable Access to 3D City Models. In proceedings of International Symposium on Geo-information for Disaster Management (Gi4Dm), Delft, The Netherlands, 21-23 March 2005.

V. King, 1999. Fully dynamic algorithms for maintaining all-pairs shortest paths and transitive closure in digraphs. In Proc. 40th IEEE Symposium on Foundations of Computer Science (FOCS'99), pages 81–99

Wüst, T., Nebiker, S. and Landolt, R., 2003. Applying the 3D GIS DILAS to Archaeology and Cultural Heritage Projects – Requirements and First Results. [Electronic Version] from http://www.isprs.org/istanbul2004/comm5/papers/587.pdf

Zhan, F. B., & Noon, C. E. 1998. Shortest Path Algorithms: An Evaluation using Real Road Networks. Transportation Science 32(1): 65-73

Zhu Qing, Li Yuan and Tor Yam Khoon, 2006. 3D Dynamic Emergency Routing. GIM-International June 2006, Volume 20, Issue 6. [Electronic Version] from http://www.gim-international.com/issues/articles/id674-D_Dynamic_Emergency_Routing.html

Zipf, A. and Schilling, A., 2003. Generation of VRML City Models for Focus Based Tour Animations. Integration, Modeling and Presentation of Heterogeneous Geo-Data Sources. In: Proceedings of the 8th Int. Symp. on Web 3D Technology Web3D 2003, March 2003 in Saint Malo, France.

Zlatanova, S., 2000. PhD thesis, 3D GIS for Urban Development, Thesis ICGV, GrazUT, Austria and ITC, The Netherlands, ITC Dissertation Series No. 69, http://www.gdmc.nl/zlatanova/PhDthesis/pdf/content.html

Zlatanova S., Holweg D., 2004. 3D Geo-information in emergency response: a framework. Proceedings of the 4th International Symposium on Mobile Mapping Technology (MMT'2004), March 29-31, Kunming, China.

Zlatanova S., Holweg D., Coors V., 2004. Geometrical and Topological Models for Real-time GIS. Proceedings of UDMS 2004, 27-29 October, Chioggia, Italy.

Zlatanova S., Holweg D. and Stratakis M., 2005. Framework for Multi-Risk Emergency Responce.

Dueker, K. J., & Butler, J. A. 1997. GIS-T Enterprise Data Model with Suggested Implementation Choices. [Electronic Version] from http://www.upa.pdx.edu/CUS/publications/docs/PR101.pdf

GDF4.0 Manual. Available at http://www.ertico.com/en/links/gdf_-_geographic_data_files.htm or http://www.4dtechnologies.com/writegdf/GDF4_Specs.zip

NavTech. Available at http://www.navteq.com/sdalformat/

ACKNOWLEDGEMENT

I would like to convey my deepest acknowledgment firstly to Assoc. Prof. Dr. Alias Abdul Rahman and Prof. Dr. Volker Coors for the continuous support, guidance and motivation throughout the research. Secondly, my sincere appreciation to the Ministry of Science, Technology and Innovations (MOSTI) for providing us the research grant Vot Number 79198, entitle 3D Navigation for 3D-GIS which covers my expenses for this conference. Finally, Universiti Teknologi Malaysia (UTM) for maintaining well services for our research facilities needs.