Optimizing Durkins Propagation Model Based on TIN Terrain Structures

Leonid Djinevski[1], Sonja Stojanova[2], and Dimitar Trajanov[2]

[1] FON University,
Av. Vojvodina, 1000 Skopje, Macedonia,
leonid.djinevski@fon.edu.mk
[2] Ss. Cyril and Methodious University,
Rugjer Boshkovikj 16, 1000 Skopje, Macedonia
stojsonja@gmail.com, dimitar.trajanov@finki.ukim.mk

Abstract. In order to bring wireless ad hoc networks simulation a step closer to real-life scenarios, 3D terrains have to be taken into account. Since 3D terrain involves larger amounts of data, network simulations with heavy traffic load, requires compute intensive calculations. In this paper we evaluate the usage of efficient point location for network simulation in 3D terrains, in order to increase the performance of the overall simulation. Our experimental results show a reasonable speedup using the jump-and-walk point location algorithm when computing the propagation between two wireless nodes, as well as for the overall performance increase for a complete simulation scenario.

Keywords: Network Simulators, TIN, Point Location, Durkins Radio Propagation, Parallel.

1 Introduction

When researching new scenarios and protocols in a controlled and reproducible environment, network simulators are the main tools that allow the user to represent various topologies, simulate network traffic using different protocols, visualize the network and measure the network performances.

Many network simulators have been developed as open-source tools by the academia and proprietary tools by the industry. NS-2 network simulator stands out as the most popular and most widely adopted, thus it is considered as de facto standard regarding network simulation. Simulation for 802.11 ad hoc networks with mobile nodes, special routing protocol and fully specified network traffic is supported by the NS-2 network simulator.

Most of the commonly used propagation models when assessing performance of ad hoc networks, take into account the following mechanisms of reflection, diffraction, scattering, penetration, absorption, guided wave and atmospheric effects [10]. However, the available models in NS-2, do not take the terrain profiles in consideration, thus the obtained results for the received signal strength from a transmitter are not close to real life scenarios.

Since the communication between nodes in wireless ad hoc networks is usually carried out in irregular terrains, the terrain profile should be taken into account. Authors in [13] layout such implementation called the Durkins propagation model, which is an extension of the NS-2 simulator [1], where the terrain profile is represented by Digital Elevation Model (DEM) [21] data structure.

An improvement of the extension of the Durkins propagation model is presented in [27], where the implementation uses Triangular Irregular Networks (TIN) [24] based terrains. This tool allows the simulation modeler to conduct more realistic simulation scenarios, and analyze the way the terrain profile affects the ad hoc network performances. Although the implementation in [27] is very useful, it does not scale well, thus starting a simulation for medium to large networks, results in few hours up to days, even weeks of execution time which is unsuitable for investigating protocols. Additional optimization was proposed by authors in [12] where they have speed up the point location of Durkins propagation model, by parallel execution using GPU devices.

In this paper, we present an analysis of three main efficient access algorithms implemented for point location in the Durkins propagation model based on TIN terrain profile. Additionally we investigate how the efficient point location method affects the overall execution time of a given simulation, for different detail levels of the terrain.

The obtained results help the simulation modeler to choose the appropriate point location algorithm for a terrain profile with a given detail level, in order to achieve faster simulations.

The remainder of this paper is organized as follows: In Section 2 we present a small introduction to mobile ad-hoc network (MANETs) and the Durkins terrain aware radio propagation model for the NS-2 simulator. We present the DEM and TIN data structure that describe a given terrain profile in Section 3. Theoretical analysis of point location methods are presented in Section 4, followed by a short presentation of the applied testing methodology in Section 5. The obtained results are analyzed in Section 6. We conclude in Section 7.

2 MANETs and the Durkin's propagation model

Communication among people on the move has evolved remarkably during the last decade. One of the possibilities for such mobility is the mobile radio communications industry growth by orders of magnitude, thus making portable radio equipment smaller, cheaper and more reliable [17]. The large scale deployment of affordable, easy-to-use radio communication networks has created a trend of a demand for even greater freedom in the way people establish and use the wireless communication networks [9].

One of the consequences to this ever present demand is the rising popularity of the ad hoc networks. A MANET does not rely on any infrastructure, thus, it can be established anywhere on the fly [26]. Wireless mobile nodes communicate directly with the absence of base station or an access point. Therefore, nodes establish the network environment by self organization in a highly decentralized manner. In order to achieve this goal every node has to support the so-called multihop paths. The multihop path concept is introduced to allow two distant nodes to communicate by the means of the intermediate nodes to graciously forward the packets to the next node that is closer to the destination. This is controlled by a special ad hoc routing protocol [8] that is concerned with discovery, maintenance and proper use of the multihop paths.

The independence of existing infrastructure, as well as the ability to be created instantly, that is, on demand, has made the ad hoc networks a very convenient and irreplaceable tool for many on-the-go situations like: rescue teams on crash sites, vehicle to vehicle networks, lumber activities, portable headquarters, late notice business meetings, military missions, and so on. Of course, every one of these applications demands a certain quality of service from the ad hoc network and usually the most relevant issue are the network performances in terms of end-to-end throughput. However, the tradeoff of having no infrastructure and no centralized manner of functioning has influenced the ad hoc networks performances greatly on many aspects.

3 Terrain data structures

Many applications utilize Geographical Information Systems (GIS) technology for which geographical referenced information of sea bottom or physical land surface is required [28][18][6][25][7]. This technology provides representation of terrain elevation data using two broad methods: raster images or vectors. The raster data format is consisted of rows and columns of cell, wherein each cell a numerical value describes an attribute of the surface. Therefore each cell is a single pixel, and the grid of all the cells with its width and length represent the resolution of a terrain. The most widely spread raster data format is described by the Digital Elevation Model (DEM) standard [23][5][3][4]. On the other hand, the vector data format uses geometrical units like points, lines and polygons to represent objects. Hence, compared to the raster data format, much less data is needed to represent the same object, resulting in smaller files for storing the data.

The most popular vector data format for representing elevation is the Triangular Irregular Networks (TIN) [2][29][20]. The basic building block of the TIN data structure is a record that is consisted of points that associate 3D coordinates, which are interconnected by lines, distributed in such way to form triangles, where no triangles are overlapping each other. Both of the data structures (DEM and TIN) for the same terrain are visualized on Figure 1.

The raster data format is more suitable for modeling surfaces, although the vector data format can produce much more precise representation of a given surface with the same amount of data. Beside the accuracy of the surface, the raster data format regarding hill peaks and ridges is reliable as much as the network resolution allows it, while for the TIN data format hill peaks and ridges are very precise.

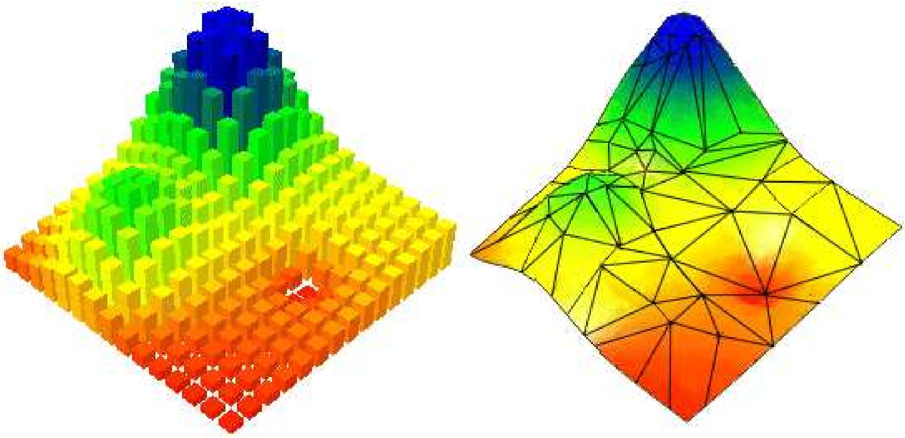

Fig. 1. Terrain representation using DEM data (left), and TIN data (right)

3.1 Optimal access to the network

Since the TIN networks are vector structures, there is no direct way to obtain an elevation of a point on the terrain. The most simple method for point location is to traverse a TIN network by visiting all triangles only once [15][14][16]. However, this method is ineffective when dealing with large-scale TIN networks. There are several point location strategies for the TIN model [19]. Some of these strategies include: hierarchical data structures, spatial indexing structures, quad trees, etc. Although, these solutions are very effective, the implementation effort for the data structures they use, can be nontrivial, so in practice more simple methods like walking algorithms are used. These algorithms solve the point location problem by traversing the triangles of the TIN, starting from a random triangle, until the triangle containing the query point is reached.

4 Theoretical Analysis

In this section we analyse few of the most popular and widely used algorithms for point location in TIN terrains: quad trees, jump-and-walk, and windowing.

4.1 Quad trees

As the name of the algorithm implies, quad trees data structure is used, which is a regular search tree of degree 4 as presented in Fig. 2. The root square is decomposed in 4 subsquares recursevly in the same fashion as the quad tree expands. The decomposition continues until the subsquare data is simple enought. The algorithm is consisted of two steps:

– Traverse down the tree unti the data structure terminates with a triangle.,

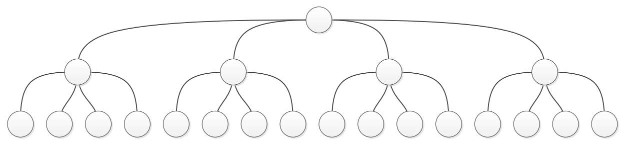

Fig. 2. Quad-tree algorithm

– Traverse the subsquare structure in order to find the triangle that contains the point that is queried.

4.2 The jump-and-walk algorithm

The jump-and-walk [11] is one of the most competitive walking algorithms in which the traversal from the starting to the final triangle is determined from the line segment (p,q) where p is the starting and q is the query point (Fig. 3 on the left). The jump-and-walk algorithm consists of three main stages:

– Obtaining m random, but uniformly distributed starting points in the XY-plane, where m is around $n^{1/3}$,
– Determine the index i , for $i = 1..m$, of the starting point such that the Euclidian distance between p_i and q is minimal,
– Traverse the triangles that intersect the line segment (p_i, q) until the triangle that contains the query point is located,

The additional data structure required is for storing the index of the neighboring triangles for a given triangle in the TIN terrain.

4.3 The windowing algorithm

Windowing as an approach for the point location where the events take place in given squarearea, for example silnal propagation between two (Transiver-Receiver) nodes. The windowing algorithm is as folllows:

– in this step the position of the window needs to be located,
– and all triangles that intersect with the window need to be located.

5 Testing Methodology

All of our tests were performed on the same hardware infrastructure: (Intel i7 920 CPU at 2.67GHz, with 12GB RAM at 1333MHz, GPU NVIDIA Tesla C2070), with Linux operating system Ubuntu 10.10. The implementations were compiled with the NVIDIA compiler nvcc from the CUDA 4.2 toolkit.

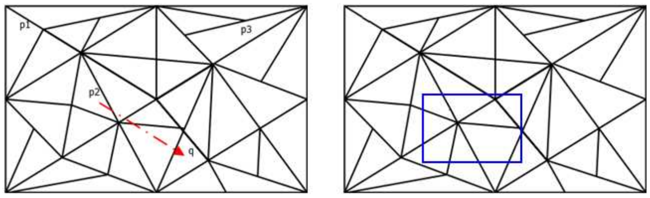

Fig. 3. Jump-and-walk algorithm (left), and windowing algorithm (right)

The testing methodology is based on 2 experiments which show inverse execution time dependence of problem size (number of triangles in the TIN terrain) for the jump-and-walk approach and overall performance increase in a simulation scenario.

Experiment 1 determines the inverse execution time compared to different terrain requirements. We increase the terrain size and measure the speed for the executions of jump-and-walk approach, standard sequential approach of traversing all triangles and the parallel implementation of the standard approach. Our hypothesis to be confirmed experimentally is to achieve the highest speed for the jump-and-walk approach.

In Experiment 2 we implement the jump-and-walk approach in given NS-2 simulations, by varying different node mobility (nodes moving with speed of 1, 2 and 5 m/s) and different traffic load (from 0.1 to 7 Mbps). We use a TIN terrain with dimensions of 1 million square meters, and highest relative point of 200m. We have 100 nodes that are uniformly dispersed in the simulation area. The node transmission range is set to the standard 250m given by the use of the IEEE 802.11b standard wireless equipment. The antenna height is set to 1.5m and it has no relative offset against the wireless node. For route discovery and path set up we adopt the AODV protocol [22]. The simulation time is set to 1.5 hours as to the average battery life of a notebook. During the mobile simulations, the nodes are moving according to the random direction model in the terrain boundaries. The average node speed is varied from 0 (static nodes) to 1, 2 or 5 m/s with standard deviation of 0.1 m/s. We measure the average execution time for all network traffic scenarios by varying the network load from 0.1 to 7 Mbps using UDP data packets with 1 KB size. Our hypothesis to be confirmed experimentally is to achieve higher speedup for the average execution time using the jump-and-walk approach, compared to the average execution time without the jump-and-walk approach.

6 Results of the Experiments

This section presents the results that show performance increase in wireless simulation using the jump-and-walk, quad trees, and windowing for point location algorithms.

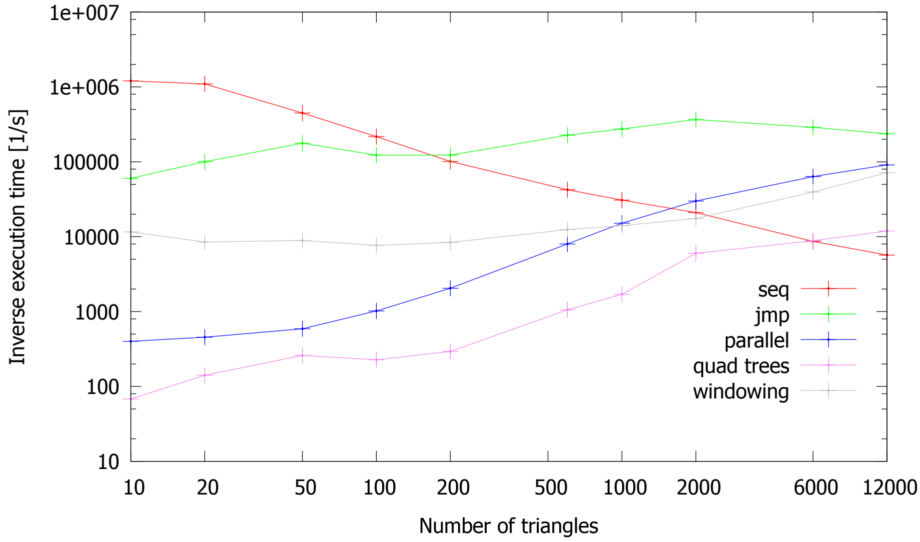

Fig. 4. Inverse execution time of different point location algorithms

6.1 Experiment 1

Figure 4 depicts the inverse execution time for different terrain requirements, where seq stands for the sequential implementation, parallel stands for the parallel implementation, jmp stands for jump-and-walk implementation, quad trees stands for the quad trees algorithm, and windowing for the windowing algorithm. Thus, it is easy to notice that the jump-and-walk implementation is the fastest for larger number of triangles. For smaller number of triangles the parallel implementation does not utilize the GPU resources, therefore there is slow down and the speed is significantly lower than the sequential implementation. Only for terrains with more than 1000 triangles, the parallel implementation expresses higher speed than the sequential implementation. The speed of the sequential implementation is the best for small number of triangles. However, by increasing the number of triangles, the speed of the sequential implementation is getting worse.

6.2 Experiment 2

In our second experiment we are evaluating the speedup of jump-and-walk implementation and the standard parallel implementation for compared to given terrain requirements, for average node velocity of 1, 2 or 5 m/s. The results are depicted in Figure 5 where the dotted lines represent the standard parallel implementation, and the solid lines represent the jump-and-walk implementation. For jump-and-walk implementation the results confirm our expectation to achieve higher performance.

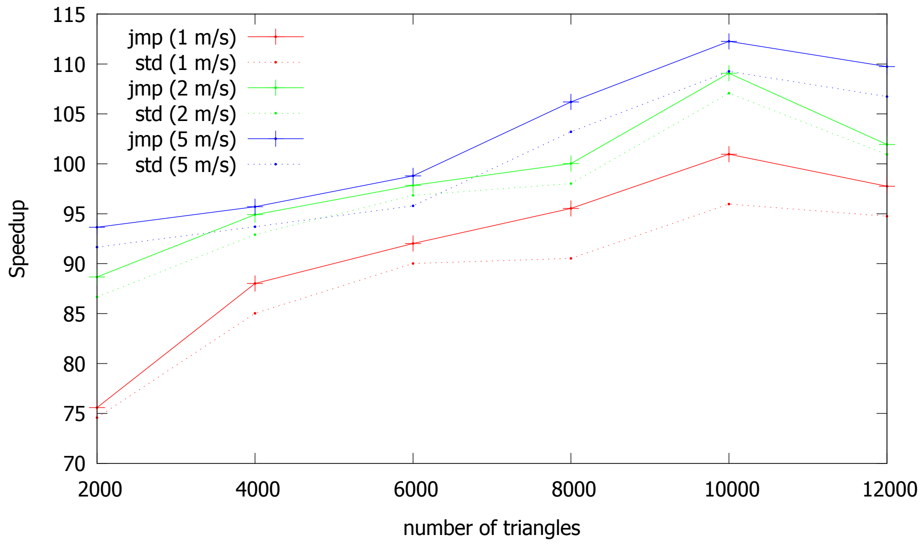

Fig. 5. Overall NS-2 network simulation speedup using different point location algorithms

7 Conclusion and Future Work

Implementations of three efficient access algorithms are presented in TIN terrain representation, as an alternative to DEM terrain representation discussed in previous work [13].

There are many proprietary and open-source network simulators that are available. However, NS-2 Simulator is the most widely used and adopted by the research community as de facto standard regarding network simulation. For NS-2 Simulator we developed our efficient access implementations.

We performed two experiment that evaluate the performance of the optimization using the jump-and-walk approach and overall performance increase in a NS-2 simulation.

The experiment 1 confirms that the jump-and-walk algorithm achieves best speed for terrains with more than 200 triangles. Since terrains with less than 1000 triangles do not describe the terrain well, the results point out that jump-and-walk algorithm is the fastest. Additionally, the experiment 2 confirmed that the effective point location implementation for the jump-and-walk algorithm achieves performance increase for overall network simulation scenarios of terrain aware radio propagation model.

Although there are theoretically more efficient algorithms for point location, we have chosen the jump-and-walk algorithm. In order the jump-and-walk algorithm to be effective, the assumption that the starting points are uniformly distributed across the terrain needs to be fulfilled, which is the case with the nodes in our simulation scenario.

Additionally, the jump-and-walk algorithm requires a very simple data structure, therefore there are very few requirements needed to modify the standard implementation of the terrain aware extension for the Durkins propagation model in NS-2 Simulation.

8 Acknowlegement

We gratefully acknowledge the Nvidia Corporation for support in graphics hardware as part of the CUDA Teaching Center program.

References

1. ns-2, network simulator (1989)
2. Unit 39-the tin model (2011), `http://www.geog.ubc.ca/courses/klink/gis.notes/ncgia/u39.html`
3. U.s. geological survey national mapping division: Part 1 general, standards for digital elevation models (2011), `http://nationalmap.gov/standards/pdf/1DEM0897.PDF`
4. U.s. geological survey national mapping division: Part 2 general, standards for digital elevation models (2011), `http://nationalmap.gov/standards/pdf/2DEM0198.PDF`
5. Usgs dem (2011), `http://edc.usgs.gov/guides/usgs_dem_supplement.html`
6. What is gis? (June 2011), `http://www.gis.com/content/what-gis/`
7. Abdul-Rahman, A., Pilouk, M., et al.: Spatial data modelling for 3d gis (2008)
8. BR, A.K., Reddy, L.C., Hiremath, P.S.: Performance comparison of wireless mobile ad-hoc network routing protocols. IJCSNS 8(6), 337 (2008)
9. Brown, J.S., Duguid, P.: Social life of information. Harvard Business Press (2002)
10. Cichon, D.J., Kurner, T.: Propagation prediction models. COST 231 Final Rep (1995)
11. Devroye, L., Mücke, E.P., Zhu, B.: A note on point location in delaunay triangulations of random points. Algorithmica 22(4), 477–482 (1998)
12. Djinevski, L., Filiposka, S., Trajanov, D., Mishkovski, I.: Accelerating wireless network simulation in 3d terrain using gpus. Tech. Rep. SoCD:16-11, University Ss Cyril and Methodius, Skopje, Macedonia, Faculty of Information Sciences and Computer Engineering (June 2012)
13. Filiposka, S., Trajanov, D.: Terrain-aware three-dimensional radio-propagation model extension for ns-2. Simulation 87(1-2), 7–23 (2011)
14. Gold, C.M., Charters, T., Ramsden, J.: Automated contour mapping using triangular element data structures and an interpolant over each irregular triangular domain. ACM SIGGRAPH Computer Graphics 11(2), 170–175 (1977)

15. Gold, C., Cormack, S.: Spatially ordered networks and topographic reconstructions. International Journal of Geographical Information System 1(2), 137–148 (1987)
16. Gold, C., Maydell, U.: Triangulation and spatial ordering in computer cartography. In: Proc. Canad. Cartographic Association Annual Meeting. pp. 69–81 (1978)
17. Hekmat, R.: Ad-hoc networks: fundamental properties and network topologies. Springer (2006)
18. Kennedy, M.: Introducing Geographic Information Systems with ArcGIS: A Workbook Approach to Learning GIS. Wiley (2009)
19. van Kreveld, M.: Digital elevation models and tin algorithms. In: Algorithmic foundations of geographic information systems, pp. 37–78. Springer (1997)
20. Li, Y., Yang, L.: Based on delaunay triangulation dem of terrain model. Computer and Information Science 2(2), P137 (2009)
21. Maune, D.F.: Digital elevation model technologies and applications: The DEM users manual. Asprs Pubns (2007)
22. Meltzer, R., Zeng, C.: Micro-benchmarking the c2070 (Jan 2013), `http://people.seas.harvard.edu/~zeng/microbenchmarking/`
23. Peckham, R.J., Jordan, G.: Digital terrain modelling: development and applications in a policy support environment, vol. 10. Springer (2007)
24. Rognant, L., Goze, S., Planes, J., Chassery, J.: Triangulated digital elevation model: definition of a new representation. International Archives of Photogrammetry and Remote Sensing 32, 494–500 (1998)
25. Shi, W.: Principles of modeling uncertainties in spatial data and spatial analyses. CRC Press (2010)
26. Tonguz, O.K., Ferrari, G.: Ad Hoc Wireless Networks: A Communication-Theoretic Perspective. John Wiley & Sons, Ltd (2006)
27. Vuckovik, M., Trajanov, D., Filiposka, S.: Durkins propagation model based on triangular irregular network terrain. In: ICT Innovations 2010, pp. 333–341. Springer (2011)
28. Wise, S.: GIS basics. CRC Press (2002)
29. Zeiler, M.: Modeling our world, environmental systems research institute. Inc. Redlands, California (1999)