MODIFIED A* ALGORITHM IMPLEMENTATION IN THE ROUTING OPTIMIZED FOR USE IN GEOSPATIAL INFORMATION SYSTEMS

S. M. Ayazi [a, *], M. F. Mashhorroudi [b], M. Ghorbani M. [c]

[a] Head of Data Processing in NCC- NE General Administration, PhD. Student at Department of Surveying and Geomatics Engineering, University of Tehran, Iran - Info@Ayazi.ir
[b] MSc. Student of GIS/Remote Sensing, Expert of Computer- NE General Administration, Mashhad, Iran
Mf.mash@Gmail.com
[c] MSc. GIS/Remote Sensing, Expert of RS in NCC- NE General Administration, Mashhad, Iran
MaryamGhorbani15@Yahoo.com

KEY WORDS: GIS, Routing Optimization, Graph Theory, A* Algorithms, Modelling

ABSTRACT:

Among the main issues in the theory of geometric grids on spatial information systems, is the problem of finding the shortest path routing between two points. In this paper tried to using the graph theory and A* algorithms in transport management, the optimal method to find the shortest path with shortest time condition to be reviewed. In order to construct a graph that consists of a network of pathways and modelling of physical and phasing area, the shortest path routes, elected with the use of the algorithm is modified A*.At of the proposed method node selection Examining angle nodes the desired destination node and the next node is done. The advantage of this method is that due to the elimination of some routes, time of route calculation is reduced.

1. INTRODUCTION

With the growth of cities, especially in big cities, need to an effective plan to find ways to avoid wasting money and time. Among the issues that will be helped in this context is the routing and find the shortest routing path. To solve this problem, algorithms like Dijkstra's algorithm, A*, Bellman-Ford, Floyd Varshal, Johnson and artificial intelligence techniques, etc. provided. Dijkstra method to find the shortest path, all vertices are examined which can be time-consuming and more expensive. Dijkstra's algorithm is a graph traversal algorithms for the shortest path problem for weighted graphs, solve that do not have the edge with a negative weight. ArcGIS software as one of the most popular GIS software uses of the algorithm for routing. [YANG Yu-Jun, 2006]

There is a graph consists of a set of nodes and edges in the routing problem. Each edge has weight that it is proportional to its features. Set consists of two components of a graph G (V, E) that V is a finite set and non-empty and E is defined as a binary relation on V. [Boundy, J.A. and U.S.R. Murty, 1999]

V is the set of vertices in a graph and E consist the edges of the graph.

The purpose of routing find a path that can meet the needs of the problem. [J. Saberian, M Hamrahm,88]

2. A*ALGORITHM

Dijkstra's algorithm is a graph traversal algorithms by Dutch scientists, computer science, Dijkstra was introduced in 1959.

In this algorithm, the shortest path from the start point to the weighted graphs with no negative weight edges resolved, and finally by creating the shortest path tree, the shortest path from the start point to all vertices gives a graph calculate . Also this algorithm to find the shortest path from start to destination vertex thus use it while running the algorithm find once the shortest path from start to destination, the algorithm stopped. Dijkstra's algorithm in its simplest implementation, the data is stored in an array so that the minimum value of d for the vertices outside the set S (a set of vertices whose shortest path from the start to the specified weight) with a linear algorithm finds.

In this case time complexity will be as equation 1 that if used in the vicinity of the listed priority queue, the time complexity of the algorithm is heals as the equation 2.

$$o(|v|^2 + |E|) \qquad (1)$$

$$o(|v| + |E|)\ \log|v| \qquad (2)$$

One problem of this algorithm is to find the shortest path algorithm, all vertices in all directions during the long process of review and then reaches the destination node.

A* algorithm is computer algorithm that used widely in the navigation graph and finding a path between two nodes, this algorithm is in fact a distribution of Dijkstra's algorithm which uses innovative algorithms to achieve better performance compared to time. Due to the rapid performance and precision of this algorithm, using this algorithm in various branches of science depends expanded.

This algorithm uses the idea of the first algorithm that is best and the shortest path between two nodes that the given stat and destination with other nodes find. This method with combining a function g(n) and h(n) is evaluated nodes. (Equation 3)

$$f(n) = g(n) + h(n) \qquad (3)$$

Where g(n) = the cost of reaching to node n.
h(n) = cost of reaching the node n to goal node.

Unlike Dijkstra method, computation time, isn't the main drawback of A* algorithm but since this algorithm keeps all nodes in memory products, usually take more than a memory with a little time. [Dijkstra, E. W,1959]

[*] Corresponding author

doi:10.5194/isprsarchives-XL-2-W3-69-2014

A* search method is a combination of Uniform cost search and a greedy search method. Method A* fast greedy technique to achieve results and optimized uniform cost method to find the answer together and devoted their search for purpose.

Contrary Dijkstra's algorithm, is that all vertices in all respects to check, A* algorithm only points on the graph where the vertices are located the start and destination examine. (Equation4)

$$|h(x) - h^*(x)| \leq o\big(\log h^*(x)\big) \qquad (4)$$

Where h* (x) = the actual cost of reaching the node n to the goal.

3. A* ALGORITHM IMPLEMENTATION AND OPTIMIZATION OF THE METHOD DIRECTION

The A* algorithm be the function $f(n) = g(n) + h(n)$ calculate. h(n) estimates the distance to the destination, which can be calculated from equation 5

$$h\ (n) = \sqrt{(\mathbf{w_x} - \mathbf{r_x})^2 + (\mathbf{w_y} - \mathbf{r_y})^2} \qquad (5)$$

By calculating the Euclidean distance between the current node and the destination node, we can calculate an estimation distance that always is less than or equal to the actual distance. [Chen Hong-ying , Xiao Ting, Wang Tao, He Jin-yi, 2012]

g(n) is cost function travelled to the edge and over the edge plus the cost prior to the current edge. This amount represents the cost for each node is traversed from the start node to the current node.

f(n) represents the cost of the current node travelled plus estimated cost from the current node to the destination node. The node which has the minimum value f (n) is much better candidate to achieve the target.

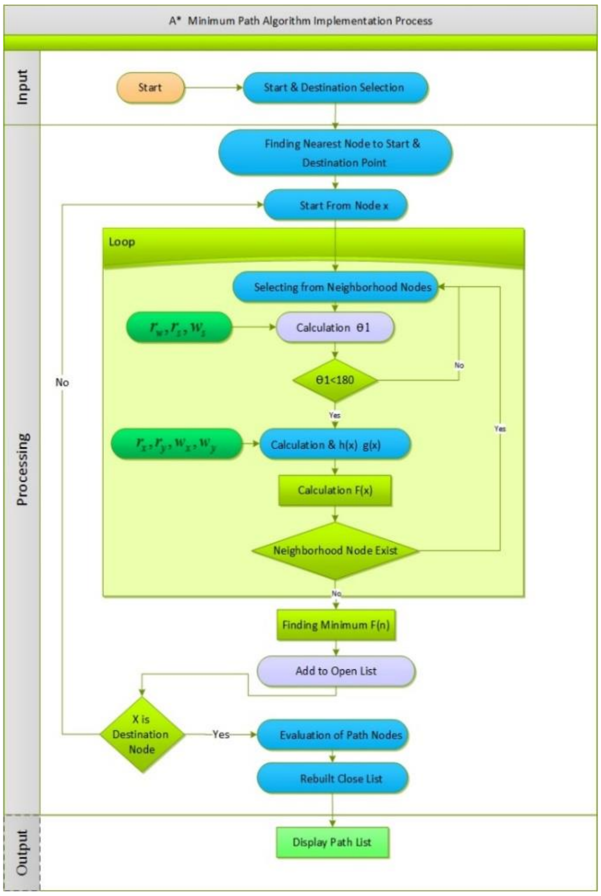


Figure 2: Implementation steps of A* algorithm

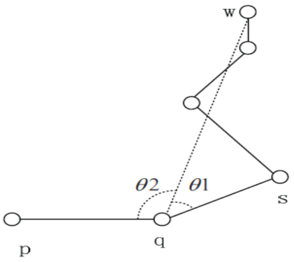


Figure 1: Angle and distance in

Each node in the lowest f (n) is chosen and developed and then is compared with a list of the current node to the specified node is still better candidate or not. Otherwise, the better node than the node list is selected.

As we move into the direction of the destination, not the destination path in the opposite direction, causing more of h(n) of the current node are, therefore, most likely not a good candidate to be the best route.

It can be use from methods that removed the destination paths that are far from the navigation list. The proposed method has been implemented in this study, the next node to the destination node can be done by calculating the angle. Experience shows that the best measure of the angle of approach, is angle of 180 degrees. In other words, the nodes in the opposite direction to the destination node is not appropriate to continue the path.

To obtain the angle less than 180 degrees angle for navigation edges have been using the following equation 6:

$$\mathbf{\cos\theta1} = \frac{|\mathbf{r_w}|^2 + |\mathbf{r_s}|^2 - |\mathbf{w_s}|^2}{2|\mathbf{r_w}|^{|\mathbf{r_s}|}} \qquad (6)$$

Flowchart in Figure 2 is shown implementation steps A* algorithm.

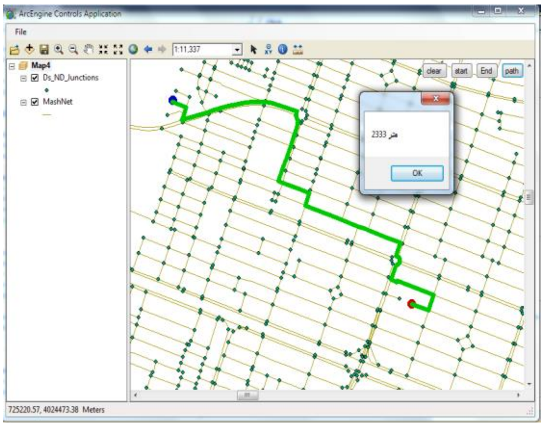
4. MODELING USING THE A * ALGORITHM IS APPLIED IN THE HOLY CITY OF MASHHAD (CASE STUDY)

The introduction of the A* algorithm, VB.Net 2012 IDE engine ArcEngine10.1 map is applied.

Before using this algorithm in the network layer streets and intersections is prepared in the ArcGIS environment and on the edges were applied the networking that the nodes and edges create.

We used the network for faster navigation in adjacent nodes.

Figure 3 is an example of running the program on the part of the city of Mashhad in the show.



2333 متر

725220.57, 4024473.38 Meters

Figure 3: Examples of performances on the streets of

5. ASSESSMENT METHODS

For evaluate the algorithms A* and Dykstra, two different areas were studied in Mashhad.

Evaluation of the two methods was examined in terms of computing time in milliseconds on a computer with Corei3 processor and the number of nodes traced by the algorithm. There are results in Tables 1-1 and 1-2 with two different route.

Algorithm	Processing time (ms)	number of Traced nodes	shortest path (m)
Modified A*	115	37	941
Dijkstra	1215	537	815

Table 1. Result of assessment in first area

Algorithm	Processing time (ms)	number of Traced nodes	shortest path (m)
Modified A*	411	143	2597
Dijkstra	12804	4543	2222

Table 2.Result of assessment in second area

Figures 4 shows, path and investigation nodes by the modified A* algorithm and Dykstra.

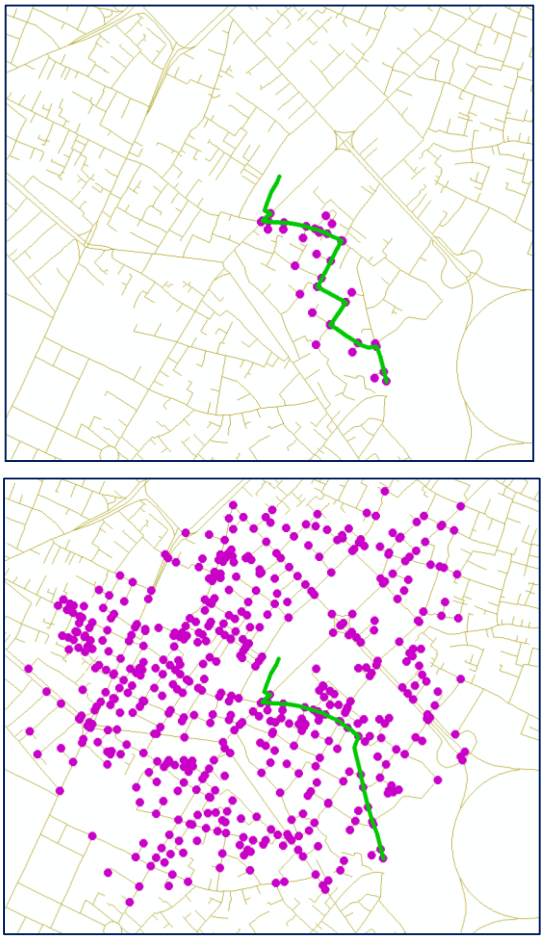


Figure 4: path and investigation nodes by the modified A* algorithm(up) and Dykstra (down).

As seen in tables 1 and 2, the time to answer Dijkstra algorithm vs. the modified A* algorithm is proportional to the extent of the selected area that increased exponentially with the number of nodes that need to be investigated.

In the modified A* algorithm, when distance is approximately three times, the number of examined nodes is almost four times, while in Dijkstra algorithm will be approximately 8.5 times.

6. CONCLUSIONS AND RECOMMENDATIONS

Dijkstra's algorithm calculate the shortest distance from node to node that is not suitable for systems that have a large number of nodes. For a larger cities, there are many of nodes, which Dijkstra algorithm requires a lot of time and cost.

However in term of shortest path, Dijkstra algorithm give better results than modified A* algorithm but this improvement just is approximately 10 to 15 percent while it has less important versus of the processing time, especially in the wide area.

A* algorithm is a heuristic algorithm to search for a new route uses heuristic information. The lower nodes and therefore less time is required.

For improve the effectiveness of method A* is used from the restriction direction, so with it eliminated nodes that are moving in the opposite direction and further the scrolling speed rises.

In future studies, determine the angular size of the proposed approach and method and even floating point values for the various routes.

7. REFERENCE

Boundy, J.A. and U.S.R. Murty, “Graph Theory with Applications”. ISBN: 964-6761-57-7, 1999.

Dijkstra, E. W. "A note on two problems in connexion with graphs". Numerische Mathematik 1: 269–271, 1959.

YANG Yu-Jun, “The Reasearch and Implementation ofshortest path in GIS”. Thesis of masters of ShiChuan university, 2006.4

ZHANG Lian-Peng,LIU Guo-Lin, JIANG Tao. “The knowledge-based method of GIS shortest path searching”. Science of Surveying and Mapping.2003,9,Vol28(3):27-3

Chen Hong-ying , Xiao Ting, Wang Tao, He Jin-yi." The Research and Implementation of Three Stages Traffic Stations Intelligent Monitor Systems Based on GIS", 2012 International Conference on Solid State Devices and Materials Science, Physics Procedia 25 (2012) 879 – 885

J. Saberian, M Hamrah, improve the performance of routing algorithms in an urban network, 88 Geomatics