A Model for Least Distance Protocol in MANET Using Artificial Intelligence Search Techniques

Khondker Shajadul Hasan¹, Radia Abdul Wahab², Munawwar Mahmud Sohul²

¹School of Computer Science, University of Oklahoma
110 W. Boyd St., Norman, OK 73019, USA
shajadal@ou.edu
²Department of Computer Science & Engineering, North South University
Bashundhara, Dhaka, Bangladesh
radiaraian@northsouth.edu, mmsohul@northsouth.edu

Abstract - As mobile adhoc networks operate with nodes carrying small amount of energy, reducing energy consumption is a major issue in adhoc networks. The paper discusses a protocol along with its implementation which incorporates Artificial intelligence search techniques, namely A* search and Dijkstra's algorithm to reduce the distance traversed by both the data and the control information. The transmission path is determined by the means of A* search, which gives the shortest path possible. This greatly reduces the distance traversed. This paper also focuses on the performance of this protocol compared with swarm intelligence which is another artificial intelligence protocol implemented by MANET (mobile adhoc network). A collection of proposed transmission protocols' characteristics has been presented in this paper. An extensive empirical study has been carried out to validate the protocol's performance.

Index Term-- Mobile Adhoc Networks, Multi-hopping, Grid system, A* search, Dijkstra’s Algorithm, Swarm intelligence, Path cost.

1. INTRODUCTION

Adhoc networks consist of nodes that communicate with each other. However, a node has limited CPU capacity, storage capacity, bandwidth, and battery power. Therefore, the chosen protocol should take all these into consideration. In infrastructure based network, the wireless network uses a specific piece of hardware which is called the base station or the central unit. The base station is connected to a wired network. As a result the wireless devices communicate with these base stations and these base stations communicate with the rest of the network. In a situation where there is no such base station or even if there is one, it is out of range; mobile nodes can still form a fully connected wireless network by operating in adhoc mode [1]. An adhoc network is therefore a wireless network formed by wireless nodes without any help of infrastructure. In such a network the wireless nodes can dynamically communicate with one another.

This paper designs such a protocol and then explains the characteristics of that protocol. Later in the paper it shows how the performance would vary when different numbers of nodes are put into consideration. An adhoc network is one that does not have an infrastructure. Each of the nodes work together to form the network. The main characteristics of MANET are that nodes of the network can communicate wirelessly, there is no fixed centralized infrastructure and that nodes are not required to remain in set positions. They can join and leave the network at any time and change position within the network.

The chief advantages of having a MANET based communication is that; there is no need of separate base station, its easy to deploy, it is reconfigurable (adaptable to topology changes), it can be utilized where there is no infrastructure; for instance in emergency disaster situations, and lastly it is robust. However, there are drawbacks as well. Such as, since it is distributed it is hard to control, it has lower capacity, higher packet delay and more severe jitter. Even more, nodes use limited battery and also network maintenance is expensive. This paper studies how artificial intelligence search techniques can greatly reduce the distance that the data needs to travel during transmission. The paper is organized beginning with a little background of MANET, A* search and Dijkstra’s algorithm. Then it discusses the design of a protocol which uses these algorithms to reduce the path traversed. The paper then proceeds and compares the performance of this protocol with swarm intelligence and other protocols [2].

2. BACKGROUND

As adhoc networks lacks infrastructure, each nodes need to work together to form the network to communicate with each other. A node can be a laptop or a PDA or any other communication device. The network is characterized for the absence of base station. Furthermore, the nodes are able to leave or enter or move around the network randomly. The nodes act as routers, and also play an important role in the discovery and maintenance of the routes from source to destination or between nodes. This is the toughest challenge in designing such networks. For instance, if a link breakage occurs, the network has to operate by building new routes. The technique that is usually used is known as multi-hopping. Multi-hopping increases the overall network performances. By multi-hopping one node is able to deliver data on behalf of another one to a determined destination. It also adds to the transmission range of the node.

Dijkstra’s algorithm finds the shortest path between node X and all the other nodes. The idea is to explore and visit all other nodes one at a time as they become visible (sorting according to the lowest path cost from the initial node X) and updating shortest path between that node and the initial
node X until all the nodes have been explored. In this way a node can communicate with all the other nodes in the network. A* search is a graphical search algorithm that finds the least cost path from a given initial node to one goal node. It uses distance plus cost heuristics function (denoted by f(x)) to determine the order in which the search visits the nodes in the tree. The distance plus cost heuristic function is a sum of two functions; the path cost function g(x) and an admissible heuristic (estimate) of:

\[ F(x) = g(x) + h(x) \]
\[ g_2(x) \geq g_1(x) + h_1(x) \]

So, A* search chooses route 1. Here, Figure 1 demonstrates the A* search procedure.

![Fig. 1. Demonstrating A* search technique.](image)

Here, the distance to the goal is h(x). The path cost g(x) is the cost from the starting node to the current node. h(x) the admissible heuristic in routing should be the straight line distance to the node. Swarm intelligence is an artificial intelligence search technique which involves the data traversing the path that is most frequently traversed. It imitates ants which follows pheromones. Ants will always choose the path with the highest concentration of pheromones.

Mobile Ad-Hoc Network (MANET) has significant different characteristics than other wired or wireless networks. Existing networking protocols cannot work in this new environment without considerable modification. This research is to study new and existing protocols for MANET. Research has shown that TCP performs poorly over MANET due to temporary link failure and frequent route changes. One solution is to effectively detect such events and put TCP sender into a "sleep" mode until the route has recovered. In this project, we study various mechanisms to detect and to respond to such events. We have explored a new approach to improve performance by detecting and responding to out-of-order packet delivery events, which are the results of frequent route changes.

3. THE PROTOCOL

All communication takes place in two stages. In the first stage route and specifications are determined and is usually referred as the control information. The second stage is the actual flow of information referred to as the transmission. In our project we would take care of the first stage (control stage) by using Dijkstra’s algorithm and the second stage (transmission stage) using the A* search.

There are many search techniques in artificial intelligence out of which the Dijkstra’s search is one that is most efficient in finding the shortest path to all the other nodes in the network. That is why Dijkstra’s search is used for sending the control information, as that way the control information can find the most efficient path to reach all the nodes in the network. The control message in our case mostly comprises of node position information as in MANET the position of the nodes is always changing dynamically without any definite order. Our work is to find the efficient path within the dynamic environment [3].

A* search on the other hand finds the shortest path to the “goal” node. That way we can find the most efficient path for sending our transmission message from the source node to the destination node. However, it requires information (heuristics) that we would be already gathering in the first stage using Dijkstra’s algorithm. Essentially what A* search does is that it chooses the least cost path to the node which is closest to the destination. Therefore if path cost to a node is g(x) and distance from destination of that node is h(x); A* search chooses the path using the following function.

A to C: \[ g(x)=1 \quad h(x)=1 \quad f(x)= g(x)+h(x) \]
\[ f(1)=1+1=2 \]

A to D: \[ g(x)=2 \quad h(x)=2 \quad f(x)= g(x)+h(x) \]
\[ f(2)=2+2=4 \]

Therefore, node A takes Route 1.

In Figure 2, essentially, what happens is that, whenever the node chooses the next node, it always looks for the shortest distance it can go to reach a new node from any available leaf nodes. That way it finds the least distance to reach every other node in the network. This is how the destination nodes in our project would communicate their position to the rest of the network in the shortest possible path and hence the most efficient way [3].
Fig. 2. Demonstration of Dijkstra’s Algorithm to find the least distance to every other node on the network.

Now we will talk about how the position of each node will be recorded and manipulated. For that we can use a grid system; which will use the Cartesian coordinate axis xyz. We need the entire 3 axis to determine the position as in most cases the MANET will occupy a three dimensional space. Later on we can use the three coordinate values to determine the pin point position of the nodes; and that will be enough. Once we start using the coordinate system, it will be very easy for us to calculate the distance values, simply using the formula:

\[
D = \sqrt{(x_1-x_2)^2 + (y_1-y_2)^2 + (z_1-z_2)^2}
\]

However, right now for making our demonstration simple, we will look into two dimensional space, only dealing with the xy-plane \((z=0)\). The path costs are given in the above diagram. Now first node B would let every other node in the network know its position. It will follow the path below, using Dijkstra’s Algorithm. Therefore the destination node takes 5 steps to reach all the nodes in the network using Dijkstra’s Algorithm. Likewise all the other nodes will do the same to make its position known to all the other nodes in the network. Then the routing table will be constructed. Though the actual format of the routing table will be our concern in the next section, for this section we will show only some details. Node A can now start the transmission. It can choose among 3 routes:

- Route 1: A to C to B
- Route 2: A to D to E to B
- Route 3: A to D to F to B

The path costs are shown below:

- B – E =1
- E – F =3
- B – C =2
- F – D =4
- D – A =5
- D – E =6
- A – C =7

Heuristic values \(h(x)\) of all the nodes are as follows:

- A: \(h(x) = 9\)
- B: \(h(x) = 0\)
- C: \(h(x) = 3\)
- D: \(h(x) = 7\)
- E: \(h(x) = 1\)
- F: \(h(x) = 3\)

In case of more than 1 path, the least value is taken into consideration. Now when Node A starts to make a selection it can choose from 3 routes; route 1, route 2 and route 3.

For Route 1:
\[f(x) = g(x) + h(x) \rightarrow f(x) = 7 + 2 = 9\]

For Route 2:
\[f(x) = 5 + 7 = 12\]

For Route 3:
\[f(x) = 5 + 8 = 13\]

Therefore, even if the initial path cost \(g(x)\) is greater in route 1, node A would choose route 1 because the total path is least as \(f(x)\) is least in route 1. That is how our MANET would function.

3.1 Ladder diagrams

Data communication requires at least two devices working together. One requires to send and the other to receive. For that reason the communication system requires flow control and error control mechanisms. These mechanisms can be demonstrated by the following ladder diagrams.
Therefore the location information is requested via Dijkstra’s algorithm and the reply is also done by Dijkstra’s algorithm. Once the mobile node is located, now the path determination begins. This is done by A* search. The transmission path is determined by the means of A* search, which gives the shortest path possible.

Fig. 5. Ladder diagram for demonstrating the full communication.

Once the shortest path is determined the destination node will respond to the path request by sending the reply via the A* search determined path. Now the source node and destination node is ready to transmit and receive data. Therefore once the path is determined transmission can be two ways. The transmitted data would be numbered and thereby the ACKs would also be numbered in order to keep record of data received and data lost. The serial number enables more than one data packet to be transmitted at a time.

3.2 Block diagrams

The block diagram shows the steps that the protocol would follow to operate efficiently. Once initialized the network would first figure out the whereabouts of the source and destination node. This would be done by Dijkstra’s algorithm and then the data transmission would be done via A* search. Throughout the progress of the communication the network would check if the node location had crossed beyond the border of the particular grid that it was in. If crossed, the location would be determined all over again from the beginning. Else the route would be continued.

Fig. 6. Block diagram to demonstrate signal flow.

4. CHARACTERISTICS OF THE PROTOCOL

This proposed protocol has the following characteristics

i. The protocol works for MANETs (Mobile Adhoc Networks)

ii. For distributing position information the protocol uses Dijkstra’s algorithm instead of broadcasting to decrease the bandwidth required and increase the efficiency.

Fig. 7. Distribute location by Dijkstra’s algorithm.

iii. For actual transmission of data and planning the hop by hop route; the protocol uses A*search algorithm
iv. In case of using A* search, the heuristic value \( h(x) \) is not chosen to be the straight line distance between the intermediate node and the destination node; rather it is the hop by hop distance between the intermediate node and the destination node. This increases the validity of the A* search.

v. Region of consideration is minimized as much as possible. Generally the region of consideration is the circle with the straight line joining the source and destination node as diameter. However for more sophisticated networks the elliptical region is taken into consideration with the major axis as the line joining the source and destination node and the minor very small. Later on if nodes are not available within that elliptical region the minor axis is increased.

vi. Multi-hopping is allowed in this mobile adhoc network. Decisions regarding multi-hopping is done based on A* search.

vii. Information regarding intermediate node arrangement is stored for a small duration equal to \( nx \) where \( x \) is the packet duration and \( n \) is a small number bigger than but close to 1.

viii. The protocol functions in the network layer of the OSI model.

ix. Based on the following aspects the protocol needs to be modified.
   a. Network size/ scalability
   b. Geographical area
   c. Topology rate of change
   d. Quality of service
   e. Energy constrained operation
   f. Limited physical security

x. The protocol works as a reactive-on-demand routing protocol. It discovers routes based on demand.

xi. It discards high values of \( h(x) \), thereby limiting the region of consideration as well as reducing work load for calculations.

xii. The source node knows the complete route to the destination

xiii. No route is cached for future use, for more than the frame time or small multiples of the frame time.

xiv. Routing loops are automatically prevented by discarding high values of \( h(x) \).

xv. The routing is not chosen by the number of hops by rather the least distance. Therefore, a path with greater number of hops but lesser distance is chosen, after problems regarding radio range are overcome.

xvi. The protocol mainly focuses on energy efficiency; adoptable by least distance.

xvii. The protocol starts functioning when there are two or more intermediate nodes between the source and destination.

xviii. In case of position determination the protocol uses the Cartesian coordinate system.

xix. We are assuming no congestion, hence multipath is not required.
xx. A grid system is used to round-up distance measurements. Plus, this makes the system quasi static as the rounded values or routes do not change as long as the node stays within the specific grid.

xxi. Part of this protocol can be integrated with other protocols to increase the distance efficiency of any adhoc network.

5. RESULTS OF EMPIRICAL STUDIES
This section contains the results of the empirical studies of the proposed protocol. Now we will show how the least distance varies when the adhoc network works with different number of nodes, 10, 100 and 1000 nodes. It can be seen that with 10 nodes even though the distance traversed remain limited, the value shows certain increment and decrement. However, overall the A* search shows a lot lesser distance traversed than other protocols.

With 10 nodes the least distance varies considerably. However, increasing the number of nodes stabilizes the distance that the data needs to traverse. The following diagram shows how the distance traversed varies with different number of nodes and with different protocols.

Next we will show how the least distance varies when the adhoc network works with different number of nodes, 10, 100 and 1000 nodes. With 10 nodes the least distance varies considerably. However, increasing the number of nodes stabilizes the distance that the data needs to traverse. The following diagram shows how the distance traversed varies with different number of nodes and with different protocols.

Now we will determine how the situation changes with different protocols. First let us see what happens when we have very few nodes, like 10 nodes.
We can see that in this situation A* protocol shows minimum distance at all times. Swarm intelligence shows occasional minimum distances but sometimes it gives unpredictably high distances. As these are under the curve for the A* protocol is the least, it indicates that A* protocol allows least total distance traversed. Swarm intelligence does show occasional efficient short paths but is unpredictable at times, overall, the A* protocol shows least path in all the situations. Now let us increase the number of nodes to 100 nodes [3].

In this situation the results are similar, A* protocol shows least distance at all times, swarm intelligence shows occasional short path, but sometimes the distance traversed is high. However, the variation is that, the distance traversed by the A* protocol stabilizes to a minimum. Now let’s see what happens with 1000 nodes.

In this situation the A* search shows minimum distance compared to all the other situations. If we compare the different scenarios we see that no matter what the number of nodes might be, A* search always guarantees least distance. Sometimes other protocols show
comparable results but most of the time they are way up in terms of distance traversed.

6. CONCLUSION
The core advantages of having a MANET based communication is that there is no need of separate base station, its easy to deploy, it is reconfigurable (adaptable to topology changes), it can be utilized where there is no infrastructure (emergency disaster situations), and lastly it is robust. Based on all the above findings we can conclude that A* search is indeed an efficient way to reduce path in mobile adhoc networks. Swarm intelligence does show occasional efficient short paths but is unpredictable at times, overall, the A* protocol shows least path in all the situations. A* protocol does give the minimum distance for the data to travel. Therefore using A* search and Dijkstra’s algorithm energy efficient MANETs can be constructed. This protocol can be used partly or fully to increase the efficiency by decreasing the path traversed our adhoc networks. Further work can be done by studying the time efficiency and security of adhoc networks using A* search.

REFERENCES