

Distance Based Route Maintenance Strategy for Dynamic Source Routing Protocol

Varun Garg¹

¹Department of Master of Computer Application
Faculty of Master of Computer Application
Gyan Ganga Institute of technology Jabalpur
M.P. India

Poornima Sharma²

²Department of Master of Computer Application
Faculty of Master of Computer Application
Gyan Ganga College of technology Jabalpur
M.P. India

Abstract- Dynamic Source Routing (DSR) [1, 2], which is an on-demand routing protocol, becomes the most popular source routing protocol for MANET. Each mobile node is required to maintain route caches that contain the source routes of which the mobile is aware. Although DSR can respond a route quickly, it yields a long delay when a route is rebuilt. This is because when source node receives RERR packet, it will try to find alternative routes from route cache. If alternative routes are not available, source node, then, will enter route discovery phase to find new routes. Finding a route in wireless network require considerable resources, such as time, bandwidth, and power because it relies on broadcasting. In this paper we introduce a route maintenance strategy as DISTANCE (Distance baSed rouTe maintenANCE). DISTANCE works by adding another node (called bridge node) into the source list to prevent the link from failure. From the simulation result, DISTANCE improves the performance of DSR in terms of packet sending ratio and delay.

I. INTRODUCTION

Dynamic Source Routing (DSR) [1, 2], which is an on-demand routing protocol, becomes the most popular source routing protocol for MANET. Each mobile node is required to maintain route caches that contain the source routes of which the mobile is aware. Entries in the route cache are continually updated as new routes are learned. In case of link/route failure,

the node, which detects link/route failure, will send route error (RERR) packet to source node. Although DSR can respond a route quickly, it yields a long delay when a route is rebuilt. This is because when source node receives RERR packet, it will try to find alternative routes from route cache. If alternative routes are not available, source node, then, will enter route discovery phase to find new routes. Finding a route in wireless network require considerable resources, such as time, bandwidth, and power because it relies on broadcasting. To solve that problem, we can use a route maintenance strategy that utilized location information. We call the strategy as DISTANCE (DIsTance baSed rouTe maintenANCE). We had developed a novel route maintenance strategy for DSR, called DISTANCE [3]. From simulation results, it shows that DISTANCE has successfully improved the performance of DSR, especially in high density nodes.

The con of DISTANCE is that DISTANCE was designed to work in synchronous communication network. It means that each node communicates in the same radius. In reality, wireless communication radius may change from time to time depend on the condition at that time, such as obstacles, atmosphere absorption, signal interferences, etc. To make DISTANCE able to work in asynchronous communication network.

II. DISTANCE BASED ROUTE MAINTENANCE

Today there is quite a lot of mobile device such as cell phone and PDA, which is equipped with GPS. Mostly the application of GPS is intended for guiding purposes such as road tracking. Besides using GPS to make known our current location, we can also use it for routing purposes. Each node may know another nodes location by exchanging their current location (we refer this as location information). By knowing another nodes location, the routing process could be optimized. Our assumptions are:

- Each node knows its current location (i.e. with Global Positioning System)
- All links are bi-directional and all nodes have the same transmission range (synchronous transmission range)
- Each node also maintains a location table that contains the positions of all its neighbours.
- The routes are already established.
- Route failure that is occurred by node disappearance (i.e. out of energy) is not considered.

Based on location information, DISTANCE tries to prevent the link from failure. Basically there are two main procedures in DISTANCE, in which, first we detect some unsafe link, and then we expand the route by adding another node into the source list.

```

// SignalPower = Current signal power level
// SrcNode = Previous-hop node
// LinkMargin = Current link margin
// Radius = Current maximum communication radius

Function CheckLastHopLink (SignalPower, SrcNode)
  LastHop = SrcNode;
  LinkMargin = CalculateLinkMargin();
  Radius = CalculateDistanceToLastHop();
  If(LinkMargin < MinimumLinkMargin)
    FindBridgeNode(LastHop, Radius);
  End If
End Function

```

Figure 1: Pseudo code of unsafe link detection.

A. Unsafe link detection

To detect unsafe link, the next hop of current active node will measure Link Margin value when receives a data packet from current active node. A link is said in safe condition if Link Margin is more than the minimum Link Margin; otherwise the link is said in unsafe state. The minimum Link Margin itself is predefined signal strength level to guarantee the communication quality. Figure 1 shows the pseudo code of the unsafe link detection algorithm.

B. Expanding the routes

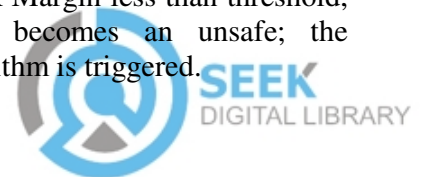
Once a link is detected unsafe, the current active node will send a local broadcast packet (one-hop packet) to its neighbours for finding a bridge node to the next hop. If neighbour nodes have a link to the next hop, then these neighbour nodes will also calculate RD to both current active and the next hop node. If RD is more than TH (using the same formula to detect unsafe link) then this node will not propose itself to the current active node as a bridge node; otherwise it will propose itself as a bridge node to the current active node. After some times, the current active node will receive some proposed bridge nodes, and then it will decide which of the proposed node to be chosen as a bridge node, based on $Min((RD(A,B1)+RD(B1,C))/2, \dots, (RD(A,Bn)+RD(Bn,C))/2)$. If neither neighbour nodes, which have RD less than TH, nor candidate nodes are available, DISTANCE will just let the link as is (no action will be taken, let normal DSR route maintenance works).

C. Exchanging Location Information

In this work, we used piggyback method to exchanging location information between nodes. Each node will update its location by piggybacking its location into packet header of any packet. From our observation during the experiment, piggyback method works better than using separate routing packet. Therefore, it can maintain the routing overhead on its reasonable value.

D. Case study

From figure 2, let say node A wants to send data packets to D. The route to reach D from A is A-B-C-D. The current active node will measure Link Margin to last-hop node. For this example, let say the threshold for Link Margin before the ADISTANCE algorithm start to works is 15dB, and M is 0.5. In early condition the link is fine; the entire Link Margin is more than the threshold. But, when node D move to D', the Link Margin from C to D' is changed to 14dB. This will make Link Margin less than threshold, so then the link becomes an unsafe; the ADISTANCE algorithm is triggered.



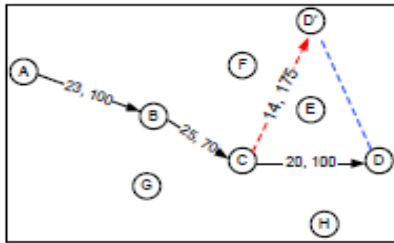


Figure 2. Node D move to D', makes link from node C to node D' becomes unsafe.

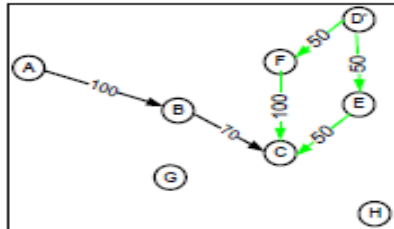


Figure 3. Node D' sends a local broadcast packet to get candidate nodes for bridging the link.

Node D' will send a local broadcast packet to its neighbours (i.e. node E, F), asking whether there is any nodes that have a link to C. In figure 4, node D' managed to get two of its neighbours (E, F), which has a link to C (assumed links from node E and F to D' are safe). Nodes E and F decide to propose themselves to node C as a bridge node. Based on the minimum cost, node C will chose node E to bridge the link between node C and D'. The new route after expansion becomes A-B-C-E-D' as shown in figure 5 as well.

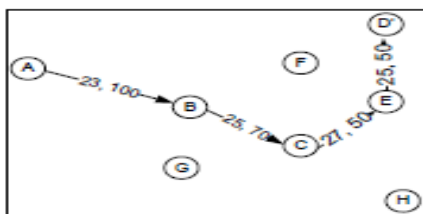


Figure 4. New routes after expansion.

III. SIMULATION ENVIRONMENTS

We used JiST-SWANS [4-6] to simulate our proposed model. We ran JiST-SWANS on a PC with 3.8 GHz Dual-Core microprocessor with 2GB RAM. Summary of simulation parameters can be seen at table 1.

As for the performance metric, we will study the following metrics to compare DISTANCE with the standard DSR route maintenance protocol:

- Packet delivery ratio – Number of received packets divide by number of sent packets.
- Packet delivery delay – Time taken to send a packet from source node to destination node.
- Routing Overhead – Number of packets that being used for routing purposes.

TABLE 1

SIMULATION PARAMETERS

Parameters	Value
Area	1000 x 1000 m2
Number of nodes	40
Number of connection	10
Mobility Model	Random Waypoint with 0 pause time
Transmission Range	250m
Data Packet	Constant, with 512bytes packet size
Simulation Time	500s
Mobility Speed	10 to 50 m/s with 10m/s increment
Multiplier Factor (M)	0.25, 0.50, 0.75

IV. SIMULATION RESULT

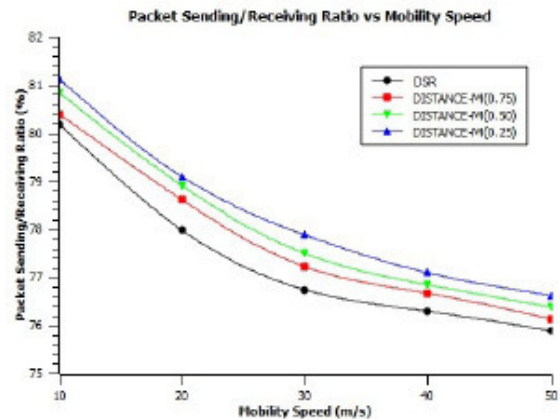


Fig.5. Packet Sending/Receiving Ratio vs Mobility Speed

From figure 5, we can see that DISTANCE gives good impact in terms of packet delivery ratio into the performance of DSR. From here we also can see that M value played important role in DISTANCE. By set M value near 0, it makes the algorithm works more often, and improves the performance of DSR more significant, but the consequences from there the

packet delay and the routing overhead will be higher (even though in some situation based on simulation result still lower than what legacy route maintenance algorithm did). Setting M value so near to 1 is also not recommended, since it will makes DISTANCE algorithm works less often, and there are two factors that contribute packet delay, first is delay because of route expansion, and second is delay because of route reconstruction (assume that current node unable to find a bridge node for route expansion). Figures 5 and 6 show this phenomenon. From simulation result, we consider the M value should be set around 0.5 and 0.75, since the overall performances works better than the legacy route maintenance algorithm.

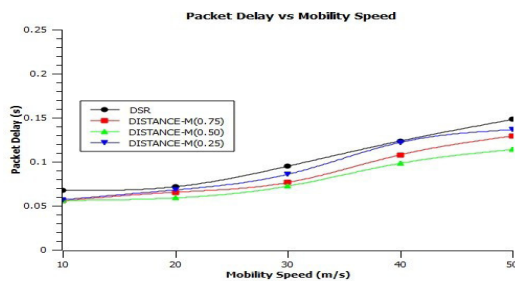


Fig.6. Packet Delay vs Mobility Speed

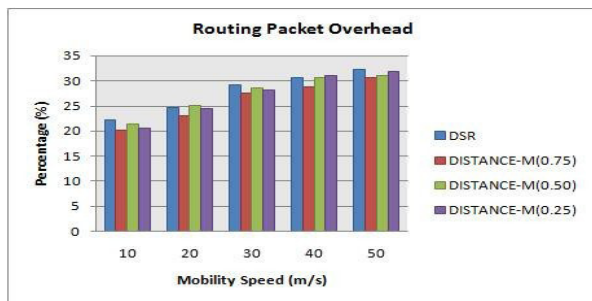


Fig.7. Routing Packet Overhead vs Mobility Speed

V. CONCLUSIONS AND FUTURE WORKS

DSR, which is an on-demand routing protocol, becomes the most popular source routing protocol for MANET. In case of link/route failure, the node, which detects link/route failure, will send RERR packet to source node. Although DSR can respond a route quickly, it yields a long delay when a route is rebuilt. This is because when source node receives RERR packet, it will try to find alternative routes from

the route cache. If alternative routes are not available, source node, then, will enter route discovery phase to find new routes.

We proposed a new route maintenance strategy for DSR, called DISTANCE. From simulation results, we showed that DISTANCE improves the functionality of DSR in terms of packet sending ratio and delay by preventing the links from failure.

As the extension of DISTANCE algorithm, not only gives ability to be able to work in asynchronous communication network, but also improves the performance of DISTANCE, which is indirectly also improves the performance of DSR. The results conclude that in high node density environment, DISTANCE can work better since the probability of founding bridge node is higher.

For the continuity of this works, we currently investigate the effect of node mobility pattern into the performance of the routing protocol. Nodes might move from one place to another place in many patterns either in high temporal and spatial mobility or just simply random. High temporal and spatial are mean that each node mobility is restricted to previous movement and limited into certain border only. This can affect the performance of MANET routing protocol. Future researchers also should focus on integrating these route maintenance algorithms with route discovery algorithm which is also exploits geographical location information. For example integrate DISTANCE with LAR or even DSR-GPS.

REFERENCES

- [1] D.B. Johnson, D. A.Maltz and Y. Hu, "The Dynamic Source Routing (DSR) Protocol for Mobile Ad Hoc Networks," in *IETF Internet Draft, draft-ietf-manet-dsr-10.txt*, work in progress, July, 2004. Visited on January 2007.
- [2] D.B. Johnson and D.A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," in *Mobile Computing*, Imielinski and Korth, Kluwer Academic Publishers.
- [3] M.F. Sjaugi, M. Othman and M.F.A. Rasid, "A New Route Maintenance Strategy for Dynamic Source Routing Protocol," in

- International Communication Information Network, IEEE Proceedings of, Busan, 2008.
- [4] Java in Simulation Time / Scalable Wireless Ad hoc Network Simulator, <http://jist.ece.cornell.edu> .
- [5] R. Barr, "SWANS User Guide," University of Cornell, Available: <http://jist.ece.cornell.edu/docs/040319-swans-user.pdf>. 2004. Visited on January 2007 .
- [6] B. Viglietta, "Dynamic Source Routing for SWANS," University of Cornell, Available: <http://jist.ece.cornell.edu/docs/040108-swans-dsr.pdf>. 2004. Visited on January 2007.