Power Aware Location Aided Routing in Mobile Ad-hoc Networks

Anamika

Computer Science, Inderprastaha Engineering College
Email: anamikachandruka@gmail.com

Abstract- Mobile ad-hoc network (MANETs) [1] is defined as an autonomous system of mobile nodes and associated hosts connected by wireless links. Network topology changes due to change in location of nodes in the network. This makes it more difficult to find the routes those packets. In this paper, a Power Aware location aided routing algorithm (PALAR) scheme by reducing size of request zone, limit flooding and use available bandwidth and battery life in an efficient manner.

Index Terms- Request Zone, Expected Zone, Location aided routing, Global positioning system, Mobile ad-hoc networks, Routing.

I. Introduction

Mobile ad-hoc network (MANETs) [1] is defined as an autonomous system of mobile nodes and associated hosts connected by wireless links. Due to the high mobility of nodes, the network topology of MANETs always changes and task of finding and maintaining routes become difficult. Several routing protocols have been proposed for, with the goal of achieving efficient routing[2]. Routing protocols for wireless ad hoc networks can be mainly arranged into the two categories: Table-driven (or Proactive) and On-demand (or Reactive/Source Initiated). Proactive protocols continuously learn the topology of the network by exchanging topological information among the network nodes[3]. Thus, when there is a need for a route to a destination, such route is readily available [2, 3].In source initiated routing protocols mobile nodes maintain path information for destinations only when they need to contact the source node or relay packets [2].

In recent times, many routing protocols have been proposed which use the Global Positioning System (GPS) [2, 4] to identify location information of destination and improve the routing performance. The coordinates of each node can be known by using GPS. Various GPS applicable protocols like LAR, DREAM,ILAR,LARDAR etc have been proposed to provide efficient routing.

In this paper, we propose a routing scheme that make use of the global positioning system (GPS) to improve routing in MANETs. First use the location information of destination node to predict a smaller triangle request zone that covers the position of destination in the past. Then decide a baseline, which is the line between the source node and the destination node, for route discovery. Then source node sends an information request to intermediate nodes and receives their replies and then identifies the node making minimum angle and have required battery life. The source then forwards route request packet to this chosen node. This procedure is repeated till the complete path to destination is found.

II. RESEARCH ELABORATION

In **Location-Aided Routing** (LAR) [5] is an on demand routing protocol in which the overhead of route discovery is decreased by utilizing location information of mobile nodes obtained using Global Positioning System (GPS) [4]. Location information is used to identify request zone and expected zone. Request zone is an rectangular area that includes both sender and receiver.

Expected Zone: Expected Zone is the region where source node S thinks that the destination node D may contained at some time *t* assuming that node S knows that the node D was at location L at time *t*0 and current time is *t*1.

From the viewpoint of S, expected zone of node D is the region that node S expects to contain node D at time *t*1 based on the knowledge that node D was at location L at time *t*0. Now, If S knows that D travels with average speed *v*, then S assumes that the expected zone is the circular region of radius *v*(*t*1- *t*0) centered at location L

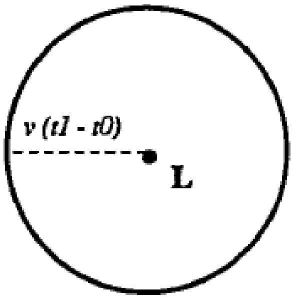

FIGURE 2.1:EXPECTED ZONE

Request Zone :The request zone is created as a rectangle with the source s in one corner and the expected zone in the opposite corner of the rectangle.

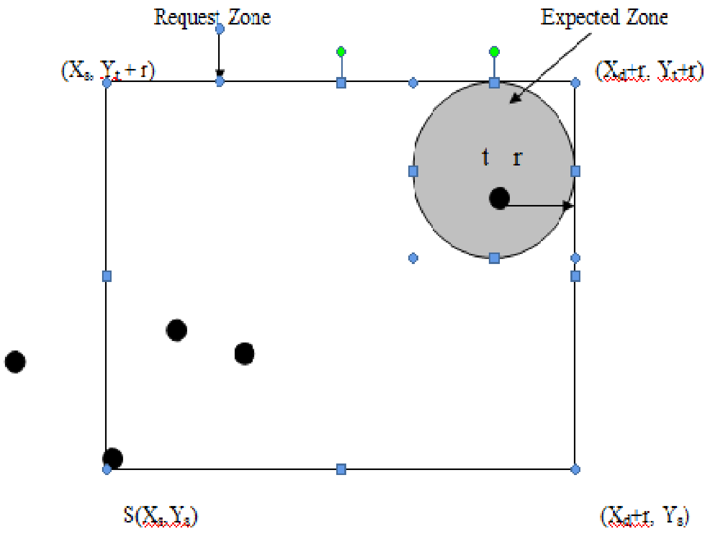

FIGURE 2.2 :REQUEST ZONE AND EXPECTED ZONE

Distance Routing Effect Algorithm for Mobility

The Distance Routing Effect Algorithm for Mobility (DREAM) protocol is table driven routing protocol reduces overhead of route discovery by storing *location information* for every other node within the network in the routing table[6].

DREAM may be considered part proactive and part reactive in nature. The nodes within a DREAM environment have a means of disseminating and collectively updating the location table entries for each other, behaving as a proactive protocol. When an information packet needs to be transported form node A to node B, node A looks up the location of B from within its tables and forwards the packet to nodes “in the direction” of B, as the next hop node. These intermediate nodes in turn perform a lookup and forward the packet “in the direction” of B. This results in the protocol mechanism reflecting a reactive nature.

Improved Location Aided Routing (ILAR)

ILAR (Improved Location Aided Routing) [2] is another location based technique which uses the concept of base line lying in between the source and destination node. Node which is closest to this line of sight will be chosen as the next intermediate node. As the transmitting node check the distance of every neighboring node from base line and find the closest neighbor for further transmission. This process will increase the delay in data which in turn decreases battery life.

Assume *S* (X_s,Y_s) and $D(X_d,Y_d)$ are the coordinated of source and destination identified using location aided routing. Now, baseline can be determined by using the equation:

$$(x_d - x_s)(y - y_s) - (y_d - y_s)(x - x_s) = 0$$

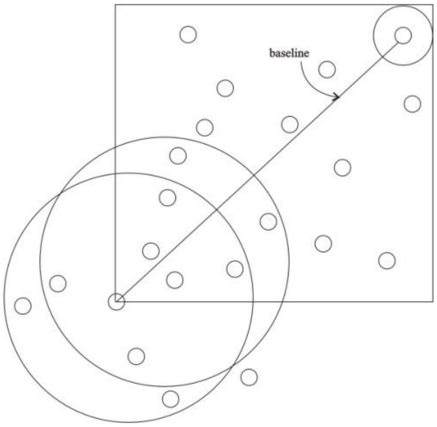

FIGURE 2.3: EXAMPLE OF BASELINE

$DIST_A$: The distance between the source node and node A.
$VDIST_A$: The distance from node A to the baseline.

Node *A* receives the route request from source node *S*, and calculates the distance between the source node and itself, denoted as $DIST_A$. Moreover, node *A* will calculate the distance from the node to the baseline, denoted as $VDIST_A$. We assume that the baseline is $ax + by + c = 0$.$VDIST_A$ can be obtained from the following equation:

$$VDIST_A = \left| \frac{ax_A + by_A + c}{\sqrt{a^2 + b^2}} \right|$$

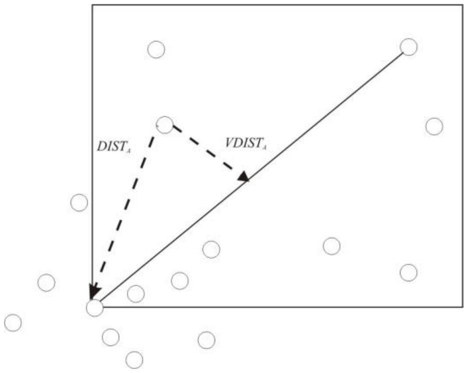

FIGURE 2.4:DIST AND VDIST

LARDAR (Location Aware Routing Protocol with Dynamic Adaptation of Request Zone for Mobile Ad hoc Networks) [7] is an on demand routing protocol which decreases the area of request zone by utilizing destination node location information and velocity which in turn reduces number of route request. Dynamic adaptation of request zone technique is used to trigger intermediate nodes using its more fresh location information of destination node or the information provide by precedent node to redefine a more precise request zone to improve the performance of routing protocol.

Request Zone

Source node is outside of expected zone: If source node S is outside of the expected zone, we define the request zone to be the smallest isosceles triangle, named TRIANGLEZONE, which includes the current location of S and the expected zone.

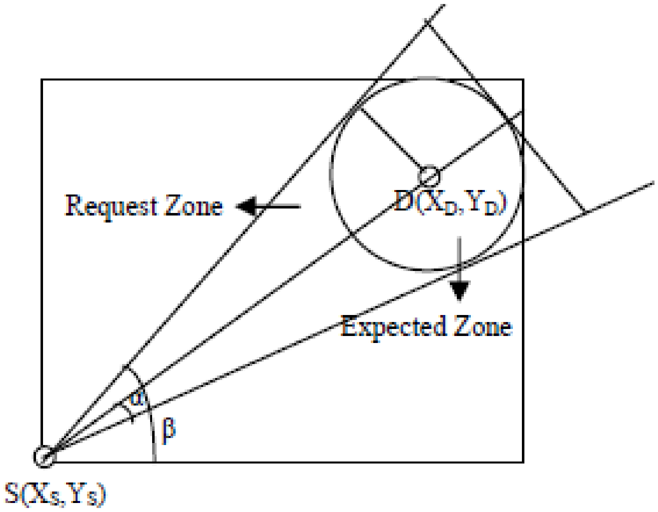

FIGURE 2.5: REQUSET ZONE AND EXPECTED ZONE

III. PROPOSED WORK

In proposed protocol the source node discovers the location of the destination and then it estimates the average velocity of the destination using its knowledge of previous destination movement's. Using this calculated average speed, the last time when it knows the position of D, and the location of D, the source node calculates the expected zone of D. Using this expected zone, it creates the request zone of D and send's request to all its 1-hop neighbors. Each neighbor on receiving the request packet checks whether it is the destination or not. If it is the destination then it replies to source to start transmission of data packets else it sends back a reply to source with $\theta_1^{'}$ and remaining battery life. Source node selects the neighbor node on the basis of smallest $\theta_1^{'}$ and battery life after receiving replies from its neighbors within the specified period of time, and sends the request packet to that node to find the path to the destination. If battery life is less than required battery than another intermediate node with smallest $\theta_1^{'}$ and this process is repeated at each node (that receives the request to find the path) till the destination is reached.

A. Message Details

Route Request packet: Source node sends route request packets to all its one hop neighbors to indicate that it is trying to find a route to a given destination node.

SID	DID	Q1	SrcPos	RB

SID: Identification Id of Source
DID: Identification Id of Destination
SrcPos: Location of source
θ_1= Angle sent by intermediate node
RB: Required battery life.

Route reply packet: Each node on receiving a route request packet, sends back a route reply .

SID'	Q'	Battery

SID'=Identification Id of intermediate node
$\theta_1^{'}$=angle subtended by the intermediate neighbour node with the source
Battery= battery life of intermediate node.

B. Calculation of θ_1

Baseline: The straight line connecting source and destination. The slope of the baseline can be calculated using the formula:

$$\theta = \tan^{-1}(y_2 - y_1)/(x_2 - x_1)$$

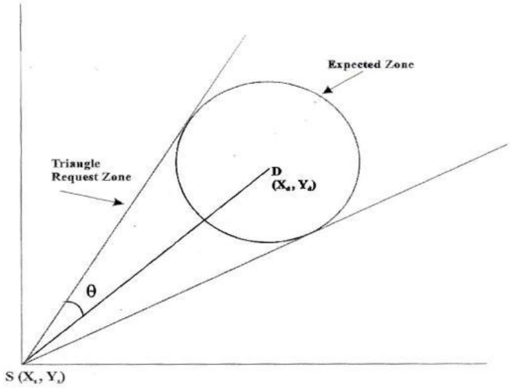

FIGURE 3.1:Calculation of Θ

Observation1: ΔASD ≅ΔBSD

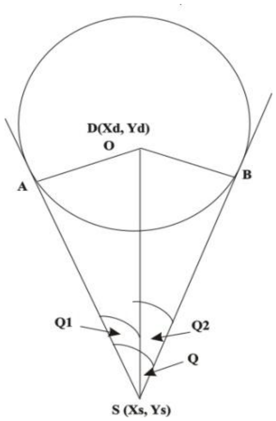

FIGURE 3.2:HOW Θ1 AND Θ2 ARE EQUAL

Proof: In ΔASD and ΔBSD

- SA=SB (Tangents to a circle from same point are of equal length)
- SO = SO (Common Side)
- DA=DB (radius of circle)

Therefore by SSS theorem, ΔASD and ΔBSD are congruent.

Observation2: $\theta_1 = \theta_2 = ½ \theta$

Proof: ΔASD ≅ΔBSD (from observation 1)

- $\theta_1 = \theta_2$
- Also $\theta + \theta_2 = \theta$

From above equations: $\theta_1 = \theta_2 = ½ \theta$

Observation 3: $\theta_1 = \sin^{-1} v (t_1 – t_0) / \sqrt{} (x_2 – x_1)^2 + (y_2 – y_1)^2$

Proof: $\angle OAS = 90^0$ (as DA is line joining centre of circle to tangent)

Therefore, $\sin(\theta_1) = v (t_1 – t_0) / \sqrt{} (x_2 – x_1)^2 + (y_2 – y_1)^2$

Hence, $\theta_1 = \sin^{-1} v (t_1 – t_0) / \sqrt{} (x_2 – x_1)^2 + (y_2 – y_1)^2$

Observation 4: $\angle I_1SD$ = Slope of SD – slope of SI

Let T be the point where line DS meets the x axis

Let P be the point where line IS meets the x axis

Now $\angle DTX$ = slope of SD (as XI_1 is parallel to SD)

Now $\angle IPX$ = slope of SI (as XI_1 is parallel to SD)

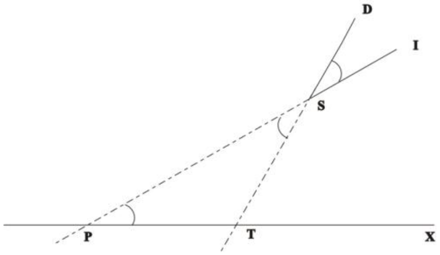

FIGURE 3.3: CALCULATION OF ANGLE SUBSTENDED BY INTERMEDIATE NODE

In ΔSPT, $\angle PST = \angle STX - \angle SPX$

$= \angle DTX - \angle IPX$

= Slope of SD – slope of I_1D

C. Algorithm

Global parameters for each node:

- **routeSearchInProgress** : Hash table that track which nodes are being searched
- **routingTable**: Table that stores the routing paths at this node. Any suitable version can be plugged in.

```
//This procedure is executed by node when it is searching for path to some node
Void searchRoute(nodeId dst)
 {
        routeSearchInProgress[dst] = 1 ;
        // calculate expected zone
        dstLocation = retrieve_location( dst); //retrieve location of dst node
        $T_1$ = locationLastUpdatedAt(dst); // last time destination location was updated
        $T_0$ = getTime();
        V = getAvgVelocity( dst);
        Expected_zone = Calculate_expected_zone( t1, to , v dstLocation );
        //calculate request zone
        srcLocation = retrieve_location( thisNode);
        Request_zone = Calculate_request_zone( expected_zone, srcLocation);
        //broadcast route request to all neighbors
        Request = Make_request (uniqueNodeId, dst , requestZone. θ , srcLocation,RB) ;
        Send (request , -1 );
        //intermediate nodes would run the process incoming request function
        Timer.setTimeout = 2 milli seconds;
        minId = -1 , minAngle = θ;
        // process all replies and choose the node making minimum angle with baseline
        While (timer)
        {
        waitPacket();
        CalculateAngle ø with the baseline (| $\theta-\theta_1$’|)
        If ( ø < minAngle )
        {       minId = N , minAngle = ø }
        }
        // send route request packet to the selected node
        If( minId != -1 )
        {       // create route request packet and send to node with id minId
        Packet p = new RouteRequest(uniqueNodeId, dst, srcLocation, θ,RB);
        send(p , minId) ;
        return              }
        Send ( p , RERR); // send packet to indicate that no route could be found
        }
```

```
//This procedure is run by all nodes when listening to the network
Void listenatintermediatenode ()
{
        Packet p;
        While (true)
      {
          WaitPacket( p); // wait till a packet comes in, and fill that packet into p
          Handle_packet(p); //helper function for processing incoming packets
       }
}

//This procedure is executed by intermediate nodes on receiving the route request packet
Void processIncomingRequest(Packet p)
{
      If( p.DID = uniqueNodeId )
      {
      Packet routeReply = new RouteReply( RREP, p.SID, p.DID,Battery);
      Send( routeReply, -1 ) ; //broadcast the route reply packet
      }
      else
      {
      $\theta_1'$ = calculate $\theta_1'$ ( p.SID, p. DID, uniqueNodeId); //as described earlier
      Packet reply = new reply (uniqueNodeId, $\theta_1'$ );
      Send (reply, p.requestSID);
      }
} //End of processing Route Request packets

//This procedure is executed by nodes on receiving a route reply packet
Void processIncomingRouteReply(Packet p)
{
      //update routing table as suitable based on this packet
      updateRoutingTable(p);
      If ( routeSearchInProgress[dst] == 1 )
      {
      //broadcast routing entry for this node if a path to it is being searched
      forwardRoute(dst);
      routeSearchInProgress[dst] = 0 ;
      }
} // end of process incoming route reply
```

IV. CONCLUSION

This paper presented and evaluated a protocol (PALAR) for routing packets between mobile nodes in an ad hoc network. Proposed protocol minimize bandwidth consumption, flooding and battery consumption of mobile nodes by decreasing area of request zone and minimizing number of requests and response evaluation at every step. PALAR will provide efficient routing by choosing the nodes which have required battery life to transmit the data packet.

REFERENCES

[1] S. Gangwar, K. Kumar, "Mobile Ad hoc Networks: A detailed survey of QoS Routing Protocols," vol. 2, NO.6, International Journal of Distributed and Parallel Systems, 2011 .

[2] N. C. Wang, Y. F. Huang, J. S. Chen, S. M. Wang, C. L. Chen, "An Improved Location-Aided Routing Protocol for Mobile Ad Hoc Networks with Greedy Approach," vol.8, Issue 8, Wseas Transaction on Communications," 2009.

[3] G. V. Kumar, Y. V. Teddy, M. Nagendra, "Current Research Work on Routing Protocols for MANET: A Literature Survey," vol. 2, NO.3, International Journal of Computer Science and Engineering, 2010, pp. 706–713.

[4] E. D. Kaplan, "Understanding GPS: Principles and Applications," Artech Hourse, Boston, MA.

[5] Y. Ko, N. H. Vaidya, "L Location-aided routing (LAR) in mobile ad hoc networks," vol. 6, NO.4, ACM Wireless Networks, 2000, pp. 307–321.

[6] I. S. Basagn, Chlamtac, V. R. Syrotiuk, "A distance routing effect algorithm for mobility (DREAM)," 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking, Dallas, Texas, USA, 1998.

[7] T. F. Shih H.C. Yen, "Location-aware routing protocol with dynamic adaptation of request zone for mobile ad hoc networks", Springer Wireless Network 14:321–333,DOI 10.1007/s11276-006-9955-y, 2008.