International Journal of Advanced Research Methodology in Engineering & Technology,
Volume 1, Issue 1, December 2016

POSITION UPDATE FOR GEOGRAPHIC ROUTING IN MANET

INDHUREKHA.S[1], CHANDRU VIGNESH.C[2], VENNILA.[3], VARSHINI.A [4]

SNS College of Technology, Coimbatore, India

[1]indhurekhame804@gmail.com

csechandru@yahoo.in

sivaranjini@gmail.com

ABSTRACT

Mobile ad hoc networks (MANETs) consist of a collection of wireless mobile nodes which dynamically exchange data among themselves without the reliance on a fixed base station or a wired backbone network. MANET nodes are typically distinguished by their limited power, processing, and memory resources as well as high degree of mobility. In mobile ad hoc network there are several routing algorithms, which utilize topology information to make routing decisions at each node. In geographic routing, nodes need to maintain up-to-date positions of their immediate neighbors for making effective forwarding decisions. Periodic broadcasting of beacon packets that contain the geographic location coordinates of the nodes is a popular method used by most geographic routing protocols to maintain neighbor positions. We contend and demonstrate that periodic beaconing regardless of the node mobility and traffic patterns in the network is not attractive from both update cost and routing performance points of view. We propose the Adaptive Position Update (APU) strategy for geographic routing, which dynamically adjusts the frequency of position updates based on the mobility dynamics of the nodes and the forwarding patterns in the network. APU is based on two simple principles: 1) nodes whose movements are harder to predict update their positions more frequently (and vice versa), and (ii) nodes closer to forwarding paths update their positions more frequently (and vice versa). Our theoretical analysis, which is validated by NS2 simulations of a well-known geographic routing protocol, Greedy Perimeter Stateless Routing Protocol (GPSR), shows that

APU can significantly reduce the update cost and improve the routing performance in terms of packet delivery ratio and average end-to-end delay in comparison with periodic beaconing and other recently proposed updating schemes. The benefits of APU are further confirmed by undertaking evaluations in

realistic network scenarios, which account for localization error, realistic radio propagation, and sparse network.

I.INTRODUCTION

Routing in ad hoc and sensor networks is a challenging task due to the high dynamics and limited resources. There has been a large amount of non-geographic ad hoc routing protocols proposed in the literature that are either proactive (maintain routes continuously), reactive (create routes on-demand) or a hybrid. For a survey and comparison. Non-geographic routing protocols suffer from a huge amount of overhead for route setup and maintenance due to the frequent topology changes and they typically depend on flooding for route discovery or link state updates, which limit their scalability and efficiency [5].

On the other hand, geographic routing protocols require only local information and thus are very efficient in wireless networks. First, nodes need to know only the location information of their direct neighbors in order to forward packets and hence the state stored is minimum. Second, such protocols conserve energy and bandwidth since discovery floods and state propagation are not required beyond a single hop. Third, in mobile networks with frequent topology changes, geographic routing has fast response and can find new routes quickly by using only local topology information [7].

Each node knows its geographic location using some localization mechanism. Location awareness is essential for many wireless network applications, so it is expected that wireless nodes will be equipped with localization techniques. Several techniques exist for location sensing based on proximity or triangulation using radio signals, acoustic signals, or infrared. These techniques differ in their localization granularity, range, deployment complexity, and cost. In general, many localization systems have been proposed in the literature: GPS (Global Positioning System), infrastructure-based localization systems, and ad-hoc localization systems. Each node knows its direct neighbors' locations. This information could be obtained by nodes periodically or on request broadcasting their locations to their neighbors [1].

II.ROUTING MECHANISM

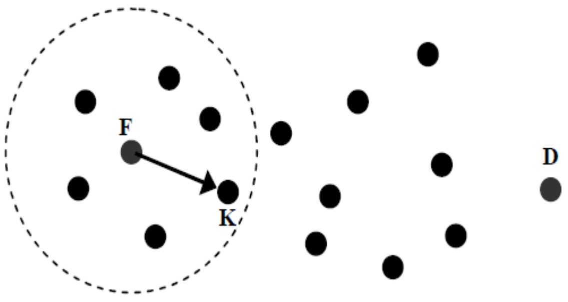


Figure 1: Greedy forwarding

In Figure 1. Greedy forwarding Node F forwards the packet to neighbor K, which is the neighbor closest to the destination D. GPSR present the common form of greedy forwarding in ad hoc networks. Packets contain the position of the destination and nodes need only local information about their position and their immediate neighbors' positions to forward the packets was shown in above Figure 1.2. Greedy forwarding is very efficient in dense uniform networks, where it is possible to make progress at each step [14].

Greedy forwarding, however, fails in the presence of voids or dead-ends, when reaching a local maximum, a node that has no neighbors closer to the destination that was shown in Figure .3. In this case, it will fail to find a path to the destination, even though paths to the destination through farther nodes may exist. Previous protocols deals with dead-ends in different ways. In MFR, if no progress could be made in the forward direction, the dead-end node sends the packet to the least backward neighbor, which is the neighbor closest to the destination among its neighbors. This could cause looping and

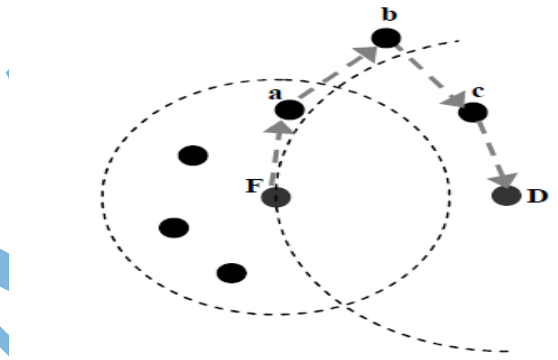


Figure 2: Greedy forwarding fails at node

Greedy forwarding fails at node F, since there are no neighbors closer to the destination D, although a path through a farther neighbor F-a-b-c-D exists nodes need to detect when they get the same packet for a second time. The existing using limited flooding for a number of hops to overcome dead-ends. When a node is reached that has no neighbors closer to the destination, it sends a search packet for *n* hops away. Closer nodes to the destination reply back and the closest node to the destination among those nodes is chosen to forward the packet. The value of n is set based on the topology structure (estimated size of voids) and the desired degree of robustness.

III FRONT END NETWORK SIMULATOR (NS)

Network simulator 2 is used as the simulation tool in this project. NS was chosen as the simulator partly because of the range of features it provides and partly because it has an open source code that can be modified and extended. Network simulator (NS) is an object–oriented, discrete event simulator for networking research. There are different versions of NS and the latest version is ns-2.1b9a, while ns-2.1b10 is under development, NS provides substantial support for simulation of TCP, routing and multicast protocols over wired and wireless networks. This simulator is a result of an ongoing effort of research and development. Even though there is a considerable confidence in NS, it is not a polished product yet and bugs are being discovered and corrected continuously.

NS is written in C++, with an OTcl1 interpreter as a command and configuration interface. The C++ part, which is fast to run but slower to change, is used for detailed protocol implementation. The OTcl part, on the other hand, which runs much slower but can be changed very fast quickly, is used for simulation configuration. One of the advantages of this split-language program approach is that it allows for fast generation of large scenarios. To simply use the simulator, it is sufficient to know OTcl. On the other hand, one disadvantage is that modifying and extending the simulator requires programming and debugging in both languages.

NS can simulate the following:

1. Topology: Wired, wireless
2. Scheduling Algorithms: RED, Drop Tail,
3. Transport Protocols: TCP, UDP
4. Routing: Static and dynamic routing
5. Application: FTP, HTTP, Telnet, Traffic generators

IV MODULE DESCRIPTION

Mobility Prediction Rule

This rule adapts the beacon generation rate to the frequency with which the nodes change the characteristics that govern their motion (velocity and heading). The motion characteristics are included in the beacons broadcast to a node's neighbors. The neighbors can then track the node's motion using simple linear motion equations. Nodes that frequently change their motion need to frequently update their neighbors, since their locations are changing dynamically.

On the contrary, nodes which move slowly do not need to send frequent updates. A periodic beacon update policy cannot satisfy both these requirements simultaneously, since a small update interval will be wasteful for slow nodes, whereas a larger update interval will lead to inaccurate position information for the highly mobile nodes. In this scheme, upon receiving a beacon update from a node i, each of its neighbor's records node i's current position and velocity and periodically track node i's location using a simple prediction scheme based on linear kinematics (discussed below). Based on this position estimate, the neighbors can check whether node i is still within their transmission range and update their neighbor list accordingly. The goal of the MP rule is to send the next beacon update from node i when the error between the predicted location in the neighbors of i and node i's actual location is greater than an acceptable threshold.

A simple location prediction scheme based on the physics of motion to estimate a node's current location. Note that, in that discussion, assumes that the nodes are located in a 2D coordinate system with the location indicated by the x and y coordinates. However, this scheme can be easily extended to a 3D coordinate system.

Given the position of node i and its velocity along the x and y axes at time Tl, its neighbors can estimate the current position of i, by using the following equations:

$$X_p^i = X_l^i + (T_c - T_l) *$$

$$Y_p^i = Y_l^i + (T_c - T_l) *$$

Note that, here $(X_l^i, Y$and $(V_x^i, Y$ refers to the location and velocity information that was broadcast in the previous beacon from node i. Node i uses the same prediction scheme to keep track of its predicted location among its neighbors. Let (Xa, Ya), denote the actual location of node i, obtained via GPS or other localization techniques. Node I then computes the deviation Di devi as follows:

$$D^i_{devi} = \sqrt{(X^i_a + X^i_p)^2 + (Y^i_a + Y^i_p}$$

If the deviation is greater than a certain threshold, known as the Acceptable Error Range (AER), it acts as a trigger for node i to broadcast its current location and velocity as a new beacon. The MP rule, thus, tries to maximize the effective duration of each beacon, by broadcasting a beacon only when the predicted position information based on the previous beacon becomes inaccurate. This extends the effective duration of the beacon for nodes with low mobility, thus reducing the number of beacons. Further, highly mobile nodes can broadcast frequent beacons to ensure that their neighbors are aware of the rapidly changing topology.

On-Demand Learning Rule

It is necessary to devise a mechanism, which will maintain a more accurate local topology in those regions of the network where significant data forwarding activities are on-going. This is precisely what the On-Demand Learning rule aims to achieve. As the name suggests, a node broadcasts beacons on-demand, i.e., in response to data forwarding activities that occur in the vicinity of that node.

According to this rule, whenever a node ove (5 a data transmission from a new neighbor, it broadcasts a beacon as a response. By a new neighbor, it imply a neighbor who is not contained in the neighbor list of this node. In reality, a node waits for a small random time interval before responding with the beacon to prevent collisions with other beacons. Recall that,it have assumed that the location updates are piggybacked on the data packets and that all nodes operate in the promiscuous mode, which allows them to overhear all data packets transmitted in their vicinity. In addition, since the data packet contains the location of the final destination, any node that overhears a data packet also checks its current location and determines if the destination is within its transmission range. If so, the destination node is added to the list of neighboring nodes, if it is not already present. Note that, this particular check incurs zero cost, i.e., no beacons need to be transmitted.

The neighbor list developed at a node by virtue of the initialization phase and the MP rule as the basic list. This list is mainly updated in response to the mobility of the node and its neighbors. The ODL rule allows active nodes that are involved in data forwarding to enrich their local topology beyond this basic set. In other words, a rich neighbor list is maintained at the nodes located in the regions of high traffic load. Thus, the rich list is maintained only at the active nodes and is built reactively in response to the network traffic. All inactive nodes simply maintain the basic neighbor list. By maintaining a rich neighbor list along the forwarding path, ODL ensures that in situations where the nodes involved in data forwarding are highly mobile, alternate routes can be easily established without incurring additional delays.

Adaptive position update protocol

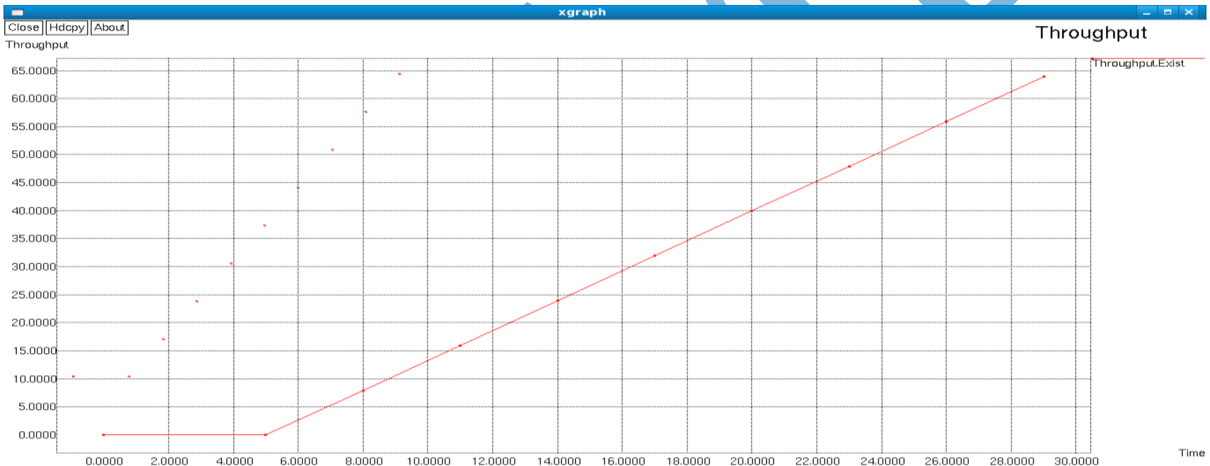
The performance of the proposed beaconing strategy, APU. It focus on two key performance measures: 1) update cost and 2) local topology accuracy. The former is measured as the total number of beacon broadcast packets transmitted in the network. The latter is collectively measured by the following two metrics: Unknown neighbor ratio. This is defined as the ratio of the new neighbors a node is not aware of, but that are within the radio range of the node to the total number of neighbors.

False neighbor ratio. This is defined as the ratio of obsolete neighbors that are in the neighbor list of a node, but have already moved out of the node's radio range to the total number of neighbors. The unknown neighbors of a node are the new neighbors that have moved in to the radio range of this node but have not yet been discovered and are hence absent from the node's neighbor table. On the other hand, false neighbors of a node are the neighbors that exist in the node's neighbor table but have actually moved out from the node's radio range (i.e., these nodes are no longer reachable).

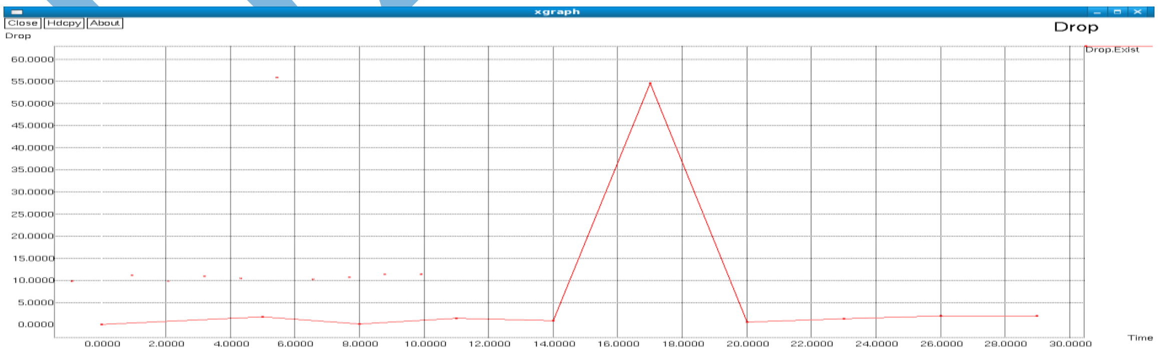
Note that, the existence of both unknown and false neighbors adversely impacts the performance of the geographic routing protocol. Unknown neighbors are ignored by a node when it makes the forwarding decision. This may lead to suboptimal routing decisions, for example, when one of the unknown neighbors is located closer to the destination than the chosen next-hop node. If a false neighbor is chosen as the next hop node, the transmitting node will repeatedly retransmit the packet without success, before realizing that the chosen node is unreachable (in 802.11 MAC, the transmitter retransmits several times before signaling a failure). Eventually, an alternate node would be chosen, but the retransmission attempts waste bandwidth and increase the delay.

V SIMULATION RESULT

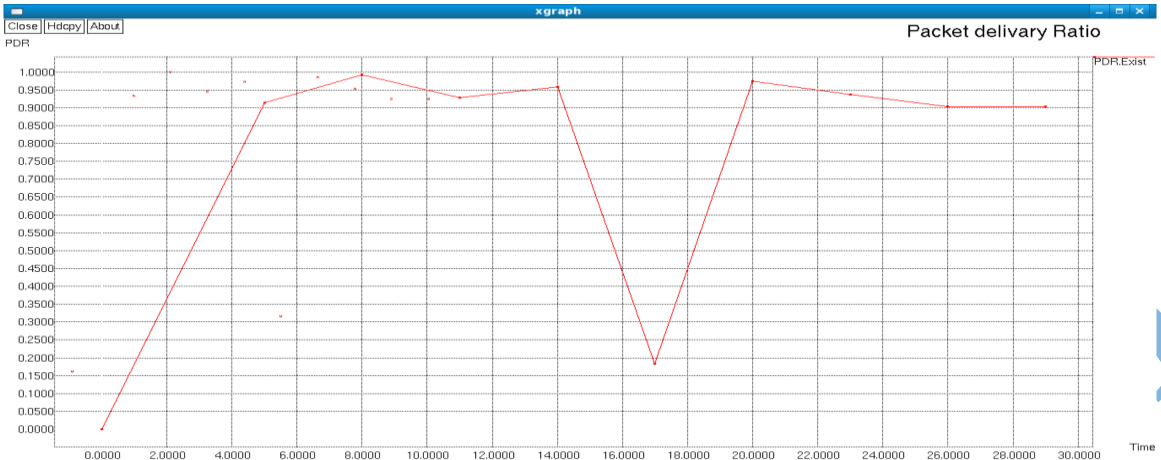
In this we present the simulation based upon the evaluation of NS2 simulator the SPU is compared with the other beacon like (i) Distance based beaconing (ii) Speed based beaconing (iii) Packet based beaconing. In the first set of simulations, we demonstrate that APU can effectively adapt the beacon transmissions to the node mobility dynamics and traffic load. In addition, we also evaluate the validity of the analytical results derived in Section 4, by comparing the same with the results from the simulations. This also include Packet delivery ratio, Energy consumption, average delay, unknown neighbor ratio, false neighbor ratio, beacon over head.



i) Performance throughput



ii) Performance Comparison packet drop



iii) Performance of PDR

V CONCLUSION

The need to adapt the beacon update policy employed in geographic routing protocols to the node mobility dynamics and the traffic load. The Adaptive Position Update strategy to address these problems has been proposed. The APU scheme employs two mutually exclusive rules. The MP rule uses mobility prediction to estimate the accuracy of the location estimate and adapts the beacon update interval accordingly, instead of using periodic beaconing. The ODL rule allows nodes along the data forwarding path to maintain an accurate view of the local topology by exchanging beacons in response to data packets that are overheard from new neighbors. Mathematically analyzed the beacon overhead and local topology accuracy of APU and validated the analytical model with the simulation results. The embedded APU within GPSR and have compared it with other related beaconing strategies using extensive NS-2 simulations for varying node speeds and traffic load. The results indicate that the APU strategy generates less or similar amount of beacon overhead as other beaconing schemes but achieve better packet delivery ratio, average end-to-end delay and energy consumption. In addition, it have simulated the performance of the proposed scheme under more realistic network scenarios, including the considerations of localization errors and a realistic physical layer radio propagation model.

VI REFERENCES

[1] Dr BalaGanesh M and Lalitha S (2009) "A Survey of Adaptive Position Update for Geographic Routing in Mobile Adhoc Networks," International Journal of Advanced Computer Technology (IJACT), ISSN: 2319-7900.

[2] Basagni S, Chlamtac I,Syrotiuk V and Woodward B (1998) "A Distance routing effect algorithm for mobility (DREAM)," in Proceedings of the Fourth Annual ACM/IEEE International Conference on mobile computing and Networking,MobiCom '98,(Dallas, Texas).

[3] Blazevic L, Giordano S, and LeBoudec J.Y (2005) "A Location Based routing method for Mobile Ad Hoc Networks," IEEE Trans. Mobile Computing, vol. 4, no. 2, pp. 97-110.

[4] Blum B, He T, Son S, and Stankovic J (2003) "IGF: A State-Free Robust communication Protocol for Wireless Sensor Networks," Technical report, Dept. of computer Science, Univ. of Virginia.

[5] Chen Q, Kanhere S.S, Hassan M, and Lan K.C (2006) "Adaptive Position Update in Geographic Routing," Proc. Int'l Conf. Comm. (ICC '06), pp. 4046-4051.

[6] Feeney L.M and Nilsson M (2001) "Investigating the Energy Consumption of a Wireless Network Interface in an Ad Hoc Networking Environment," Proc. IEEE INFOCOM, pp. 1548- 1557.

[7] Geetam S. Tomar (2010) "Position Based Routing for Wireless Mobile Ad Hoc Networks," Proc. Int'l Conf. Comm. (ICC '08), pp. 4056-4051.

[8] Heissenbuttel M and Torsten Braun (2004) "BLR: Beacon-Less Routing Algorithm for Mobile Ad-Hoc Networks," Computer Comm., vol. 27, pp. 1076- 1086.

[9] Karp B and Kung H.T (2000) "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," Proc. ACM MobiCom, pp. 243- 254.

[10] Ko Y and Vaidya N.H (2002) "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," ACM/Baltzer Wireless Networks, vol. 6, no. 4, pp. 307-321.

[11] Lee S, Bhattacharjee B, and Banerjee S (2005) "Efficient Geographic Routing in Multihop Wireless Networks," Proc. ACM MobiHoc, pp. 230-241.

[12] Misbah Jadoon, Sajjad Madani, Khizar Hayat, and Stefan Mahlknecht (2012) "Location and Non-Location Based Ad-Hoc Routing Protocols under Various Mobility Models: A Comparative Study," The International Arab Journal of Information Technology, Vol.9, No.5.

[13] Quanjun Chen, and Salil S. Kanhere (2014) "Adaptive Position Update for Geographic Routing in Mobile Ad Hoc Networks," IEEE Transactions on Mobile Computing, Vol. 12, No. 3.

[14] Rao A, Ratnasamy S, Papadimitriou C, Shenker S, and Stoica I (2005) "Geographic Routing without Location Information," Proc. ACM MobiCom, pp. 96-108.

[15] Zhou G, He T, Krishnamurthy S and Stankovic J.A (2004) "Impact of Radio Irregularity on Wireless Sensor Networks," Proc. ACM MobiSys, pp. 125-138.

[16] Zorzi M and Rao R (2004) "Geographic Random Forwarding (GeRaF) for Ad -hoc and Sensor Networks: Energy and Latency Performance," IEEE Trans. Mobile Computing, vol. 2, no. 4, pp. 349-365.