International Journal of Electronic Networks, Devices and Fields.
ISSN 0974-2182 Volume 6, Number 1 (2014), pp. 33-48

http://www.irphouse.com

Energy Efficient Geographic Routing with Apu using Gpsr in Mobile Ad-Hoc Networks

Mr. Rushikesh Shrikant Vathare[1], Dr. Lalita S. Admuthe[2], Mr. Mahesh K. Patil
Mr.Santosh P. Salgar and Mr. Prashnat Zirmite

[1]Assistant Professor in E&TC Department, at Sanjay Ghodawat Institutions, Atigre, India
[2]Vice-Principal (Academic), Professor in Electronics Engg. and Head (Electronics Engg. Dept.) at D.K.T.E.S Textile & Engineering Institute, Ichalkaranji, India
E-mail: [1]rushikesh.vathare18@gamil.com, [2]ladmuthe@gmail.com

Abstract

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.

Index Terms: Wireless communication, algorithm/ protocol design and analysis, routing protocols, Mobile Ad-hoc Network, Geographic Routing, GPSR Protocol, APU beaconing strategy.

1. Introduction

With the growing popularity of positioning devices (e.g., GPS) and other localization schemes [1], geographic routing protocols are becoming an attractive choice for use in mobile ad hoc networks. The underlying principle used in these protocols involves selecting the next routing hop from among a node's neighbors, which is geographically closest to the destination. Since the forwarding decision is based entirely on local knowledge, it obviates the need to create and maintain routes for each destination. By virtue of these characteristics, position-based routing protocols are highly scalable and particularly robust to frequent changes in the network topology. Furthermore, since the forwarding decision is made on the fly, each node always selects the optimal next hop based on the most current topology. Several studies have shown that these routing protocols offer significant performance improvements over topology-based routing Protocols such as DSR [6] and AODV [7]. The forwarding strategy employed in the aforementioned geographic routing protocols requires the following information: 1) the position of the final destination of the packet and 2) the position of a node's neighbors. The former can be obtained by querying a location service such as the Grid Location System (GLS) [8] or Quorum [9]. To obtain the latter, each node exchanges its own location information (obtained using GPS or the localization schemes discussed in [1]) with its neighboring nodes. This allows each node to build a local map of the nodes within its vicinity, often referred to as the local topology.

However, in situations where nodes are mobile or when nodes often switch off and on, the local topology rarely remains static. Hence, it is necessary that each node broadcasts its updated location information to all of its neighbors. These location update packets are usually referred to as beacons. In most geographic routing protocols (e.g., GPSR [2], [10], [11]), beacons are broadcast periodically for maintaining an accurate neighbor list at each node.

Position updates are costly in many ways. Each update consumes node energy, wireless bandwidth, and increases the risk of packet collision at the medium access control (MAC) layer. Packet collisions cause packet loss which in turn affects the routing performance due to decreased accuracy in determining the correct local topology (a lost beacon broadcast is not retransmitted). A lost data packet does get retransmitted, but at the expense of increased end-to-end delay. Clearly, given the cost associated with transmitting beacons, it makes sense to adapt the frequency of beacon updates to the node mobility and the traffic conditions within the network, rather than employing a static periodic update policy. For example, if certain nodes are frequently changing their mobility characteristics (speed and/or heading), it makes sense to frequently broadcast their updated position. However, for nodes that do not exhibit significant dynamism, periodic broadcasting of beacons is wasteful. Further, if only a small percentage of the nodes are involved in forwarding packets, it is unnecessary for nodes which are located far away from the forwarding path to employ periodic beaconing because these updates are not useful for forwarding the current traffic.

In this paper, we propose a novel beaconing strategy for geographic routing protocols called Adaptive Position Up-dates strategy (APU) [12]. Our scheme eliminates the draw-backs of periodic beaconing by adapting to the system variations. APU incorporates two rules for triggering the beacon update process. The first rule, referred as Mobility Prediction (MP), uses a simple mobility prediction scheme to estimate when the location information broadcast in the previous beacon becomes inaccurate. The next beacon is broadcast only if the predicted error in the location estimate is greater than a certain threshold, thus tuning the update frequency to the dynamism inherent in the node's motion.

The second rule, referred as On-Demand Learning (ODL), aims at improving the accuracy of the topology along the routing paths between the communicating nodes. ODL uses an on-demand learning strategy, whereby a node broadcasts beacons when it overhears the transmission of a data packet from a new neighbor in its vicinity. This ensures that nodes involved in forwarding data packets maintain a more up-to-date view of the local topology. On the contrary, nodes that are not in the vicinity of the forwarding path are unaffected by this rule and do not broadcast beacons very frequently.

We model APU to quantify the beacon overhead and the local topology accuracy. The local topology accuracy is measured by two metrics, unknown neighbor ratio and false neighbor ratio. The former measures the percentage of new neighbors a forwarding node is unaware of but that are actually within the radio range of the forwarding node. On the contrary, the latter represents the percentage of obsolete neighbors that are in the neighbor list of a node, but have already moved out of the node's radio range. Our analytical results are validated by extensive simulations.

In the first set of simulations, we evaluate the impact of varying the mobility dynamics and traffic load on the performance of APU and also compare it with periodic beaconing and two recently proposed updating schemes: distance-based and speed-based beaconing (SB) [13]. The simulation results show that APU can adapt to mobility and traffic load well. For each dynamic case, APU generates less or similar amount of beacon overhead as other beaconing schemes but achieve better performance in terms of packet delivery ratio, average end-to-end delay and energy consumption. In the second set of simulations, we evaluate the performance of APU under the consideration of several real-world effects such as a realistic radio propagation model and localization errors. The extensive simulation results confirm the superiority of our proposed scheme over other schemes. The main reason for all these improvements in APU is that beacons generated in APU are more concentrated along the routing paths, while the beacons in all other schemes are more scattered in the whole network. As a result, in APU, the nodes located in the hotspots, which are responsible for forwarding most of the data traffic in the network have an up-to-date view of their local topology, thus resulting in improved performance.

2. Related Work

In geographic routing, the forwarding decision at each node is based on the locations of the node’s one-hop neighbors and location of the packet destination as well. A forwarding nodes therefore needs to maintain these two types of locations. Many works, e.g., GLS [8], Quorum System [9], have been proposed to discover and maintain the location of destination. However, the maintenance of one-hop neighbors’ location has been often neglected. Some geo-graphic routing schemes, e.g., [14], [15], simply assume that a forwarding node knows the location of its neighbors. While others, e.g., [2], [10], [11], use periodical beacon broadcasting to exchange neighbors’ locations. In the periodic beaconing scheme, each node broadcasts a beacon with a fixed beacon interval. If a node does not hear any beacon from a neighbor for a certain time interval, called neighbor time-out interval, the node considers this neighbor has moved out of the radio range and removes the outdated neighbor from its neighbor list. The neighbor time-out interval often is multiple times of the beacon interval.

Heissenbuttel et al. [13] have shown that periodic beaconing can cause the inaccurate local topologies in highly mobile ad-hoc networks, which leads to performances degradation, e.g., frequent packet loss and longer delay. The authors discuss that the outdated entries in the neighbor list is the major source that decreases the performance. They proposed several simple optimizations that adapt beacon interval to node mobility or traffic load, including distance-based beaconing (DB), speed-based beaconing and reactive beaconing. We discuss these three schemes in the following.

In the distance-based beaconing, a node transmits a beacon when it has moved a given distance d. The node removes an outdated neighbor if the node does not hear any beacons from the neighbor while the node has moved more than k-times the distance d, or after a maximum time out of 5 s. This approach therefore is adaptive to the node mobility, e.g., a faster moving node sends beacons more frequently and vice versa. However, this approach has two problems. First, a slow node may have many outdated neighbors in its neighbor list since the neighbor time-out interval at the slow node is longer. Second, when a fast moved node passes by a slow node, the fast node may not detect the slow node due the infrequent beaconing of the slow node, which reduces the perceived network connectivity.

In the speed-based beaconing, the beacon interval is dependent on the node speed. A node determines its beacon interval from a predefined range ½a; b& with the exact value chosen being inversely proportional to its speed. The neighbor time-out interval of a node is a multiple k of its beacon interval. Nodes piggyback their neighbor time-out interval in the beacons. A receiving node compares the piggybacked time-out interval with its own time-out interval, and selects the smaller one as the time-out interval for this neighbor. In this way, a slow node can have short time-out interval for its fast neighbor and therefore eliminate the first problem presented in the distance-based beaconing. However, the speed-based beaconing still suffer the problem that a fast node may not detect the slow nodes.

In reactive beaconing, the beacon generation is triggered by data packet transmissions. When a node has a packet to transmit, the node first broadcasts a beacon request packet. The neighbors overhearing the request packet respond with beacons. Thus, the node can build an accurate local topology before the data transmission. However, this process is initiated prior to each data transmission, which can lead to excessive beacon broadcasts, particularly when the traffic load in the network is high.

The APU strategy proposed in this work dynamically adjusts the beacon update intervals based on the mobility dynamics of the nodes and the forwarding patterns in the network. The beacons transmitted by the nodes contain their current position and speed. Nodes estimate their positions periodically by employing linear kinematic equations based on the parameters announced in the last announced beacon. If the predicted location is different from the actual location, a new beacon is broadcast to inform the neighbors about changes in the node's mobility characteristics. Note that, an accurate representation of the local topology is particularly desired at those nodes that are responsible for forwarding packets. Hence, APU seeks to increase the frequency of beacon updates at those nodes that overhear data packet transmissions. As a result, nodes involved in forwarding packets can build an enriched view of the local topology.

There also exist some geographic routing protocols that do not need to maintain the neighbor list and therefore can avoid position updates, e.g., IGF [16], GeRaf [17], BLR [18], ALBA-R [19]. These protocols are commonly referred to as beaconless routing protocols. The main ideal is that, the forwarding node broadcasts the data packet to all its neighbors who then distributed decide which node relays the packet. Normally, in these protocols, after receiving a packet, each neighbor sets a timer for relaying the packet based on some metrics, e.g., the distance to the destination. The neighbor that has the smallest timer will expire first and relay the packet. By overhearing the relayed packet, other neighbors can cancel their own timers and ensure that no duplicate packet is transmitted. Hence, the beaconless routing protocols can avoid excessive position updates and are particular suitable for networks where the topology is highly dynamic, e.g., in wireless sensor network where nodes periodically switch on and off (to save energy consumption)

3. Block Diagram

In this system we have considered only 15 nodes like that we can use N number of nodes. The protocols that we propose here are GPSR and assume that all the nodes are identical in their physical characteristics and all communicate via wireless channel. We can define any node as source or as destination. The fields in beacon packet are shown in above fig. which is Source address, Destination address etc. Implementation & simulation of proposed work is conducted in NS-2[1].

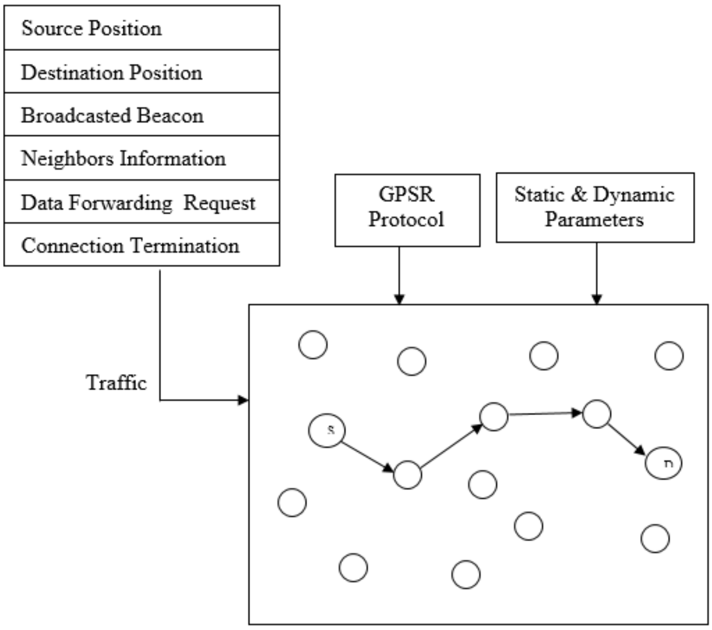


Fig. 1: Conceptual block diagram.

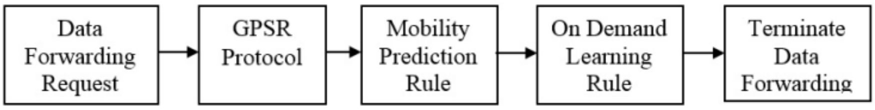


Fig. 2: General block diagram.

In the block diagram the proposed APU beaconing strategy using GPSR routing protocol will be implemented using following rules,

1. Mobility Prediction Rule
2. On Demand Learning Rule

When source node want to forward data to destination, then source generates data forwarding request packet. According to updated network scenario nodes which are in between the forwarding path helps to forward data by choosing shortest path.

4. Adaptive Position Update

Adaptive Position Update (APU) beaconing strategy for geographic routing, which dynamically adjusts the frequency of position updates (beacons) 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

Nodes closer to forwarding paths update their positions more frequently.

According to classical nature of geographic routing following assumptions required in our work:

1. All nodes are aware of their own position and velocity,
2. All links are bidirectional,
3. The beacon updates include the current location and velocity of the nodes
4. Data packets can piggyback position and velocity updates and all one-hop neighbors operate in the promiscuous mode and hence can overhear the data packets.

The beacons (position update) play an important part in maintaining an accurate representation of the local topology. Instead of periodic beaconing, APU adapts the beacon update intervals to the mobility dynamics of the nodes and the amount of data being forwarded in the neighborhood of the nodes. APU employs two beacon triggering rules, which are as follows: 1) MP Rule, 2) ODL Rule

4.1 Mobility Prediction (MP) Rule

The MP rule [1][2] uses mobility prediction to estimate the accuracy of the location estimate and adapts the beacon update interval accordingly, instead of using periodic beaconing. Neighbors can then track the node's motion using simple linear motion equations. 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.

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

$$X_p^i = X_l^i + (T_c - T_l) * V_x^i \; Y_p^i = Y_l^i + (T_c - T_l) * V_y^i \qquad (1)$$

Where $(X_{l,}^i \; Y_l^i)$ is the coordinate of node i at time T_l (included in the previous beacon),(V_x^i, V_y^i) is the velocity of node i along the direction x & y axes at time T_l(included in the previous beacon) , T_l is the time of the last beacon broadcast, T_c is the current time, (X_p^i, Y_p^i) is the predicted position of node I at the current time. $(X_{l,}^i \; Y_l^i)\&(V_x^i, V_y^i)$ 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 (X_a, Y_a) denote the actual location of node i, obtained via GPS or other localization techniques. Node I then computes the deviation as follows

$$D_{devi}^i = \sqrt{(X_a^i - X_p^i)^2 + (Y_a^i - Y_p^i)^2} \qquad (2)$$

If the deviation is greater than a certain threshold, known as the Acceptable Error Range (AER), acts as a trigger for node i to broadcast its current location and velocity as a new beacon.

4.2 On Demand Learning (ODL) Rule:

The ODL rule[3][4] 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. Local topology will not be updated and they will exclude each other while selecting the next hop node. In the worst case, assuming no other nodes were in the vicinity, the data packets would not be transmitted at all. Hence, 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 overhears a data transmission from a new neighbor, it broadcasts a beacon as a response.

In essence, ODL aims at improving the accuracy of topology along the routing path from the source to the destination, for each traffic flow within the network.

The MP rule solely may not be sufficient for maintaining an accurate local topology. Consider the example illustrated in Fig. 2, where node A moves from P 1 to P 2 at a constant velocity. Now, assume that node A has just sent a beacon while at P 1. Since node B did not receive this packet, it is unaware of the existence of node A. Further, assume that the AER is sufficiently large such that when node A moves from P 1 to P 2, the MP rule is never triggered. However, as seen in Fig. 2 node A is within the communication range of B for a significant portion of its motion. Even then, neither A nor B will be aware of each other. Now, in situations where neither of these nodes are transmitting data packets, this is perfectly fine since they are not within communicating range once A reaches P 2. However, if either A or B was transmitting data packets, then their local topology will not be updated and they will exclude each other while selecting the next hop node. In the worst case, assuming no other nodes were in the vicinity, the data packets would not be transmitted at all.

Hence, 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 overhears a data transmission from a new neighbor, it broadcasts a beacon as a response. By a new neighbor, we 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, we 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.

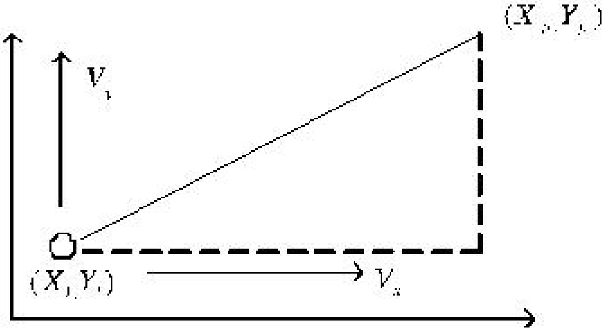


Fig. 3: Example of Mobility Prediction.

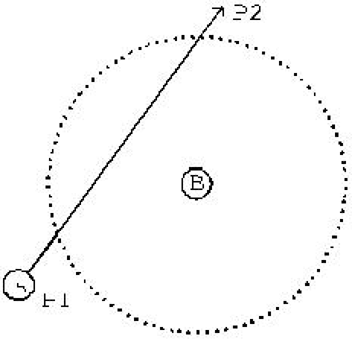


Fig. 4: Example illustrating drawback of MP Rule.

We refer to 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.

Fig.5 illustrates the network topology before node A starts sending data to node P . The solid lines in the figure denote that both ends of the link are aware of each other. The initial possible routing path from A to P is A-B-P. Now, when source A sends a data packets to B, both C and D receive the data packet from A. As A is a new neighbor of C and D, according to the ODL rule, both C and D will send back beacons to A. As a result, the links AC and AD will be discovered. Further, based on the location of the destination and their current locations, C and D discover that the destination P is within their one-hop neighborhood. Similarly, when B forwards the data packet to P , the links BC and BD are discovered. Fig. 3b reflects the enriched topology along the routing path from A to P .

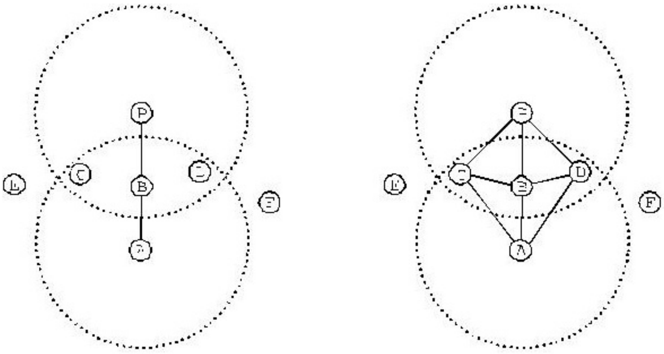


Fig. 5: An example illustrating the ODL rule.

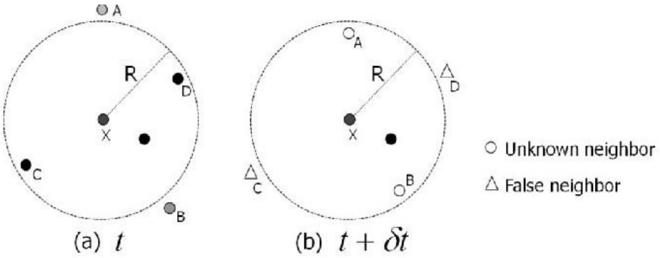


Fig. 6: Example illustrating unknown and false neighbors

Note that, though E and F receive the beacons from C and D, respectively, neither of them respond back with a beacon. Since E and F do not lie on the forwarding path, it is futile for them to send beacon updates in response to the broadcasts from C and D. In essence, ODL aims at improving the accuracy of topology along the routing path from the source to the destination, for each traffic flow within the network.

4.3 Specifications of GPSR protocol:

GPSR [4] protocol aims for scalability with increase in number of nodes in the network & increasing Mobility. GPSR Beacon broadcasts MAC address, containing Owner IP & position. Position is encoded as two 4-byte floating point for X&Y co-ordinates values[5]

Packet Header Fields in Perimeter-mode: (Destination address, Location where packet entered in perimeter mode, packet mode- Greedy or Perimeter, etc…)

GPSR implementation contains two modules:-i) GPSR daemon, ii) API (Application Programming Interface)

Consists of two methods:- a) Greedy forwarding. b) Perimeter forwarding.

5. Implementation Steps

This work is divided into two modules:

A. Implementation of classical geographic routing
B. Implementation of APU strategy for geographic routing.

Steps of implementation of above two modules in detail as follows:

A. Classical geographic routing implementation steps::

i. Each node broadcasts a beacon informing its neighbors about its presence and its current location and velocity. In proposed work GPSR routing protocol is used.
ii. The position information received from neighboring beacons is stored at each node.
iii. iii. Based on the position updates received from its neighbors, each node continuously updates its local topology and neighbor list.
iv. Only those nodes from the neighbor list are considered as possible candidates for data forwarding.

B. APU for geographic routing implementation steps:

i. Program all nodes using proper specifications.
ii. Broadcast beacon according to GPSR protocol.
iii. By using MP rule tune the frequency of beacon broadcasting and using ODL rule update neighbor list and network topology.
iv. If data forwarding request will come then send this data via shortest possible path.
v. GPSR will helps reduce beacon overheads and maintain updated network topology.
vi. .After completing data forwarding terminate the connection.

6. Simulation Results

In this section, we present a comprehensive simulation-based evaluation of APU using the popular NS-2 simulator. We compare the performance of APU with other beaconing schemes. These include PB and two other recently proposed adaptive beaconing schemes in [13]: (i) Distance-based Beaconing and (ii) Speed-based Beaconing

Table 1: Energy Consumption in Each Operation.

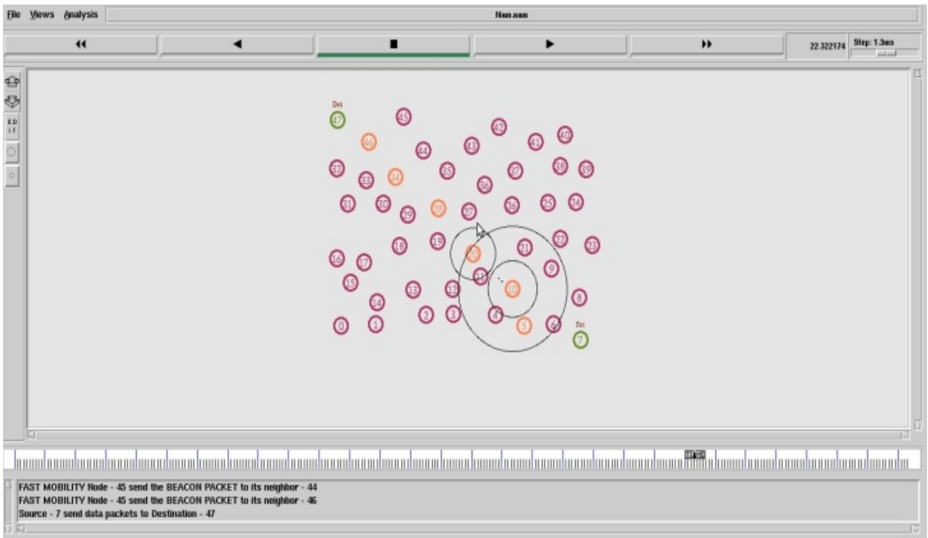
Operation	μW· sec/byte[8]	μW· sec
point-to-point send	0.48 ×size	+431
broadcast send	2.1 ×size	+272
point-to-point recv	0.12 ×size	+316
broadcast recv	0.26 ×size	+50
promiscuous recv	0.12 ×size	+83
promiscuous discard	0.11 ×size	+54

(The point-to-point communication uses date rate of 11 Mbps. The broadcasting uses data rate of 2 Mbps. Therefore, broadcasting costs more energy than point-to-point sending)

We conduct three sets of experiments. 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. In the second set of experiments, we consider the impact of real-world factors such as localization errors, realistic radio propagation, and sparse density of the network on the performance of APU. In the third set of experiments, we evaluate the impact of parameter AER (which is from MP component) on the overall performance of APU. This enables us to investigate which component (MP or ODL) contributes to the performance more significantly.

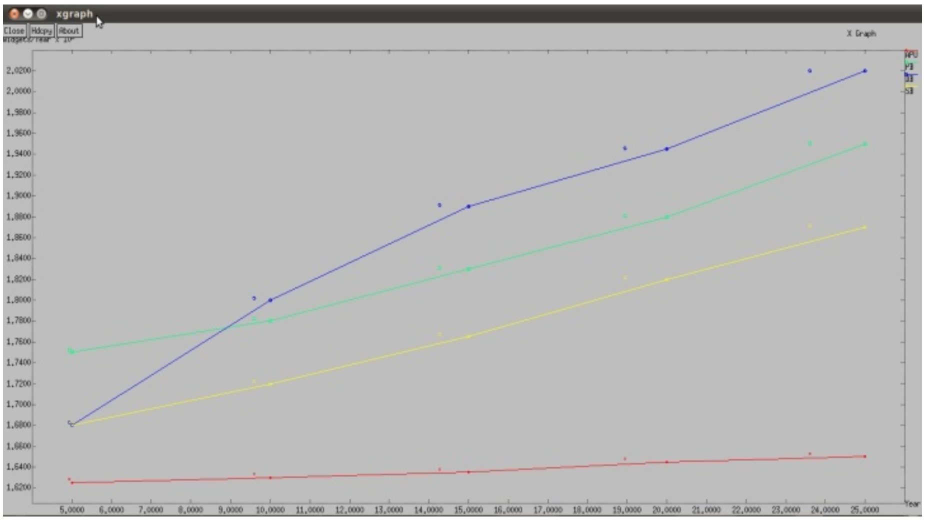
We use two sets of metrics for the evaluations. The first set includes the metrics used in our analysis, viz., beacon overhead and local topology accuracy (false and unknown neighbor ratio), which directly reflect the performance achieved by the beaconing scheme. Note that the beaconing strategies are an integral part of geographic routing protocols. The second set of metrics seek to evaluate the impact of the beaconing strategy on the routing performance. These include: 1) packet delivery ratio, which is measured as the ratio of the packets delivered to the destinations to those generated by all senders, 2) average end-to-end delay incurred by the data packets, and 3) energy consumption, which measures the total energy consumed in the network. We adopt the widely used energy consumption model, which estimates the energy consumption for each basic operation (e.g., transmitting, receiving, and over-hearing in promiscuous mode) based on empirical data collected from commercial wireless cards. The energy consumption for each radio operation is listed in Table 2. We also measured the average hop count traversed by the packets. However, we found that this metric is not an effective tool for comparing beaconing schemes (please refer to our technical report for the details). In the simulations, we have implemented GPSR [2] as an illustrative example of a geographic routing protocol. We simulate IEEE 802.11b as the MAC protocol with wireless bandwidth of 11 Mbps and assume a two-ray ground propagation model unless otherwise stated.

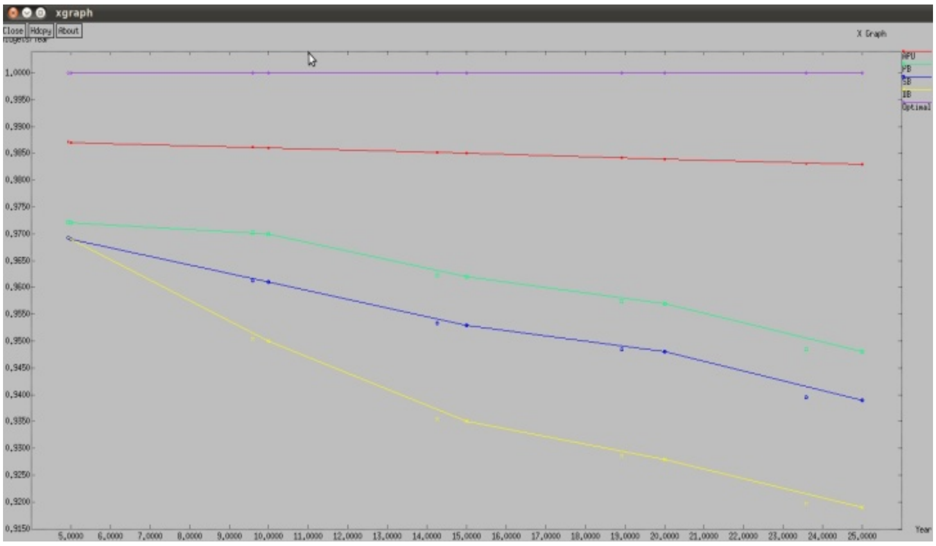
The results of simulation are as follows

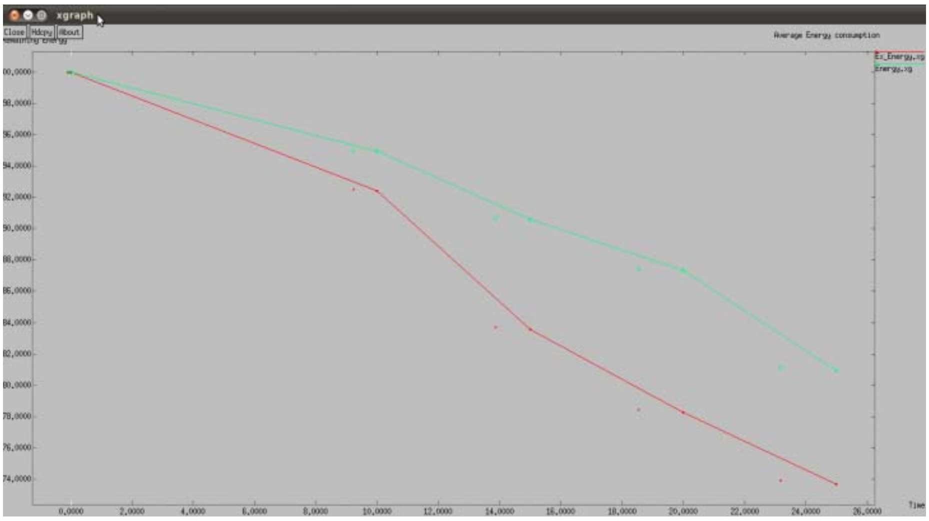


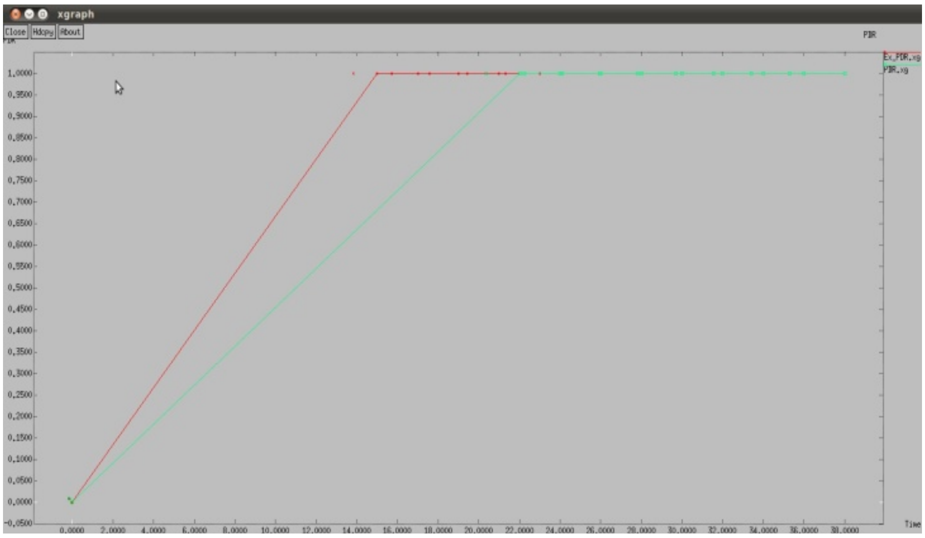
```
FAST MOBILITY Node - 45 send the BEACON PACKET to its neighbor - 44
FAST MOBILITY Node - 45 send the BEACON PACKET to its neighbor - 46
Source - 7 send data packets to Destination - 47
```

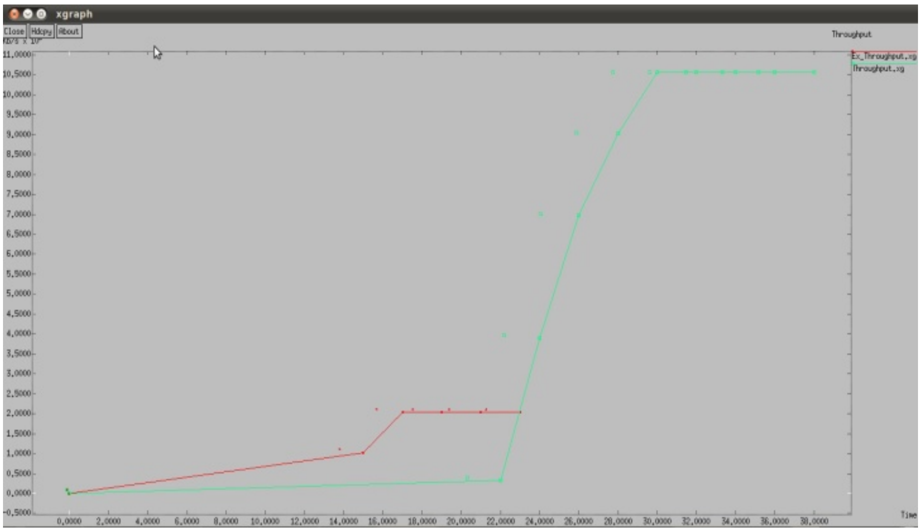
```
FAST MOBILITY Node - 45 send the BEACON PACKET to its neighbor - 44
FAST MOBILITY Node - 45 send the BEACON PACKET to its neighbor - 46
Source - 7 send data packets to Destination - 47
```






xgraph
Close
Hdcpy
About
Remaining Energy
Average Energy consumption
Ex_Energy.xg
Energy.xg
100,0000
98,0000
96,0000
94,0000
92,0000
90,0000
88,0000
86,0000
84,0000
82,0000
80,0000
78,0000
76,0000
74,0000
0,0000
2,0000
4,0000
6,0000
8,0000
10,0000
12,0000
14,0000
16,0000
18,0000
20,0000
22,0000
24,0000
26,0000
Time


xgraph
Close
Hdcpy
About
PDR
PDR
Ex_PDR.xg
PDR.xg
1,0000
0,9500
0,9000
0,8500
0,8000
0,7500
0,7000
0,6500
0,6000
0,5500
0,5000
0,4500
0,4000
0,3500
0,3000
0,2500
0,2000
0,1500
0,1000
0,0500
0,0000
-0,0500
0,0000
2,0000
4,0000
6,0000
8,0000
10,0000
12,0000
14,0000
16,0000
18,0000
20,0000
22,0000
24,0000
26,0000
28,0000
30,0000
32,0000
34,0000
36,0000
38,0000
Time


xgraph
Close
Hdcpy
About
Kb/s x 10³
Throughput
Ex_Throughput.xg
Throughput.xg
11,0000
10,5000
10,0000
9,5000
9,0000
8,5000
8,0000
7,5000
7,0000
6,5000
6,0000
5,5000
5,0000
4,5000
4,0000
3,5000
3,0000
2,5000
2,0000
1,5000
1,0000
0,5000
0,0000
-0,5000
0,0000
2,0000
4,0000
6,0000
8,0000
10,0000
12,0000
14,0000
16,0000
18,0000
20,0000
22,0000
24,0000
26,0000
28,0000
30,0000
32,0000
34,0000
36,0000
38,0000
Time

7. Conclusion

In this paper, we have identified the need to adapt the beacon update policy employed in geographic routing protocols to the node mobility dynamics and the traffic load. We proposed the Adaptive Position Update strategy to address these problems. 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. We mathematically analyzed the beacon over-head and local topology accuracy of APU and validated the analytical model with the simulation results. We have 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. Our 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, we 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. Future work includes utilizing the analytical model to find the optimal protocol parameters (e.g., the optimal radio range), studying how the proposed scheme can be used to achieve load balance and evaluating the performance of the proposed scheme on TCP connections in Mobile Ad hoc Networks.

References

[1] Quanjun Chen, Salil S. Kanhere and Mahbub Hassan, “Adaptive Position Update for Geographic Routing in Mobile Ad Hoc Networks”, IEEE Transactions On Mobile Computing, Vol. 12, No. 3, MARCH 2013

[2] Xiaojing Xiang, Xin Wang and Zehua Zhou, “Self-Adaptive On-Demand Geographic Routing for Mobile Ad Hoc Networks”, IEEE Transactions Mobile Computing, Vol. 11, No. 9, SEPTEMBER 2012

[3] L. Blazevic, S. Giordano, and J.-Y. LeBoudec, “A Location Based Routing Method for Mobile Ad Hoc Networks,” IEEE Trans. Mobile Computing, Vol. 4, No. 2, pp. 97-110, MARCH 2005.

[4] B. Karp and H.T. Kung, “GPSR: Greedy Perimeter Stateless Routing for Wireless Networks,” Proc. ACM MobiCom, pp. 243-254, AUGUST 2000.

[5] J. Hightower and G. Borriello, “Location Systems for Ubiquitous Computing,” Computer INFOCOM, vol. 34, no. 8, pp. 57-66, AUGUST 2001.

[6] Zygmunt J. Haas and B. Liang, “Ad Hoc Mobility Management with Uniform Quorum Systems” IEEE/ACM Trans. Networking, vol. 7, no. 2, pp. 228-240, APRIL 1999.

[7] Joo-Han Song, Student, IEEE, Vincent W. S. Wong, Member, IEEE, and Victor C. M. Leung, Fellow, IEEE “Efficient On-Demand Routing for Mobile Ad Hoc Wireless Access Networks”, IEEE Journal In Communications, Vol. 22, No. 7, SEPTEMBER 2004

[8] Chih-Hsun Chou, Kuo-Feng Ssu, Member and Hewijin Christine Jiau, ”Geographic Forwarding With Dead-End Reduction in Mobile Ad Hoc Networks”, IEEE Transactions On Network Technology, Vol. 57, No. 4, JULY 2008

[9] Surendra Boppana, Student Member and John M. Shea,” Improving the Performance of Overlapped Transmission in Wireless Ad Hoc Networks IEEE Journal On Selected Areas In Communications, Vol. 29, No. 10, DECEMBER 2011

[10] Ben Liang, Member, Xin Wang and Yuanyuan Yang,”Stateless Multicasting in Mobile Ad Hoc Networks”, IEEE Transactions on Computers, VOL. 59, NO. 8, AUGUST 2010

[11] Yuanyuan Yang and Rafael Marin-Perez, ”Beacon-Less Geographic Routing Made Practical: Challenges, Design Guidelines, and Protocols”, IEEE Communications Magazine, Vol. 19, No. 5, AUGUST 2009

[12] A. Rao, S. Ratnasamy, C. Papadimitriou and I. Stoica, “Geographic Routing Without Location Information”, IEEE INFOCOM, Vol.15, pp.153-181, FEBRUARY 2007.

[13] Juan A. Sanchez and N.H. Vaidya, “Location-Aided Routing (LAR) in Mobile Ad Hoc Networks,” ACM/Baltzer Wireless Networks, Vol. 6, No. 4, 307-321, SEPTEMBER 2002.

[14] Guoxing Zhan, Weisong Shi and Julia Deng, “Design and Implementation of TARF: A Trust-Aware Routing Framework for WSNs” IEEE Transactions On Dependable And Secure Computing, Vol. 9, No. 2, MARCH/APRIL 2012

[15] Pangun Park, Piergiuseppe Di Marco, Carlo Fischione and Karl Henrik Johansson, “ Modeling and Optimization of the IEEE 802.15.4 Protocol for Reliable and Timely Communications” IEEE Transactions on Parallel and Distributed Systems,Vol.24, No. 3, MARCH 2013