PERFORMANCE TUNING FOR GEOGRAPHIC ROUTING IN WIRELESS NETWORKS

Akshay K. Gondalia*

Dr. Dhaval R. Kathiriya**

A B S T R A C T:

To accomplish fast and location error independent and energy-efficient message delivery in ad hoc and wireless sensor networks, we propose PEOGR, a simple geographic routing protocol. POEGR is reliable routing protocol in that nodes do not need to set up or maintain routing or neighbour tables; instead, POEGR routing via its ‘‘Angled Probabilistic Relaying” mechanism and uses the ‘‘Backup and relay cancellation” mechanism to reduce contention and the number of retransmissions. One of the main features of PEOGR is a message based ACK mechanism where a relayed message is used as an ACK to a previous sender.

In this paper we provide an algorithm and extensive analysis of POEGR in terms of average hop count and average delay per hop. Simulation results also show that POEGR outperforms other protocols in terms of average delay and number of transmissions per message delivery.

Keywords: Ad hoc Network, Geographic Routing, Routing protocol, Position based routing, Geo routing.

* Asst. Prof, LDRP-ITR, Sector-15 Gandhinagar (GUJ), India.

** Director (IT), AAU, Anand (GUJ), India.

1.0 INTRODUCTION:

Geographic routing (also known as position-based routing or geometric routing) is a technique to deliver a message to a node in a network over multiple hops by means of position information. Routing decisions are not based on network addresses and routing tables; instead, messages are routed towards a destination location. With knowledge of the neighbors location, each node can select the next hop neighbor that is closer to the destination, and thus advance towards the destination in each step.

Geographic routing is attractive for dynamic networks such as wireless ad hoc and sensor networks. In such networks, acquiring and maintaining routing information is costly as it involves additional message transmissions that require energy and bandwidth and frequent updates in mobile and dynamic scenarios.

2.0 RELATED WORK:

Early proposals for geographic routing were based on pure greedy approaches: a packet at an intermediate node is forwarded to the neighbour who is the closest to the destination [1, 2]. Each intermediate node applies this greedy principle until the destination is reached. However, greedy routing does not guarantee delivery even if there is a path from source to destination. The reason being that this principle fails if there is no one-hop neighbour that is closer to the destination than the forwarding node itself. Then, we look at the nature of planar graph based algorithms. Planarization techniques, such as Gabriel Graphs and Relative Neighbourhood Graphs (RNG) are based on Unit Disk Graphs (UDG). That is, all nodes must have equal transmission ranges. This is not true in real-life and especially for obstructed scenarios. The basic principle of planarization techniques fails when applied to scenarios with obstacles [5]. In other words, applying planarization techniques may disconnect a connected network with particular patterns of obstacles between nodes. In addition, asymmetric radio links and inaccurate node location information could cause errors that violate the assumption of the unit disk graph. The causes and effects of planarization failures are discussed in details in [6]. These failures motivated studying approaches that are not based on planarization such as Cross Link Detection Protocol (CLDP)

[7], which was criticized to be complex and costly in [8]. So it led to another proposal, which employs a Greedy Distributed Spanning Tree Routing (GDSTR) approach [8]. It is said that GDSTR is also robust to location errors [8]. Although it is mentioned in GDSTR that it does not require radio ranges to be uniform [8], the protocol is simulated with unit radio ranges. However, this protocol also suffers from a number of limitations that are discussed in this section. For more details, please see [8]. One other unrealistic design factor with planar graph based techniques is that they are studied in a static network at the routing layer, for example, by assuming that routing takes places faster than network changes. Implementation of the Face-2 algorithm is not done for dynamically changing networks [2]. GOAFR+ also executes a routing algorithm on temporarily stationary nodes and leaves the integration of network movement for future work. It is explicitly mentioned in GOAFR+ that it assumes routing takes place much faster than node movement [3]. Overall, it is under such unrealistic assumptions that different face routing protocols claiming to guarantee delivery of the packet to the destination [2].

Improvements are made to planarization-based algorithms such as proposals to improve the efficiency of the paths by for example identifying a better path as in Progress Face of [9] or reducing the number of hops as in VR-forwarding of [10] or the non-message less proposal by [11], which is a technique that exchanges messages to construct a planar graph. However, routing algorithms based on planarization suffer disconnection problems in sparse networks [12].

Communication overhead is an important evaluating factor, which refers to the number of messages that are required to be transmitted when handling a void [4]. Face routing has high overhead. In addition, most planar graph based techniques require periodic one-hop beacons. Flooding based protocols also incur a lot of overhead due to the nature of flooding. Among the reviewed protocols, passive participation and active exploration have low overhead. Some of the reviewed protocols have high complexity in that they are difficult to implement, or require extra resources or have complex processing. Of those with high complexity we can mention Partial Hop-by-hop Routing, Partial Source Routing, Anchored Geodesic Packet Forwarding, alternate network, active exploration and all hybrid void handling techniques.

Flooding, one-hop flooding, passive participation and void avoidance are simple. The rest are of medium complexity [4]. Cost-based protocols have relatively high complexity and overhead to maintain and adjust the cost for optimal paths [5]. In fact, cost-based void handling techniques are designed with a static network in mind. Otherwise, network performance will get worse due to too much overhead imposed by cost adjustment and maintenance.

Inaccurate location information is another issue that is worth considering in real-life scenarios. All of the reviewed protocols assume accurate position information. There are studies about the effect of inaccurate location information on greedy forwarding and how to make them more tolerant of inaccurate position information.

Overall, for our future research, we are interested in designing a light-weight, low-cost geographic routing protocol that works well and in the presence of node mobility. In other words, topology changes must be taken into account. To the best of our knowledge, there is no clear understanding of the behaviour of discussed recovery strategies when the topology of the network changes due to mobility, while a packet is being forwarded. In addition, the planar based geographic routing protocols claim to guarantee delivery of the packet to the destination in networks [2]. However, as mentioned before, these techniques are studied in a static network.

The basic idea of our proposal in-line with the goals mentioned above. The proposed solution employs a restricted direction flooding with recovery strategy. We would like to investigate the efficiency and performance of our proposed protocol by looking at different metrics such as total number of hops per route (i.e., total number of hops each packets takes to its destination), overhead, which could be measured as the total number of routing traffic sent in the entire network, end-to-end delay and packet delivery ratio (i.e., data traffic received / data traffic sent). We plan to collect the statistics by conducting simulations.

3.0 ARCHITECTURE:

We have introduced new architecture as link Measure for geographic routing in wireless networks. Geographic routing with new architecture provides an adaptive routing strategy, which is general and can be used for various link value types. We have presented techniques for link value evaluation. In these environments, the combination of new architecture and value evaluation techniques outperforms the current geographic routing scheme. A new architecture also finds paths whose value is close to the optimum. Value-aware routing schemes including benefit greatly from fast and accurate link value evaluation, and we plan to investigate this issue further in the future. We have treated each link value type independently. However, if we consider multiple interdependent values simultaneously, choosing the next hop based on one value type may not be always the best choice for other values.

3.1 Link Measure parameter:

Link value is the power utilization required for a packet transmission over the link. For example, even though two neighbors require the same power, in geographic routing we prefer the neighbor closer to the destination. The goal of this section is to propose a new architecture for geographic routing that can be generalized to various value types (i.e., Packet error charge, power utilization, link interruption).

The implicit goal of this strategy is to minimize the hop count between source and destination. Let us consider an intermediate node the amount of decrease in distance by a neighbor N, which we call Link Value (LV).

LV (N) =Dist(S, T)-Dist(S, N) [Equation 1]

Where Dist(x,t) denotes the distance from node x to T. For example, Lundgren identify gray zones [13], where due to high-error probability, nodes cannot exchange long data packets in most cases. The goal of our work is to balance the trade-off, so that we can select a neighbor with both large LV and good link quality. So we now adopt new type of LV which is:

NLV(N) = LV(N)/LC(N) [Equation 2]

3.2 Power utilization:

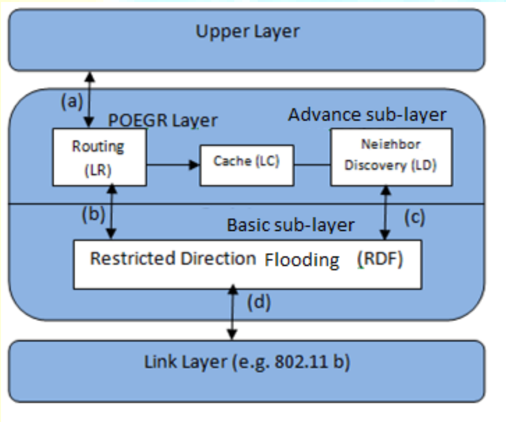
Many wireless systems have a control mechanism for transmission power adjustment to save battery and reduce interference. We assume that using such a mechanism, nodes know the appropriate transmission power level to each neighbor. Then, the sub layer can retrieve the value and calculate the actual system power consumption [16] considering additional components of power utilization.

LC (power) = Energy used to transmit.

NLV (power) = LV/ LC(power) [Equation 3]

4.0 PACKET ARCHITECTURE:

Figure shows functional components of POEGR. We divide the POEGR layer into two sublayers: basic sub-layer and advance sub-layer. The basic sub-layer supports conventional geographic routing function, restricted direction forwarding. The advance sub-layer implements two



main functions, the neighbor discovery and routing.

Figure:1 *Functional Components of POEGR*

The overall packet processing procedure of POEGR is as follows. A packet is passed from the upper layer through interface (a). LR looks up the LC and makes a decision either to initiate the neighbor discovery function or to pass the packet to RDF for delivery. If neighbor discovery is not necessary, LR passes the data packet after possibly adding a header to RDF. RDF will forward the data packet to the next hop using a geographic routing protocol. If neighbor discovery is necessary, LR triggers the discovery (LD). Initiated by LR, LD creates an FLD control packet to a specified destination node and passes it down to RDF for delivery through interface (c). RDF forwards the FLD packet as usual. When the next-hop node receives a packet through interface (d), RDF first checks if the packet is a control packet (FLD or BLD) or data packet. In the case of a control packet, RDF passes it to LD. In the case of a data packet, RDF forwards the packet to the next node unless the node is the final or intermediate target. We only need to add the simple functions such as the identification of control packets to the existing geographic routing protocols.

4.1 POEGR Packet Format:

Before describing the detailed protocol operation, let us examine the packet format of POEGR. Figure shows the generic packet structure of POEGR. A packet consists of two parts namely a RDF header and a NDR header. The RDF header is used for ordinary packet forwarding at basic sub-layer, while the NDR header is used for neighbour discovery and routing.

In the RDF header, RDF-specific optional information contains some related state information necessary for restricted direction flooding. The NDR header has a type-specific information structure.

There are three types of NDR packets: FLD, BLD and MLD. FLD and BLD and MLD specify FLD/BLD/MLD control packets, respectively. The type-specific information of FLD (Type = FLD) and BLD (Type = BLD) packets are almost the same except BLD packets additionally include the forwarding node id and relay cancellation flag(RCF) as shown in figure. Note that a FLD packet can carry the data payload when it is available. MLD (Type=MLD) packet are broadcast at a movement of mobile node greater then threshold value. MLD control packet update the routing table of node working as ad hoc router. Ad hoc router updates its routing table according to mobile nodes update.

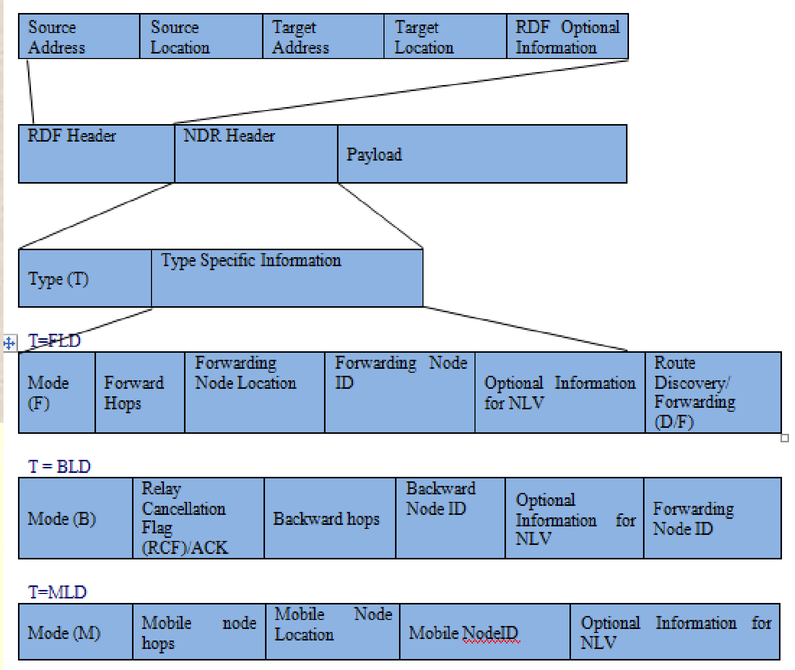


***Figure 2**: POEGR Packet Format*

4.2 Selection of Location services and Forwarding strategies:

We present the basic idea of position based addressing and routing, and gives criteria for taxonomy of the various proposals. We cover technique for location services and outline position based forwarding with a qualitative comparison and point out the open issues and possible direction.

4.2.1 Location service:

Before a packet can be sent, it is necessary to determine the position of its destination. Typically, a location service is responsible for this task. Existing location services can be classified according to how many node host the service. This can be either some specific nodes or all nodes of the network. Furthermore, each location server may maintain the position of some specific or all nodes in the network. The four possible combinations are as below:

- Some-for-some
- Some-for-all

- All-for-some
- All-for-all

In mobile ad hoc networks, a centralized approach is viable only as an external service that can be reached via non ad-hoc means. There are two main reasons for this. First, it would be difficult to obtain the position of a position server if the server were part of the ad-hoc network itself. Second, since an ad-hoc network is dynamic, it might be difficult to guarantee that at least on position server will be present in a given ad hoc network.

4.2.2 Forwarding strategy:

There are following forwarding strategies in geographic routing [17]:

- Greedy forwarding
- Restricted directional flooding
- Next-hop selection
- Recovery strategy
- Hierarchical approaches

The node holding the message is only aware of its own location, its immediate one hop neighbors, and the destination location.We concentrate on decentralized location services that are part of the ad-hoc network with restrict flooding

5.0 PROPOSED GEOGRAPHIC ROUTING: POEGR

Our contribution: In this section we proposed probabilistic geographic routing protocol (PEOGR) for ad-hoc and sensor networks. PEOGR is an energy-aware decentralized routing protocol. Our approach differs from the previous work in a number of ways. First, the nodes do not need any global knowledge of the data flow, or any clustering scheme in order to accomplish energy balancing. Secondly, POEGR probabilistically selects the next hop from a set of candidate nodes instead of the conventional deterministic approach. This helps eliminate the complexity of route selection introduced by most deterministic approaches. To our knowledge this approach has not been explored before. In the existing probabilistic schemes, each node decides whether it will relay a packet or not, and if it does, the packet is broadcasted to all the neighbors, such as in the probabilistic flooding.

We have implemented PEOGR in NS-2, and have compared its performance in terms of throughput, end-to-end delay, and network lifetime to GPSR, probabilistic flooding, PGR.

5.1 Functionality:

First we describe our assumptions throughout the paper. Each node in the network is aware of its (x,y) coordinates in the plane. The node can either be equipped with a GPS device, or use some other localization scheme, such as the signal-strength based localization [18-22]. Every node which has a packet to send, called the source, needs to know the location of the destination node.

This could be accomplished using a location database [23]. Any intermediate node that forwards the packet toward the destination does not need to know the location of the target since this information is included in the message header.

We assume the wireless links are asymmetric. The existence of asymmetric links in wireless networks has been empirically shown in [24].

5.2 Neighbor Detection and Maintenance:

At the beginning of the deployment, the nodes need to gather some initial information on who their neighbors are, and how good of a connection they have to each neighbor. The discovery time is decided at the deployment. The longer the discovery period is, the better the initial link estimation is. During the discovery phase each node sends control messages. These messages contain the geographic location of the sending node, its NLV (New Link Value), and a list of its neighbors with the corresponding NLV. These messages are used during the detection phase to estimate the link NLV from a node to all of its neighbors. We are using our Link Value parameter for evaluation of link.

At the end of the detection phase, every node has a list of all the nodes it can hear from with the corresponding link value and evaluation. For example, node i has a list of all nodes j that it can hear from, with NLVji. Lets assume that the size of the neighbor table of node i is N. At the end of the detection phase, node i picks N of its neighbors with the highest of NLV, i.e. NLVji. Note that we use the backward NLV since it corresponds to the link quality from node i to its neighbor j, i.e. how well node j hears node i. It is also possible to pick all the nodes with NLVji lower than a given threshold, if the neighbor table size is not a limitation. Every entry in the neighbor table consists of the corresponding node id, the geographical location and the corresponding link NLV.

After the initial setup phase, the nodes enter the maintenance phase. During the maintenance, each node continues to send "hello" messages as before. The frequency of the beaconing can be decreased in the maintenance phase to reduce the energy consumption and the protocol overhead. In order to keep the neighbor table up-to-date, the nodes refresh their neighbor table every T seconds. If a neighbor's link NLV has decrease over time, that node will be replaced by a neighbor with higher NLV.

5.3 Angled Probabilistic Geographic Routing Protocol

As mentioned before, we assume the nodes know their geographical location and the location of the destination node. If the source does not know the destination's location, it can use a location service scheme [23] to determine the coordinates of the target. We explain the routing algorithm using figure.

Let us assume that node S is the source, D is the destination, and nodes 1 through 8, are in S's neighbor table. When S wants to send a packet to D, it looks in its neighbor table, and selects the neighbors that fall within an angle θ from D, as shown in figure. The initial value for θ can be picked arbitrarily. If S cannot find more than one neighbor within the initial angle, it will increase θ until it finds at least two neighbors. If S has to open the angle θ beyond 180˚ to find a neighbor, it stops at that point, and drops the packet. The motivation behind this approach is to guarantee an "almost" loop-free protocol. Given the angle θ will never be opened up beyond 180˚, a packet will never be sent in the "backward" direction, and will always have a positive progress toward the destination. However, to ensure that a packet does not cycle in a loop, every time the packet is forwarded, the id of the forwarding node is included in the packet header. Once a packet is received by a node, its header is examined, and if the current node is listed in the header, the packet is dropped.

5.4 Backup Mechanism and Relay cancellation:

Now let us assume that S was able to find at least two neighbors within the angle θ, nodes 1 through 4 in our example of figure. These four nodes are the candidates for the next hop. Source S then proceeds to assign probabilities to each of these candidate nodes using their corresponding NLV. (Candidate Nodes are node having NLV greater then minimum threshold.)

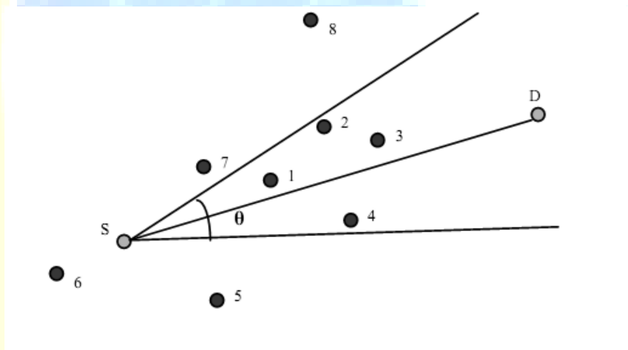


Figure: 3 *Backup Mechanisms and Relay Cancellation*

Node with higher NLV values are selected as next hop in relay zone, node having higher NLV among neighbor selected for next hop and is used for forwarding packets and remaining candidate node will work as backup mechanism. Assume that node 1 is selected as next hop due to higher NLV values. Node 1 having higher NLV values compare to other candidate nodes, so Node 1 will work as forwarding node and remaining will work as backup mechanism, so packet forwarded to Node 1 will have a forwarding variable flag value = F in routing table and remaining will have a forwarding variable flag value = B.

Node selected for next hop will again follow the same process if it not found at least two nodes at a specified value of angle θ the relay zone is increased by changing the value of angle θ. If value of angle θ reaches the value 180 then backup mechanism is in action with relay cancellation for the forwarding selected node. For backup mechanism the symmetric link are essential, now node with highest NLV will select from backup node to forwarding a packet. Forwarding mechanism will be as above.

Therefore, if the backward reliability to one candidate node is not as good as another candidate, the probability corresponding to that neighbor will be lower. Also, if the NLV of a candidate node is lower than another candidate neighbor, its corresponding probability will be smaller. This approach will ensure that the energy consumption is balanced among nodes, and a node with a reliable link will not be drained out of energy due to being selected continuously. Once the probabilities are assigned to each of the candidate neighbors, a Roulette wheel selecting algorithm is used to pick a node proportional to its assigned probability.

Roulette-wheel selection is also known as Fitness proportionate selection, a genetic operator used in genetic algorithms for selecting potentially useful solutions for recombination.

In this selection, as in all selection methods, the fitness function assigns fitness to possible solutions or chromosomes. This fitness level is used to associate a probability of selection with each individual chromosome. If fi is the fitness of individual i in the population, its probability of being selected is pi, where N is the number of individuals in the population.

$$\mathrm{Pi} = \frac{f_i}{\sum_{j=1}^{N} f_j}$$

5.5 POEGR Functional Design:

Our aim is designing a geographic routing protocol for adhoc sensor network. The network may be heterogeneous, and sensor nodes may be mobile. Assume that each node knows its position either via GPS or some other localization methods. We first present the structure of the routing table along with the forwarding algorithm. Then we show how to construct routing tables and how to trigger updates.

We propose POEGR which is an efficient way to obtain route optimality in adhoc sensor network and is an effective approach to achieve good scalability. In POEGR, each node discover one-hop neighbour for a specific value of θ in the network.

Each POEGR node also maintains a Link value NLV for energy efficient routing. When forwarding a packet, it picks the node with maximum NLV, according to which it looks up the best next hop in the routing table. The angle θ will never be opened up beyond 180˚, a packet will never be sent in the "backward" direction, and will always have a positive progress toward the destination. However, to ensure that a packet does not cycle in a loop, every time the packet is forwarded, the id of the forwarding node is included in the packet header. Once a packet is received by a node, its header is examined, and if the current node is listed in the header, the packet is dropped.

6.0 ROUTING TABLE AND FORWARDING ALGORITHM:

The routing table of a node u is defined as RT_A = {(NodeID, NodePos, NLV, hops,ForwardingFlag, UpdateSeqNum, LinkStatus }, where RT_A denotes the routing table of node A; NodeID is the identification of a candidates node for A; NodePos is the position of that candidates node for A; hops is the minimum number of hops from node A to candidate node A; ForwardingFlag is Flag denotes that the candidate node used for forwarding or backup mechanism; UpdateSeqAngle is Angle at which candidate node is discovered by ad hoc router; LinkStatus is Link Symmetric show BICONNECTED link or CANCELLED Link.

The forwarding algorithm at each node A is described in pseudo code as follows.

ForwardPacket(packet, A)

{

Declare ϴ = 60^o;

```
If (Examine header: find entry of node A as forwarding node==Match)
{
        GoTo NextHopSelection;
}
SendAck(forwarding node);
If (node A is the destination of packet)        {
        Send packet to the upper layer;         }
Else
{
        If (Examine header: find entry of destination in RTA[*] ==Match)
        {
        ForwardPacket(packet, destination)
        }
        Else {
                If (RT_A not Initilize)
                {
                        InitRoutingTable(A);
                        ConstructRoutingTable(A, θ);
                }
                ElseIf (RT_A update TimeStamp is Old)
                {
                ConstructRoutingTable(A, θ);
                }
                Label: NextHopSelection;
```

NLVmax(RT_A) ;

SelectedCandidate node X= SelectNode(RT_A[*].forwardingFlag =F)

RT_A[*].forwardingFlag =B

Forwardpacket (packet, X);

}

}

}

6.1 Constructing Routing Tables

POEGR constructs routing tables in the similar way to DSDV that is based on the distributed Bellman-Ford algorithm. There are two main differences. The first is that each POEGR node has a limited one hop of the network, whereas each DSDV node knows all the other nodes in the network. The second is that, if a routing table changes, DSDV broadcasts the change immediately, whereas POEGR does not unless the corresponding NLV is changed. Therefore, in comparison with DSDV, POEGR is much more scalable and costs much less.

At the beginning, the routing table RT_A of each node A contains only one entry that is related to itself. The entry is always stored in RT_A[1], the number 1 row in A's routing table, which is initialized as follows.

InitRoutingTable(A)

{

RT_A[1].Sr = 1;

RT_A[1].NodeID = A.NodeID;

RT_A[1].NodePos = A.NodePos;

RT_A[1].NLV = A.NLV;

RT_A[1].hops = 0;

RT_A[1].ForwardingFlag = S;

RT_A [1].UpdateSeqNum=0

RT_A[1].LinkStatus=B

}

After initialization, each node may be triggered somehow to broadcast its information to its neighbours by control messages. When a node receives a neighbor's information, say RTneigh, it updates its own according to the neighbor's knowledge. The procedure is described as follows.

```
ConstructRoutingTable(A, θ)
{
Broadcast Control Packet from A;
For each Control packet of Aneigh of A at angle θ
{
        if (SearchNode(Aneigh,RT_A ) != Match)
        {       Insert Aneigh into RT_A as a new entry Enew ;
                RT_A[Enew].Sr += 1;
                RT_A[Enew].NodeID = Aneigh.NodeID;
                RT_A[Enew].NodePos = Aneigh.NodePos;
                RT_A[Enew].NLV = CalculateNLV(Aneigh, A);
                RT_A[Enew].hops = 1;
                RT_A[Enew].ForwardingFlag = B;
                RT_A[Enew].UpdateSeqAngle= θ
                RT_A[Enew].LinkStatus= B
        }
        Else
        {       Update Aneigh into RT_A as a entry Eold ;
```

```
            RT_A[Eold].NodePos = Aneigh.NodePos;
            RT_A[Eold].NLV = CalculateNLV(Aneigh, A);
      }
}

      If(RT_A.Count <=2 and θ >= 180° or RT_A[*].NLV < MinThreshold)
      {     BackupMechanism(A);
            Drop Packet;
      }
      Else
      {     θ += 30°;
      ConstructRoutingTable(A, θ)
      }
}
```

7.0 MOBILITY SCENARIO:

As consider in mobility for ad hoc network, either the mobile node will join the network or may leave the route region. If node joins the route region the node may either candidate node or non-candidate node similar can be consider for mobile node leave the route region, the node leave the network may be candidate node or non-candidate node. Along with this ad hoc router node can also leave the route region. If mobile nodes are there in network there location may vary after each time interval T.

Taking care of all above possibilities appropriate solution must be in taken for our algorithm. Note that no update will be done by the nodes which are not working as ad hoc router. Triggering update must be done for specific mobility varying value.

Ad hoc router node look updateSeqAngle field in routing table when a candidate node joins the route region, if current angle is less than in routing table then update for redundant node is made in routing table. Similar for non-candidate node if angle of node discovery is greater then the current angle new entry for non-candidate node will be done.

When candidate node departure the route region and due to mobility threshold value it broadcast control message updates is mentioned in routing table of ad hoc router and if non-candidate node departure the route region no updates are made in ad hoc router.

Although ad hoc router node depart the route region and it is candidate node so it send mobility message to all its 1-hop ad hoc router nodes to update the information. Probability of ad hoc router broadcast the mobility message depends on mobility threshold value. As mobility threshold values is small ad router nodes routing table will hold more live values.

TriggeringUpdates(Forwarding node A, packet)

{

If (packet = Mobility Message)

{

if (SearchNode(Aneigh, RT_A) != Match and RouteDiscoveryAngle of A > C_θ) { Insert Aneigh into RT_A as a new entry Enew ;

RT_A[Enew].Sr += 1;

RT_A[Enew].NodeID = Aneigh.NodeID;

RT_A[Enew].NodePos = Aneigh.NodePos;

RT_A[Enew].NLV = CalculateNLV(Aneigh, A);

RT_A[Enew].hops = 1;

RT_A[Enew].ForwardingFlag = B;

RT_A[Enew].UpdateSeqAngle= θ

RT_A[Enew].LinkStatus= B

}

Else {

Update Aneigh into RT_A as a entry Eold ;

RT_A[Eold].NodePos = Aneigh.NodePos;

RT_A[Eold].NLV = CalculateNLV(Aneigh, A);

}

}

Else

If (packet !=ACK)

{

RTA.LinkStatus[NFN] = C;

ForwardPacket(packet, A);

}

}

8.0 BACKUP MECHANISM:

In POEGR backup mechanism will execute in two possible situations. First forwarding node does not find minimum two candidate nodes in the relay zone up to 180^o of angle. Second the forwarding node does not find candidate node with minimum threshold of NLV.

The pseudo code for backup mechanism is as follows:

BackupMechanism(A)

{

Broadcast control packet from A having _

(flag= relay cancellation, Ɵ = 360^o, backward forwarding NodeID)

Backward forwarding node BFN;

RT_{BFN}[A].LinkStatus = C

Forwardpacket (packet, BFN);

}

9.0 SIMULATION RESULTS AND EVALUATION:

Performance of PEOGR with the prior work, we simulated probabilistic flooding (PF), GPSR, GRS, NADV and LBLSP as well. These protocols are the adjoining to our protocol; the first one has the probabilistic nature, and the second protocol uses geographical routing, third uses Greedy routing, fourth uses Link Cost and fifth protocol uses Load Balanced Local Shortest Path

9.1 Simulation Environment

We simulated our protocol using NS-2 and the wireless extension to NS-2 which was developed at Carnegie Mellon Ns-2 is a discrete network simulator developed at UC Berkeley. We also used the codes for PF, GPSR, GRS, NADV and LBLSP that already existed in NS-2.

In the following we describe NS-2 network model in some detail.

Radio Propagation Model: The physical model we used is called TwoRayGround. The TwoRayGround model is the most realistic model to use for ad hoc wireless network simulations. The advantage of the TwoRayGround model is that it extends the ideal circle model to a statistical model where the nodes near the edge of the circle can only probabilistically communicate.

Energy Model: NS-2 has an implementation of a simple energy model in which every time a packet is transmit, the total energy of the node decreases by the value, The same formula applies for decreasing the energy when a packet is received.

Energy Reduced = Packet Transmit ×Transmit *Time*

9.2 Comparison Metrics

In order to compare the performance, we choose the following metrics:

Packet delivery ratio: This is defined as the ratio of the number of packets received by the destination, to the number of packets originated by the source.

Delay: The average time taken between when a packet was initially sent by the source, and the time it was *successfully* received at the destination.

Path Length: Path length is defined as the number of hops a packet takes to reach its destination.

Number of Retransmissions: The total number of valid retransmissions required to go from a source to destination. Valid retransmissions are defined as retransmissions due to not receiving an acknowledgement (ACK) for a transmitted data packet.

9.3 Simulation Setup

In our simulations we have compared using the metrics described. Summary and scenarios are as follows.

Radio Propagation Model	TwoRayGround
Number of Nodes	50-100
Square Area	1000 x 1000 m
Antenna type	Omni Directional
Transmission Range	200m
Interference Range	500m
CBR Traffic	20

CBR Flow	4 Packet/sec
Packet size	512
Max Packet size	1000
Simulation time:	500 ms
Mobility model	Random Waypoint

9.4 Simulation Results

The results we present here have been averaged

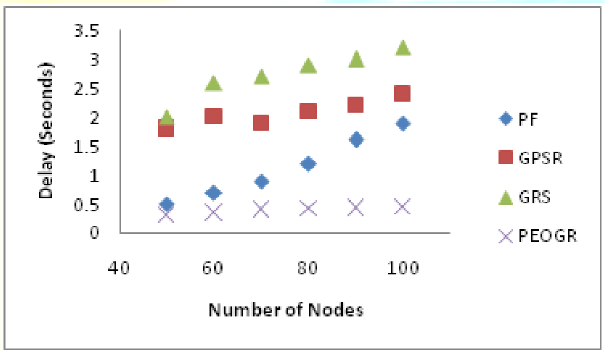


Figure: 4 *Delay (seconds) with 20 CBR source/destination pairs*

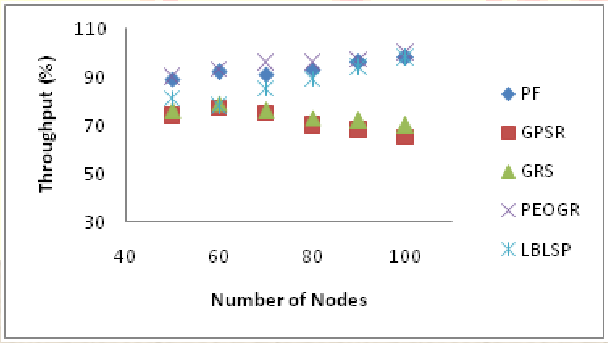


Figure: 5 *Throughput (%) with 20 CBR source/destination pairs*

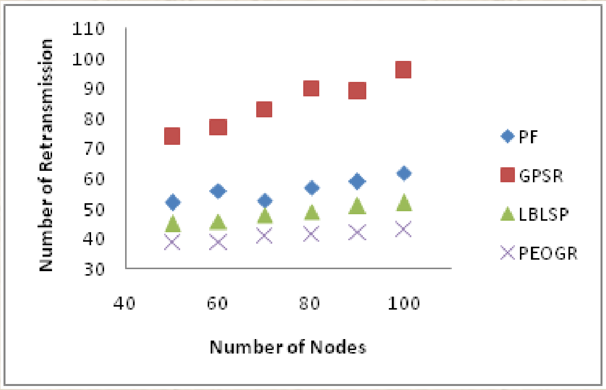


Figure: 5 *Number of Retransmissions*

10.0 CONCLUSION:

We introduced PEOGR which is a non cluster energy aware routing protocol for wireless ad hoc and sensor networks. PEOGR uses geographical location along with location error, remaining energy, and obstacle geometry and link reliability information to make routing decisions. Instead of deterministically choosing the next hop, PEOGR assigns angled probabilities to the candidate nexthop nodes.

Using the remaining energy in the cost function ensures that nodes with more reliable links are not drained out of energy too quickly. This will in turn increase the lifetime of the network. In addition, PEOGR attempts to locally minimize the number of retransmissions.

Reducing the number of retransmissions contributes to saving energy, and increases the overall lifetime of the network. The other advantage of PEOGR is that it does not require keeping a state per route. This will reduce the routing protocol overhead and the amount of routing information that needs to be stored at each node. This makes PEOGR simple to use.

Given the next hop is chosen angled probabilistically, and it back off and relay termination mechanism blocks unusual flooding in network. This will help expedite the routing process; if transmission to a neighbour fails, instead of the trying to retransmit on the same link multiple times, the MAC layer has the freedom to choose a different neighbour for retransmission.

An angled probabilistic routing scheme has a built-in security advantage. Since the next hop is chosen arbitrarily and not based on a deterministic rule, it is more complicated for an opponent to attack and capture a message.

References:

- Finn. Routing and Addressing Problems in Large Metropolitan-scale Internetworks. Technical Report ISI/RR-87-180, USC/ISI, March 1987.
- H. Frey and I. Stojmenovic, “On delivery guarantees of face and combined greedy-face routing in ad hoc and sensor networks,” in Proc. ACM MobiCom, Los Angeles, CA, Sep. 2006.
- F. Kuhn, R. Wattenhofer, Y. Zhang, and A. Zollinger. Geometric ad-hoc routing: Of theory and practice. In Proc. PODC 2003, Boston, MA, USA, July 2003
- Chen, D., Varshney, P.K.: A survey of void handling techniques for geographic routing in wireless networks. Communications Surveys and Tutorials, 50–67 (2007)
- Gustavo Weber Denardin, Carlos Henrique Barriquello, Alexandre Campos, and Ricardo Nederson do Prado. “A geographic routing hybrid approach for void resolution in wireless sensor networks”, J. Syst. Softw. 84, 10, October 2011.
- Young-Jin Kim, Ramesh Govindan, Brad Karp, and Scott Shenker. On the pitfalls of geographic face routing. In Third ACM/SIGMOBILE International Workshop on Foundation of Mobile Computing, DIAL-M-POMC, 2005.
- Y.-J. Kim, R. Govindan, B. Karp, and S. Shenker. Geographic routing made practical. In Proceedings of the 2nd Symposium on Networked Systems Design and Implementation (NSDI 2005), May 2005.
- Leong, B., Liskov, B., & Morris, R. (2006). Geographic routing without planarization. In 3rd Symposium on Networked Systems Design & Implementation (NSDI), San Jose, California, USA, May 2006.

- C.H. Lin, S.A. Yuan, S.W. Chiu, and M.J. Tsai, "Progressface: An algorithm to improve routing efficiency of GPSR-like routing protocols in wireless ad hoc networks , "IEEE Transactions on Computer", vol. 59, no. 6, pp. 822 – 834, June 2010.
- Young Il Ko, Chang-Sup Park, In Chul Song, and Myoung Ho Kim. 2009. An efficient void resolution method for geographic routing in wireless sensor networks. J. Syst. Softw. 82, 963-973, June 2009.
- H. Frey and I. Stojmenovic. "On Delivery Guarantees and Worst Case Forwarding Bounds of Elementary Face Routing Components in Ad Hoc and Sensor Networks". IEEE Trans. on Computers, 2011.
- Seada, K., Helmy, A., And Govindan, R. Localization errors on geographic face routing in sensor networks. In Proc. IEEE IPSN Workshop, Berkeley, CA, USA, Apr. 2004.
- HenrikLundgren, ErikNordstr, ChristianTschudin, Coping with communication gray zones in IEEE 802.11b based adhoc networks, in: Proceedings of the Fifth ACM International Workshop on Wireless Mobile Multimedia, 2002, pp.49–55.
- Douglas S.J. De Couto, Daniel Aguayo, John Bicket, Robert Morris, A high-throughput path metric for multi-hop wireless routing, in: Proceedings of the Ninth ACM/IEEE MobiCom, 2003.
- Seungjoon Lee, Bobby Bhattacharjee, Suman Banerjee, Efficient geographic routing in multihop wireless networks, in: Proceedings of MobiHoc, 2005, pp. 230–241.
- Suresh Singh, Mike Woo, C.S. Raghavendra, Power-aware routing in mobile ad hoc networks, in: Proceedings of the Fourth ACM/IEEE MobiCom, ACM Press, 1998, pp. 181–190.
- Brad Karp, H.T. Kung, GPSR: greedy perimeter stateless routing for wireless networks, in: Proceedings of the Sixth ACM/IEEE MobiCom, ACM Press, 2000, pp. 243–254.
- Stojmenovic; Position-Based Routing in Ad Hoc Networks; IEEE Communication Magazine; July 2002
- Ward, A. Jones, A. Hopper; A New Location Technique for the Active Office. IEEE Personal Communications, October 1997

- Woo, Suresh Singh, C. S. Raghavendra; Power-Aware Routing in Mobile Ad Hoc Networks; ACM/IEEE Mobicom, 1998.
- Woo, T. Tong, David Culler; Taming the Underlying Challenges of Reliable Multihop Routing in Sensor Networks; Sensys 2003
- Woo; Evaluation of Efficient Link Reliability Estimators for Low-Power Wireless Networks
- Girod, D. Estrin; Robust Range Estimation Using Acoustic and Multimodal Sensing; IEEE/RSJ International Conference on Intelligent Robots and Systems, October 2001
- Shah, Jan M. Rabaey; Energy Aware Routing for Low Energy Ad Hoc Sensor Networks