Optimized Solution to Geographic Routing Protocol for Wireless Sensor Networks

Asha Sohal[1], R.L. Ujjwal[2]

[1]*USIT, Student of M.Tech (IT), Guru Gobind Singh Indraprastha University, Delhi*
[2]*USIT, Assistant Professor, Guru Gobind Singh Indraprastha University, Delhi*

ABSTRACT

Geographic routing is mostly used in wireless sensor networks. In this paper, we use geographic routing algorithm in which we can use different levels of mobility by changing its factors depending on the network in which it is running. Routing decisions are dependent on directions and geographical positions of the nodes. One of the most effective geographic routing protocol is GPSR (Greedy perimeter stateless Routing). There are still some problems for this type of protocol like large routing protocol overhead and less reliability for long link. We propose a novel geographic routing protocol, Modified RGRP (Reactive Geographic Routing Protocol), which consists of reactive routing mechanism and geographic routing. Basically Modified reactive routing mechanism is used to reduce the packets for routing discovery and end-to-end delay. Furthermore, geographic routing is used to find the optimal path between different numbers of nodes. Finally, we make experiments and comparison between modified RGRP and GPSR and simulation results shows the performance of our protocol.

1. INTRODUCTION

A wireless sensor network is a collection of nodes organized into a cooperative network. Each node consists of processing capability (one or more microcontrollers, CPUs or DSP chips), may contain multiple types of memory (program, data and flash memories), have a RF transceiver (usually with a single Omni-directional antenna), have a power source (e.g., batteries and solar cells), and accommodate various sensors and actuators. The nodes communicate wirelessly and often self-organize after being deployed in an ad hoc fashion. Systems of 1000s or even 10,000 nodes are anticipated. Such systems can revolutionize the way we live and work. Currently, wireless sensor networks are beginning to be deployed at an accelerated pace.

Geographic routing [1] is effective for large multi-hop wireless networks in which single nodes are not reliable. With the help of sufficient information about the geographic location of nodes these protocols allow each node to determine the next hop to forward the packet. GPSR [1] is the most helpful routing protocol based on geographic information. We opt GPSR as a base routing protocol and try to identify the difficulties of general geographic routing protocol since many geographic routing protocols are similar to GPSR on the aspects of greed forwarding decision and use Calculate the best paths because here we have to calculate shortest path between source nodes to destination node.

Modified Reactive Routing Protocols

As I earlier mentioned above, routing protocols for wireless sensor networks can be classified as proactive and reactive routing protocol. Proactive routing protocols are used whenever a message needs to be forwarded. There are many routing table to calculate the shortest path between source nodes to destination node [2]. These protocols are categorized into two parts: proactive and reactive routing protocols. In proactive routing protocols, since every node keeps a route table which specify how to forward a message and information. Therefore the routing overhead incurred by table update messages can become unacceptably high. On the other hand in reactive routing protocols, we try to delay any preparatory actions as long as possible. With the help of flooding techniques a node can send a message and find its destination. The rest of this paper is organized as follows. Section 2 defines a brief overview of related work. Section 3 explains Modified RGRP routing protocol. Section 4 presents simulation results. Finally, conclusions and future works are presented in Section 5.

2. RELATIVE WORK

Geographic Routing Protocol

Greedy forwarding mechanism is the most important component of geographic routing techniques. Greedy routing protocols have is based on distances, progress, and/or direction. Each node forwards the packet to the neighbour which is the closest to the destination with the help of distance-based protocol. Each node forwards the packet to the neighbour that provides the most progress towards the final destination with the help of progress-based protocols [3]. With direction based routing the packet is forwarded to the neighbour that minimizes the angle between the neighbour, the forwarding node itself, and the destination. In addition, we alter the traditional perspective of geographic routing with two new measures, and from simulation we conclude that our new routing protocol has excellent performance than GPSR in overhead, reliability and efficiency. We compare Dynamic Source Routing (DSR),[4]-[5] introducing the idea of a sequence of nodes established before sending data, giving the whole path to the sender and Dynamic Destination-Sequenced Distance-Vector Routing (DSDV),[6] where each node broadcasts their current neighbours to the network. Ad hoc On-Demand Distance Vector (AODV) Routing presented a reactive distance vector algorithm broadcasting paths only when needed. Other proposals such as the Optimized Link State Routing (OLSR) Protocol have some pieces of information being broadcasted to all the nodes, using it to

protocols which create routing information and neighbour node information such as the DSDV [6] , the TBRPF [7] and the OLSR [8]. Besides this, reactive routing protocols do not need to send hello packet to its neighbour nodes frequently to maintain the coherent between nodes just like as the AODV [9] and DSR [10] . It does not need to distribute routing information and to maintain the routing information which indicates that the routing links have been already broken. The neighbour table and routing information would only be created when a message needed to be forwarded and nodes maintain this information just for a certain lifetime. When the lifetime of information is over, nodes discard all these routing and neighbour information. Then if another message needs to be forwarded or a routing path needs to be found, nodes would create new routing and neighbour information for the next time.

Here we are using a modified novel to produce low overhead and high efficiency geographic routing protocol based on reactive routing mechanism. But without reactive routing protocols we cannot use geographic information. So we propose a new geographic routing protocol, Modified RGRP (Reactive Geographic Routing Protocol).

3. OUR NEW PROTOCOL

In this paper, we provide an efficient, high-reliability and low-power routing protocol. Our main focus is on reactive routing mechanism and geographic information from which we can modify RGRP (Reactive Geographic Routing Protocol). Here we are using GPSR to calculate the shortest distance between destination node and neighbor node in the neighbour table. We have two new measures to improve the reliability of routing protocol. First, we use reactive routing mechanism to reduce the routing protocol overhead. Second, we use two steps to finish the shortest path finding instead of classical geographic routing mechanism, which maybe produce voids in networks and these two steps helps in reducing the packet loss rate.

3.1 Reduction of Packets Used for Route Discovery

In RGRP, Basically we have only two types of packet for route discovery: Route Request (RREQ) and Route Reply (RREP). These two types of packets fulfil different functions in route discovery. Most importantly, nodes only maintain the route table they need for the lifetime of every route entry. In the other word, nodes do not need to maintain all of the route information in the network, any route information beyond the route needed does not exist in the network.

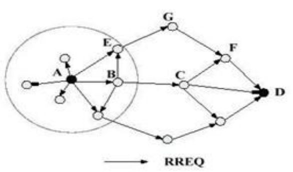

Fig. 1: How to show RREQ in network

distance the node calculates at the first time when it receives RREQ or the distance at current time. The ***gap*** is distance the node calculates when it receives RREQ again.

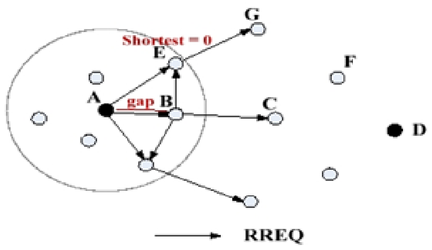

Fig. 2: How to show RREP in network

As shown in Fig. 1, when a message needs is forwarded from node A (source node) to node D (destination node), node A will broadcast RREQ, which is identified by a broadcast ID and the address of node A. Then nearest node B or E, receives this RREQ, it will check the destination address of RREQ. If the destination address is itself, it will create and send RREP. Otherwise, it continues broadcasting this RREQ. By this method, the RREQ will finally be delivered to the node D (destination node).

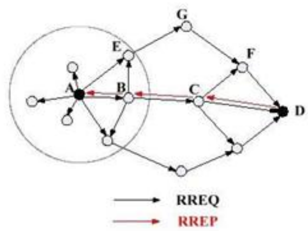

Fig. 3: Reverse route entry and calculate distance in RREQ

As shown in Fig. 2, when node D finds that the destinations address of the received RREQ is itself, it will create and unicast RREP to node A (source node) by the reverse route table which we will introduce by following. Note that only the destination node could create RREP and unicast RREP to the source node. Any other node which is not the destination of the RREQ could only forward RREQ and RREP.

In the Modified RGRP, we use two methods to decrease the number of packets used for route discovery. (1) We use reactive routing mechanism as our main routing mechanism. It is not necessary for nodes to maintain route tables and neighbor tables for a long time. Nodes will create new route entry when a message needs to be forwarded. (2) We do not use neighbor tables to maintain the coherent between nodes. We update the route table to finish the whole progress of route discovery.

3.2 Searching Shortest Path

With the help of GPS, we have to calculate the shortest path from source node to destination node. However, unlike with GPSR, we calculate the shortest path between source node and destination node by two steps, which are executed both in forwarding RREQ and RREP.

1) Reverse Route Calculation in RREQ

Each node will create a route table called reverse route table when it receive a RREQ. First of all, when a node receives RREQ, it will create a route entry which indicates the next hop (the node forwarding the RREQ) to the source node and calculate the distance between this next hop node and the source node. Second, this node will also make the similar decision when it receives RREQ, update route table or discard RREQ. For convenience, we use two variables (***shortest*** and ***gap***) to indicate how to make reverse route calculation in RREQ. The ***shortest*** is

As shown in Fig. 3, when node A broadcasts RREQ to node B and E, node B and E will create a reverse route entry which indicates the next hop to the source node when packet arrives at node B and E. Besides, node B and E would calculate the distance between forwarding node and source node. In this situation, the next hop to source node for node B and E is node A and the ***shortest*** for node B and E is 0, because node A is both the forwarding node and source node. And then, when node B forwards the RREQ to node E, node E will calculate the ***gap*** which is the distance between forwarding node (node B) and the source node (node A). Then, node E will compare the ***gap*** with ***shortest*** (the first distance when node E receives RREQ from node A). Since ***gap*** >***shortest***, the node discard this RREQ.

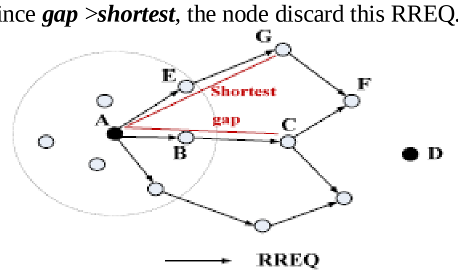

Fig. 4(a): update route table in RREQ

This is the easiest way to find the shortest path between the source nodes and destination nodes. But every time we have to check the nearest nodes as well as the route. After that we have to update the route table also so that it is easy to understand which nodes covered the route successfully. This gives us a better and efficient result in updating the route Table details.

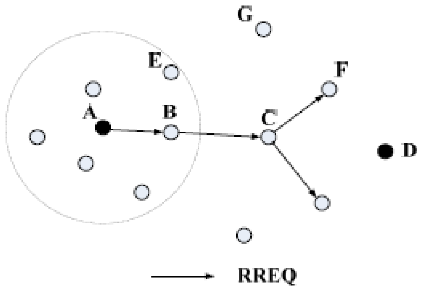

Fig. 4(b): The result of updating route table in RREQ

As shown in Fig. 4(a), node F creates reverse route entry when it receives RREQ from node G and select node G as the next hop to the source node (node A). The same process happens when node F receives the same RREQ again from node C. Node F calculates the ***gap*** between the forwarding node (node C) and the source node (node A). Since ***gap*** <***shortest***, node F updates the route table and select node C as the next hop to the source node. Fig. 4(b) is the finally route after node C broadcasts RREQ.

2) Reverse Route Calculation in RREP

We use same concept to get the optimal path in forwarding RREP. Distance factor is only difference when we calculate in RREP is from the node forwarding RREP to the destination node.

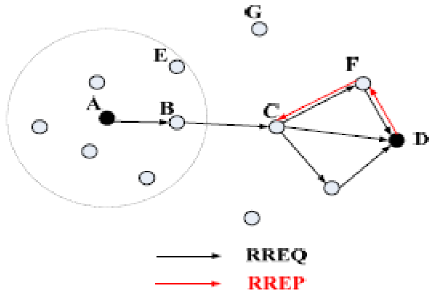

Fig. 5: Reverse route entry and calculate distance in RREP

As shown in Fig. 5, the destination node (node D) receives the RREQ from node F and then creates the RREP and unicasts it to node F. Node F forwards this RREP to node C according to the route table created by forwarding the RREQ.

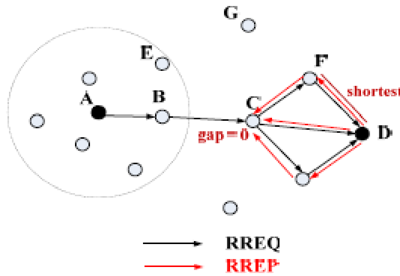

Fig. 6(a) Update route table in RREP

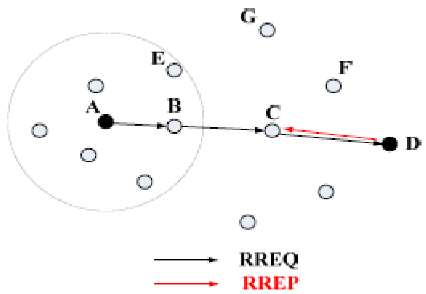

Fig. 6(b): the result of updating route table in RREP

As shown in fig. 6(b), node C receives the RREP from node F, it creates the route entry and calculates the ***shortest***, which indicates the next hop is node F when the message whose destination node is node D arrives at node C. And then, when node C receives RREP from node D, it will calculates the ***gap*** and finds that ***gap*** < ***shortest***, node C updates the route table, and then finally optimal path is found.

4. PERFORMANCE EVALUATION

A. Simulation Environment

We simulated Modified RGRP and GPSR in NS-2.35 [10]. It includes full simulation of the IEEE 802.11 physical and MAC layers, and the propagation model is a two ray ground model. In the simulations, our simulations are for networks of 5, 10, 15, 20, 25 and 50 nodes with 802.11 Wave LAN radios, with a nominal 250-meter range. All these only for configuration of nodes and we placed all the nodes randomly in square fashion. We performed tests on one random UDP connection between random source node to random destination node. Additionally, we simulated in three different cbr transmission models: transmit rate is 2Kbps, 128Kbps and 512Kbps, data size is 64byte, 250byte and 1000byte respectively, to verify the performance for low, middle, high demands for routing protocol. Each simulation lasts for 60 seconds of simulation time. Moreover, since the interval of sending Hello packet of GPSR is a key element for its performance, we choose two types of interval to compare with modified RGRP. The variable of interval is ***B*** = 2.5s and 5.0s. Table 1 summarizes the different scenarios for our simulation.

TABLE I

SIMULATED TOPOLOGY CHARACTERISTICS

Nodes	Region	Transmit Rate & Data Size
5	800m*800m	2Kbps&64byte
10	800m*800m	128Kbps&250byte
15	800m*800m	
20	1000m*1000m	512Kbps&1000byte
25	1000m*1000m	
50	1000m*1000m	

The evaluation of modified RGPR and GPSR are based on three parts, number of packet used for route discovery to substantiate the low consumption, number of packet loss to substantiate the reliability and average end-to-end delay to substantiate the efficiency.

B. Calculating Routing Protocol Overhead

We calculate the number of packet used for route discovery to verify the overhead of routing protocol. As mention above, there are only two types of packet used for route discovery, RREQ and RREP. Besides, since we do not implement neighbor table in Modified RGRP and use the reactive routing mechanism, the number of packet used for route discovery is much more less than GPSR. And since GPSR sends hello packet frequently to maintain neighbor table and chooses the nearest node to the destination node in neighbor table, the number of packet increase drastically as the number of node increasing.

.

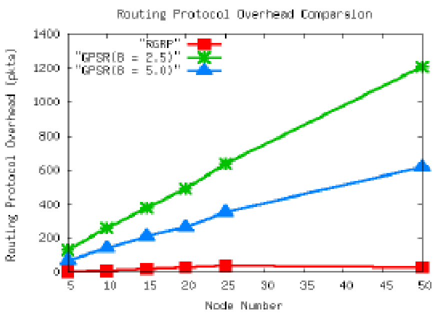

Fig. 7: Route Discovery Overhead Comparison

Since the routing protocol overhead is not relevant with the state of date transmission, the overhead of routing protocol is always the same when the cbr transmit rate and data size changed. As shown in Fig. 7, the overhead in the RGRP is less than 50 at 25-node networks, but the GPSR uses almost 352 and 641 packets for B is 5.0s and 2.5s. The less interval of sending Hello packet, the more packets sent in GPSR. Then, the green line is above the blue line. The most importantly part; packets of RGRP used for route discovery are not relevant with the number of the nodes in the network. Only the number of the nodes in optimal path affects the number of the RREQ and the RREP. That is why we observe that the number of packet used at 50-node is less than at 25-node. This special feature is flexible to apply at many scenarios, such as dense networks or sparse networks or even mobile environment.

C. Calculating Packet Loss Rate

We calculate packet loss rate to verify the reliability of Modified RGRP and GPSR. In the same scenario, we calculate the number of packet loss by counting the number of packet sent by source node and the number of packet received by destination node. When the cbr transmission rate is 2Kbps and data size is 64byte, the packet loss rate is all zero since it is low demands for routing protocol. As we earlier told you that all the nodes are chosen randomly and they are in square according to configuration of nodes.

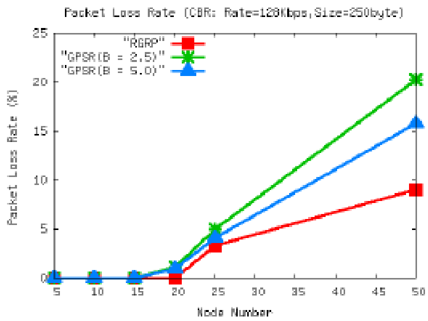

Fig. 8(a): Packet Loss Rate for 128Kbps, 250byte

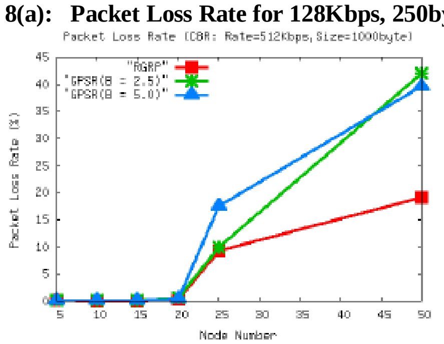

Fig. 8(b): Packet Loss Rate for 512Kbps, 1000byte

As shown in Fig. 8 (a) and (b), the packet loss rate of Modified RGRP and GPSR are almost same for 5, 10, 15, 20 nodes in the network for two different transmit rates and data sizes. While when the number of nodes is increasing at 25, we can observe the number of packet of modified RGRP is slightly lower than GPSR for both scenarios. As the node number is increasing at 50, we can make a definite conclusion that the packet loss rate of modified RGRP is 50% lower than GPSR.

5. CONCLUSION AND FUTURE WORK

In this paper, we are trying to impose a new geographic routing protocol combined with modified reactive routing mechanism. In which we have two methods by which we can calculate the shortest path: first, we calculate the shortest path to the source node and create reverse route table, second, we filter these paths by calculating distance to the destination node. From simulation, we can observe definitely that the performance of Modified RGRP is better than GPSR in routing protocol overhead and packet loss rate.

As future work, we will consider the energy consumption as another important fact for route discovery. Also, we will make some measurement to increase the reliability of modified RGRP, especially on how to fix the link when a node died or another new node joints in the network.

6. REFERENCES

[1] M. Zuniga, K. Seada, B. Krishnamachari, and A. Helmy, "Efficient geographic routing over lossy links in wireless sensor networks," *ACM Transactions on Sensor Networks*, vol. 4, no. 3, pp. 1–33, May 2008.

[2] V. Kranakis, H. Singh, and J. Urrutia, "Compass routing on geometric networks," in *Proc. Canadian Conference on Computational Geometry '99*, Vancouver, Canada, Aug. 1999.

[3] Fabian Kuhn, Roger Wattenhofer, Yan Zhang and Aaron Zollinger, "Geometric Ad-Hoc Routing: Of Theory and Practice," in

Principles of Distibuted Computing, 2003.

[4] Dongjin Son, Ahmed Helmy, and Bhaskar Krishnamachari, "The effect of mobility-induced location errors on geographic routing in mobile ad hoc and sensor networks: analysis and improvement using mobility prediction," in *IEEE Trans. On Mobile Computing*, Vol. 3(3), 2004.

[5] S. Das, A. Nayak, S. Ruhrup, I. Stojmenovic, "Semibeaconless power and cost efficient georouting with guaranteed delivery using variable transmission radii for WSNs," in *Proc. of LOCAN*, Pisa, Italy, 2007.

[6] C. Perkins and P. Bhagwat, "Highly dynamic Destination-Sequenced Distance-Vector Routing (DSDV) for mobile omputers," in *Proc. ACM SIGCOMM'94*, 1994, pp. 234–244.

[7] R. Ogier, F. Templin, and M. Lewis, "Topology Dissemination Based on Reverse-Path Forwarding (TBRPF)," IETF RFC 3684, Feb. 2004.

[8] T. Clausen and P. Jacquet, "Optimized Link State Routing Protocol (OLSR)," IETF RFC 3626, Oct. 2003.

[9] C. Perkins and E. Royer, "Ad-hoc on-demand distance vector routing," in *Proc. 2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90–100.

[10]NS-2.[Online]. Available: http://www.isi.edu/nsnam/ns/