IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 6, June 2018

# An Efficient Geographic Routing using Spanning Tree and Greedy Distributed Routing in WSN

**Linsy A[1], Dr. R. Deepa[2]**

Department of Computer Science, NGM College, Pollachi[1,2]

**Abstract:** Geographic routing specially for location information based routing. It is mainly proposed for wireless networks and based on the idea that the source sends a message to the geographic location of the destination instead of using the network address. The existing trap array topology model that provides a unified framework to uncover the limiting behavior of 10 representative geo-routing algorithms. The problem with such a trap array approach is that it is doubtful to route the packet efficiently, the approach can guarantee that a packet will be delivered no more than 2 - 3 hops, need to add more hops to deliver the packet. No planarization techniques applied to avoid the high traffic link. It won't detect the cross link if any traffic contains in that way while packet delivering to destination point. The proposed geographic routing algorithm, Spanning Tree Based On Greedy Distributed Routing (STGDR), this routing algorithm finds best shorter routes path and generates less traffic compare with existing location based routing algorithms. A multi hull tree designed based on multi spanning tree where each node has a related multi convex hull that contains within it the locations of all its successor nodes in the tree. Multiple hull trees deliver a way of gathering location information and they are built by convex hull information up the tree. This routing information is used in routing to avoid paths null path tree; instead that able to traverse a significantly reduced subtree, consisting of only the nodes with convex hulls tree that contain the destination node point uses new caring of multi spanning tree, which called multi hull tree, for use in networks where each node has an allocated coordinate. The experimental result shows the routing to avoid paths that will not be productive; instead it is able to traverse a significantly reduced subtree, consisting of only the nodes with convex hulls tree that contain the destination node point.

**Keywords:** Spanning Tree Based On Greedy Distributed Routing (STGDR), Wireless Sensor Network (WSN), Delaunay Triangulation (DT).

## I. INTRODUCTION

A Wireless Sensor Network (WSN) is intended for monitoring an environment. The main task of a wireless sensor node is to sense and collect data from a certain domain, process them and transmit it to the sink where the application lies. However, ensuring the direct communication between a sensor and the sink may force nodes to emit their messages with such a high power that their resources could be quickly depleted. Therefore, the collaboration of nodes to ensure that distant nodes communicate with the sink is a requirement. Sensor networks are networks of small embedded low-power devices that can operate unattended to monitor and measure different phenomena in the environment. Distributed shortest-path routing protocols for wired networks either describe the entire topology of a network or provide a digest of the topology to every router. They continually update the state describing the topology at all routers as the topology changes to find correct routes for all destinations. Hence, to find routes robustly, they generate routing protocol message traffic proportional to the product of the number of routers in the network and the rate of topological change in the network. Current ad-hoc routing protocols, designed specifically for mobile, wireless networks, exhibit similar scaling properties. It is the reliance of these routing protocols on state concerning all links in the network, or all links on a path between a source and destination, that is responsible for their poor scaling. In wireless sensor networks with the traditional routing algorithms, the problem of traffic is occurred while finding the routes to the end point. Also in these networks, each node requires the full topological information of the network. Geographic routing algorithms have been proved as a promising successor of ad-hoc routing techniques, because in this network it is not needed that every node to know about the network's topology. But they suffer with the problem that traffic routed to the same region always follows the same path. In networks comprised entirely of wireless stations, communication between source and destination nodes may require traversal of multiple hops, as radio ranges are limited. Traditional approaches to the routing problem of finding a sequence of hops between the originator of a packet and the packet's destination work by modeling the network as a graph, and computing all-pairs shortest paths on the edge weights of this graph. These distributed shortest-path routing algorithms describe the entire topology of the network, or a digest of it, to all routers on the network to find correct routes. The proposed algorithm solves the existing problem Geo-cast is most efficient over a minimal spanning tree that includes all the nodes in the target region. GDSTR

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 6, June 2018

and trap array does not attempt to build a minimal tree; it simply uses its existing hull trees. When the network is large, the global hull trees are large and hence while geo-cast over these trees achieves correctness, it is not an ideal technique. It would be preferable to geo-cast over smaller trees that cover an area that is just a little larger than the target area. Spanning tree based on greedy distributed routing (STGDR) use of local hull trees presents with an opportunity to improve geo-cast efficiency.

## II. RELATED WORK

There were several prior proposals to apply Delaunay Triangulation (DT) to geographic routing. None of them addressed the underlying technical issue that the DT graph of a wireless network is, in general, not a subgraph of its connectivity graph. In, requirements are imposed on the placement of nodes and links in 2D such the DT graph is a subgraph of the connectivity graph. In other approaches, the restricted DT graph and the k-localized DT are both approximations of the DT graph. These graphs were shown to be good spanners with constant stretch factors. However, being DT approximations, they do not provide guaranteed delivery. Furthermore, they were designed for nodes in 2D with connectivity graphs restricted to unit disk graphs.

i. **Shortest-Path Routing:** Here classical approach, a distributed form of Dijkstra's shortest path algorithm, such as Distance-Vector and Link-State, is used to find shortest paths. Every node stores its next hop on shortest path to every other node. For a network with nodes, this requires per-node state and message traffic, posing a great challenge to the network's scalability. To reduce the overhead, an on-demand route discovery approach is proposed. The algorithm uses flooding to find a route when needed, so intermediate nodes do not need to maintain up-to-date routing information. However, for applications requiring frequent route discovery, approach incurs excessive traffic.

ii. **Near-Shortest-Path Routing:** Algorithms in this category no longer insist on shortest paths, but instead try to bound the stretch with a small number, using significantly reduced per-node state. A systematic study on this problem is provided by the compact routing theory, which is primarily concerned with the fundamental stretch-state tradeoffs of general graphs. Since Peleg and Upfal's seminal work, numerous algorithms have been proposed to achieve different points in the tradeoff space for an overview. A few proposals have recently looked at how to translate the centralized algorithms into distributed and implementable protocols. For example, the S4 protocol, based on Thorup and Zwick's stretch 3 routing scheme (the TZ scheme), realizes compact routing on wireless ad hoc networks, with an emphasis on distributed control and failure resilience. Ford evaluates an alternative distributed version of the TZ scheme with stretch and state. Other work has focused on different aspects of real systems such as flat names, mobility, etc. Virtual Ring Routing (VRR) maintains a logical network using techniques from DHTs. It uses roughly per node state, but does not provide a bound-on stretch for general topologies.

iii. **Hierarchical Routing:** Hierarchical routing (HR) attempts to reduce per-node state by recursively grouping nodes into clusters. An early proposal Landmark Routing (LR), for example, uses a hierarchical set of landmark nodes that periodically send scoped messages for route discovery. In LR, a node only needs to hold state for their immediate neighbors and their next hop to each landmark. Other designs following a similar principle include Safari and LANMAR. Recently, an implementation confirms the practicality of HR in wireless sensor networks. With respect to stretch, HR suffers from the boundary effect, where two nodes nearby may fall in different clusters, so their route may have to go a long way through cluster heads, resulting in a large detour and hence unbounded stretch.

iv. **Geometric Routing:** Geo-routing algorithms mainly differ in the way local minima are dealt with. It classifies the existing algorithms into four categories:

1) Localized approach, in which the algorithm routes off local minima with strictly local information, typically using a face routing method. The best-known algorithms in this category are GFG, GPSR, and GOAFR+.

2) Abstracting approach, in which the algorithm uses non-localized data structures to abstract the network's geometry, so as to provide guidance for the algorithm to avoid routing traps. Examples include GDSTR, MDT, and MAP.

3) Embedding approach, in which the algorithm embeds the network into a metric space. By the space, every node is assigned a set of coordinates on which geometric routing can be performed. The coordinates can be landmark-based (e.g., BVR, LCR, HopID, Euclidean (e.g., NoGeo), polar (e.g., GEM), or hyperbolic (e.g., hyperbolic routing).

4) Partitioning approach, in which the algorithm divides the network into relatively regular pieces where the geometry is simpler and hence greedy forwarding is efficient. Examples include GLIDER and recent proposal DRP. In parallel to the efforts on compact routing, another line of research, namely geometric routing (or geo-routing), has considered specialized routing methods for wireless networks by taking advantage of the nodes' geometric positions. In this approach, it is assumed that every node knows its own position, and the source of a message knows the position of the destination (through for example a distributed hash table). The algorithm forwards packets in a greedy manner by selecting next hops that are progressively closer to the destination. When the packet encounters a local minimum (LM) and cannot move forward, a recovery scheme is executed. The defining characteristic of geo-routing is that its

performance depends on the network's geometric properties. The Greedy Perimeter Stateless Routing algorithm: greedy forwarding. In this chapter, we define the greedy forwarding rule; define a simple beaconing protocol for nodes to learn their neighbors' positions; identify the desirable properties of greedy forwarding; define the topologies on which greedy forwarding fails; and characterize the frequency of greedy forwarding failure by the density of nodes in a network. The failure of pure greedy routing to find paths in the presence of voids, by introducing two perimeter traversal algorithms for forwarding packets around voids. After describing the right-hand rule for traversing a graph now present the full Greedy Perimeter Stateless Routing algorithm, which combines greedy forwarding on the full network graph with perimeter forwarding on the planarized network graph where greedy forwarding is not possible. Recall that all nodes maintain a neighbor table, which stores the addresses and locations of their single hop radio neighbors. This table provides all state required for GPSR's forwarding decisions, beyond the state in the packets themselves.

## III. PROPOSED APPROACH

The proposed geographic routing algorithm, Spanning Tree Based On Greedy Distributed Routing (STGDR), this routing algorithm finds best shorter routes path and generates less traffic compare with existing location based routing algorithms. A multi hull tree designed based on multi spanning tree where each node has a related multi convex hull that contains within it the locations of all its successor nodes in the tree. Multiple hull trees deliver a way of gathering location information and they are built by convex hull information up the tree. This routing information is used in routing to avoid paths null path tree; instead that able to traverse a significantly reduced subtree, consisting of only the nodes with convex hulls tree that contain the destination node point uses new caring of multi spanning tree, which called multi hull tree, for use in networks where each node has an allocated coordinate. The experimental result shows the routing to avoid paths that will not be productive; instead it is able to traverse a significantly reduced subtree, consisting of only the nodes with convex hulls tree that contain the destination node point. In STGDR, a node will first attempt to forward a packet greedily as before. If greedy forwarding fails, it will switch to the new greedy-hull forwarding mode by using the information contained in the convex hulls of a local hull tree. By local, we mean that the tree contains only the nodes in a limited locality. Since correctness cannot be guaranteed, forwarding can sometimes fail using the local tree and in such a case, a node will switch to forwarding on one of the two original global hull trees, which is guaranteed to succeed.

### 3.1 Network Model

Unit Disk Graph (UDG): A UDG is a special instance of a graph in which each node is identified with a disk of unit radius r=1, and there is an edge between two nodes u and v if and only if the distance between u and v is at most 1. The model is depicted. Each node's transmission range is drawn as a dotted circle. The edges, which connect nodes, are drawn as straight lines. The neighbors of node u are node v, node w, node y and node z are shown in the simplified graph.
Quasi Unit Disk Graph (QUDG): In a QUDG, each node is identified with two disks, one with unit radius r=1 and other with radius q= [0,1]. It can be observed that a QUDG with q=1 is an UDG. The edges between nodes d away from each other are identified with respect to the below listed rules:
There is an edge between two nodes if d= [0,q]. There is a possible edge connecting two nodes if d=[q,1] There is no edge between two nodes if d= [1,] Lower bound for the performance of geographic routing algorithms in three and four dimensions -Lower bound graph for geographic routing algorithms. Nodes represented by solid squares lie on the surface of a sphere with mutual distance at least 2. Nodes printed as diamonds lie also on the surface and connect these points. The round(red colored) nodes lie on lines leading from the surface nodes towards the center. A single dedicated surface-node w has an extended line leading to node t in the center of the sphere.

### 3.2 Greedy Routing

The beaconing algorithm provides all nodes with their neighbors' positions: periodically, each node transmits a beacon to the broadcast MAC address, containing only its own identifier (e.g., IP address) and position. It encodes position as two four-byte floating point quantities, for x and y coordinate values. To avoid synchronization of neighbor's beacons, as observed, jitter each beacon's transmission by 50% of the interval B between beacons, such that the mean inter-beacon transmission interval is B, uniformly distributed in [0:5B; 1:5B] beacon from a neighbor for longer than timeout interval T, a GPSR router assumes that the neighbor has failed or gone out-of-range, and deletes the neighbor from its table. The 802.11 MAC layer also gives direct indications of link-level retransmission failures to neighbors. It interprets these indications identically. Greedy forwarding's great advantage is its reliance only on knowledge of the forwarding node's immediate neighbors. The state required is negligible and dependent on the density of nodes in the wireless network, not the total number of destinations in the network. On networks where multi-hop routing is useful, the number of neighbors within a node's radio range must be substantially less than the total number of nodes in the network. The position a node associates with a neighbor becomes less current between beacons as that neighbor moves. The accuracy of the set of neighbors also decreases; old neighbors may leave and new neighbors may enter radio range. For these reasons, the correct choice of beaconing interval to keep nodes' neighbor tables current depends on the rate of mobility in the network and range of nodes' radios. It shows the effect of this interval on GPSR's performance in our simulation results. By keeping current topological state for a one-hop radius about a router is the minimum required to do any routing, no useful forwarding decision can be made without knowledge of the topology one or more hops away. In particular, because of the local tree building algorithm, that each local hull tree completely covers all the nodes in a given grid square. Hence, if the target region of a geo-cast message is completely contained in a grid square, can know that it will be broadcast correctly to all the required targets within the region. Because two forests are available, even if the target region of a geo-cast message is not completely contained within a grid square for one forest, it is likely to be contained within a grid square of the other. If indeed a suitable local tree cannot be found, correctness can be guaranteed by broadcasting on one of the global hull trees as before. Because the constraints for correctness are much stricter for planarizations, it will in general require more effort to maintain a planar subgraph than a spanning tree. In fact, a distributed spanning tree has only two criteria for correctness:

1. Each node, except for the root node, has exactly one parent node.
2. Each node must be connected. It guarantees this by ensuring that every node has a common view of the root of the tree.

Both these conditions can be checked locally by a node by communicating only with immediate neighbors. On the other hand, the only known technique for detecting and eliminating non-planar edges in a connected graph requires non-local face traversals. The STGDR routing algorithm will work correctly as long as have a rooted spanning tree.

### 3.3 Minimal-Depth Spanning Tree algorithm

A hull tree is a spanning tree where each node has an associated convex hull that contains the locations of all its descendant nodes. Hull trees provide a way of aggregating location information and they are built by aggregating convex hull information up the tree. Information is used in routing to avoid paths that are not productive; instead we traverse a significantly reduced sub tree, consisting of only the nodes with convex hulls containing the destination point. Each node in basic hull tree stores information about the convex hulls that contain the coordinates of all the nodes in sub trees associated with each of its child nodes. The convex hull information is aggregated up the tree. Each node computes its convex hull from the union of its coordinate and the points on the convex hulls of all its child nodes, and communicates to the parent node. Consequently, the convex hull associated with the root node is the convex hull of the entire network and contains all the nodes in the network. The convex hull for a set of points is the minimal convex polygon that contains all the points; it is minimal because the convex hull will be contained in any convex polygon that contains the given points. The hull is represented as a set of points (its vertices), and this set could be arbitrarily.

#### 1) Minimal-Depth Spanning Tree

Determine the set of neighboring nodes that have minimal depth, i.e., are at the smallest number of hops from the root. If there is only one node in the set, choose that node as the parent.

- If there is more than one node in the set, choose the node that is closest in geometric distance to n as the parent. Closely related is the following Minimal-Path Spanning Tree.

#### 2) Minimal-Path Spanning Tree

Determine the set of neighboring nodes that have minimal path length to the root.

- If there is only one node in the set, choose that node as the parent.
- If there is more than one node in the set, choose the node that is closest in geometric distance to n as the parent.

These algorithms will produce minimal spanning trees (in terms of either path length or hops) rooted at extremal nodes. The expected advantage of these trees is that a packet is will be able to traverse the entire tree in a small number of hops or path length. The actual routing performance is related to D, the diameter of the network. The disadvantage of such trees is that when the network density is high, some intermediate nodes may end up with a large number of children. Since each child has an associated hull, the amount of state stored per node will therefore be proportional to network density, and not constant. The results with local trees are marginally better than those with only two global trees. Surprisingly, the results with three and four global trees are worse. By this can suspect that the latter is due to an artifact in the experimental setup. The effective difference in the results is small it translates to a difference in two or three packets for each geo-cast instance. The results for networks with obstacles. These results show that obstacles have a marginal effect on geo-cast performance. The fact that geo-cast with three global hull trees performs marginally worse than the rest is likely an artifact of the experimental set up, since the grid squares and target regions are squares that are aligned with the x and y axes. On the other hand, the rays that are used to choose the roots for the three global trees are not aligned in the same way. This is likely to have an effect on the orientation of the resulting hull trees.

 DOI 10.17148/IJARCCE.2018.7633 

ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
IJARCCE
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 6, June 2018

### 3.4 Algorithm Implementation

**Step 1:** Check for Geocast Mode: If p: mode = Geocast, follow step 6.

**Step 2:** Check Reached Broadcast Tree: If v has a child with a convex hull that intersects with R, follow step 5. Otherwise, follow step 3.

**Step 3:** Find Tree Mode: If p: mode = FindTree: If v is the root node for p: Tree, algorithm terminates here. Otherwise, forward p to the parent node in p: Tree.

**Step 4:** STGDR Routing: Route packet to destination t according to Algorithm. If packet is undeliverable, set p: mode: = FindTree and follow step 3.

**Step 5:** Pick Hull Tree for Geocast: If R is contained in either of the grid squares of the local hull trees, set p: Tree as the local tree (in a grid square that contains R) with a root that is closest to the t.
If the grid squares of the local hull trees do not completely contain R, set p: Tree as the global tree with a convex hull that contains R; if such a global tree does not exist, pick the global tree with a root that is closest to t. Follow step 6.

**Step 6:** Broadcast to Target Set: Determine target set B for message broadcast with respect to p: Tree according to the following rules:

If p: mode = Geocast, the node from which geocast message was originally received is not to be included in set of targets.

If p: Tree is a local tree, each neighboring node that has an associated convex hull (from v's perspective) that intersects R is added to the target set.

If p: Tree is a global tree, each child node that has a convex hull that intersects R is added to the target set. If the convex hull of associated hull tree p: Tree fully contains R and none of the conflict hulls H intersects R, do not add the parent node to target set. Otherwise, add the parent node to the target set. If p: mode 6= Geocast, set p: mode: = Geocast. Broadcast p to all nodes in target set B. Overall, geocast with local hull trees (STGDR) incurs 10% less overhead than geocast with only two global hull trees in sparse networks with large voids. All the variants seem to perform equally well in dense networks. These results also suggest that can likely implement geocast using hull trees with no more than two times the minimum number of messages (since the Estimated Geocast Stretch is a loose upper bound).

## IV. EXPERIMENTAL RESULTS

The experimental simulations, we use a simple radio model: all nodes have unit radio range; two nodes can communicate if and only if they are within radio range of each other and if their line-of-sight does not intersect an obstacle. The simulator supports linear, polygonal and circular obstacles. Wireless losses are not simulated since our goal is to compare the basic algorithmic behavior of STGDR to other geographic routing algorithms. To understand the effects of network density on routing performance and maintenance costs, we generated networks with 25 to 500 nodes randomly scattered over a 100x100 unit square. This process generated networks with average node degrees between 0.7 to 14.4. For each density, we generated 200 networks, and then routed 20,000 packets using each algorithm between randomly chosen pairs of source and destination nodes. The performance measurements presented are the average over the 200 times 20,000 data points. We also used these topologies to evaluate the effects of parameters like the number of hull trees and the value of r, the maximum size for the convex hulls.

Table 1: Performance under STGDR Model with Connection Stretch

| Protocol | Connection Stretch | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **10** | **20** | **30** | **40** | **50** | **60** | **70** | **80** |
| GPSR | 2.0 | 8.4 | 9.1 | 9.7 | 10.9 | 12.5 | 15.4 | 18.6 |
| Trap Array | 1.8 | 4.1 | 6.2 | 7.9 | 8.1 | 10.0 | 12.1 | 14.2 |
| STGDR | 1.2 | 1.4 | 1.4 | 1.5 | 1.7 | 1.8 | 1.9 | 2.1 |

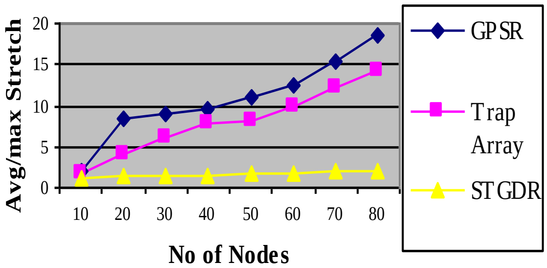

Figure 1: Comparison of different connection Stretch with no of nodes

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 6, June 2018

Table 2 Performance under the STGDR R model with Avg/State per Node

| **Protocol** | **No of Nodes** | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | **10** | **20** | **30** | **40** | **50** | **60** | **70** | **80** |
| GPSR | 45 | 58 | 61 | 74 | 81 | 88 | 91 | 94 |
| Trap Array | 20 | 37 | 48 | 52 | 69 | 74 | 82 | 85 |
| STGDR | 10 | 17 | 24 | 32 | 41 | 47 | 53 | 59 |

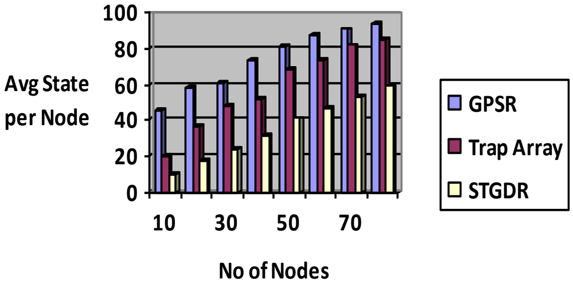


Figure 2: Comparison of different state per node with k

Table 3: Performance under the obstacle model with Stretch

| Protocol | No of Obstacle | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 2 | 4 | 6 | 8 | 10 | 12 | 14 | 16 |
| GPSR | 43 | 48 | 49 | 52 | 54 | 57 | 59 | 61 |
| CGR | 21 | 25 | 28 | 31 | 35 | 38 | 41 | 44 |
| GDSTR | 12 | 15 | 18 | 21 | 26 | 28 | 32 | 36 |

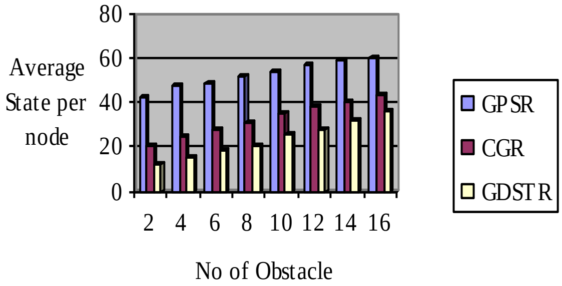


Figure 3: Comparison of different obstacles state

Table 4: Performance under different traffic Load

| **Protocol** | **No of Obstacle** | | | | |
|---|---|---|---|---|---|
| | **2** | **4** | **8** | **16** | **32** |
| GPSR | 643 | 1043 | 2264 | 2866 | 3873 |
| CGR | 554 | 887 | 1056 | 1164 | 1257 |
| GDSTR | 321 | 534 | 755 | 1078 | 1135 |

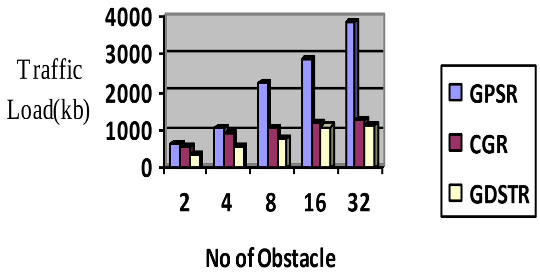


Figure 4: Comparison of different traffic load

IJARCCE
ISSN (Online) 2278-1021
ISSN (Print) 2319-5940
International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 7, Issue 6, June 2018

Table 5: Performance Transmission message cost

| Protocol | No of Nodes | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| | 10 | 20 | 30 | 40 | 50 | 60 | 70 | 80 |
| GPSR | 153 | 175 | 187 | 196 | 254 | 289 | 297 | 321 |
| CGR | 141 | 154 | 164 | 176 | 196 | 234 | 265 | 298 |
| GDSTR | 123 | 134 | 157 | 163 | 173 | 186 | 197 | 211 |

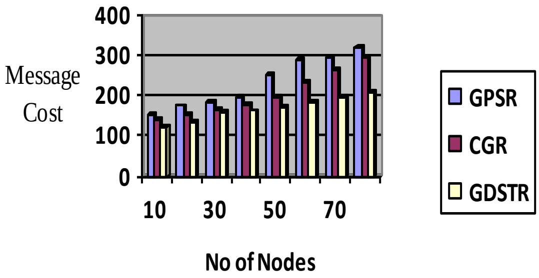


Figure 5: Comparison of different Message Cost

## V. CONCLUSION AND FUTURE ENHANCEMENT

The proposed geographic routing algorithm, spanning tree based on greedy distributed routing (STGDR), this routing algorithm finds best shorter routes path and generates less traffic compare with existing location based routing algorithms. A multi hull tree designed based on multi spanning tree where each node has a related multi convex hull that contains within it the locations of all its successor nodes in the tree. Multiple hull trees deliver a way of gathering location information and they are built by convex hull information up the tree. This routing information is used in routing to avoid paths null path tree; instead that able to traverse a significantly reduced subtree, consisting of only the nodes with convex hulls tree that contain the destination node point uses new caring of multi spanning tree, which called multi hull tree, for use in networks where each node has an allocated coordinate. The future work of mobility on STGDR is another area that remains to be explored. In particular, it may be useful to see if variants of STGDR can be developed for heterogeneous networks, consisting of a mixture of both mobile and stationary nodes, and where individual nodes have differing amounts of available energy.

## REFERENCES

[1] M J. Bruck, J. Gao, and A. Jiang, "MAP: Medial axis based geometric routing in sensor networks," in Proc. MobiCom, 2005, pp. 88–102.
[2] M. Caesar, M. Castro, E. Nightingale, G. O'Shea, and A. Rowstron, "Virtual ring routing: network routing inspired by DHTs," in Proc. ACM SIGCOMM, 2006, pp. 351–362.
[3] L. Cowen, "Compact routing with minimum stretch," J. Algor., vol. 38, no. 1, pp. 170–183, 2001.
[4] T. Eilam, C. Gavoille, and D. Peleg, "Compact routing schemes with low stretch factor," in Proc. ACM PODC, 1998, pp. 11–20.
[5] R. Fonseca et al., "Beacon vector routing: Scalable point-to-point routing in wireless sensornets," in Proc. NSDI, 2005, vol. 2, pp. 329–342.
[6] B. Karp and H. T. Kung, "GPSR: Greedy perimeter stateless routing for wireless networks," in Proc. MobiCom, 2000, pp. 243–254.
[7] R. Kleinberg, "Geographic routing using hyperbolic space," in Proc. IEEE INFOCOM, 2007, pp. 1902–1909.
[8] S. S. Lam and C. Qian, "Geographic routing in d-dimensional spaces with guaranteed delivery and low stretch," in Proc. SIGMETRICS, 2011, pp. 217–228
[9] A. Rao, S. Ratnasamy, C. Papadimitriou, S. Shenker, and I. Stoica, "Geographic routing without location information," in Proc. MobiCom, 2003, pp. 96–108.
[10] G. Tan, M. Bertier, and A-M. Kermarrec, "Convex partition of sensor networks and its use in virtual coordinate geographic routing," in Proc. IEEE INFOCOM, 2009, pp. 1746–1754.

## BIOGRAPHY

**Dr. R. Deepa** received her B.Sc statistics from P.S.G College of Arts and Science College, Coimbatore, India. She had her Master of Computer Applications From Bharathidasan in teaching. She is presently working as an Assistant Professor in NGM College, Pollachi. Her research interest includes Data Mining,Big Data Management, and Image Compression. Now she is pursuing her Ph.D Computer Science in Dr.Mahalingam Center for Research and Development at NGM College Pollachi.