Location Management in Sparse Ad hoc Networks

Sumesh J. Philip
Department of Computer Science
Western Illinois University
Macomb IL 61455
Email: sj-philip@wiu.edu

Joy Ghosh, Chunming Qiao
Department of Computer Science and Engineering
SUNY at Buffalo
Amherst NY 14260
Email: {joyghosh,qiao}@cse.buffalo.edu

Abstract — *Geometric routing using source–destination locations has been suggested as a scalable alternative to conventional routing approaches in mobile ad hoc networks. Prior studies have shown that the location of a destination can be found efficiently in large/dense ad hoc networks using intelligent location management schemes by recruiting nodes in specific unit regions of the terrain as location servers. In this work, we show that certain location management protocols that use a grid based approach suffer from the empty server region problem and that their performance can be seriously degraded with decreasing node density in sparse or irregular ad hoc networks. In order to tackle this problem, we introduce proxy based location management, a novel enhancement that can be used in conjunction with existing location management protocols to operate efficiently in sparse or irregular ad hoc networks. Extensive simulations show that proxy based location management combined with routing on an overlay graph constructed from the unit regions operates more effectively in sparse networks than SLURP/GPSR, an existing location management scheme and a geometric routing protocol that routes packets on a planar graph extracted from the unit disk graph.*

Keywords: Ad hoc networks, Location management, Geometric routing, Planar graph routing

1 Introduction

Geometric routing (also known as position/location based or geographic routing) has been widely suggested as an alternative to conventional routing approaches in mobile ad hoc networks to achieve routing scalability. In geometric routing, it is assumed that mobile nodes are aware of their own location via the use of a GPS receiver or other localization schemes. A localized periodic broadcast protocol enables all nodes to have approximate knowledge of their neighbors' locations. Several proposals have been described in literature that make use of this neighborhood position knowledge to route packets to a known location of the destination. Most of these schemes are greedy in nature in that they minimize a metric locally to carry out routing at each intermediate node. The metric to be optimized can be the remaining distance to the destination [1], [2], progress [3], [4], or direction [5].

One of the major attractions to position based routing is its localized nature of operation. While existing ad hoc routing protocols make use of source routes (e.g. Dynamic Source Routing [6]) or state based route construction/maintenance (e.g. DSDV [7], AODV [8]), these routes are highly error prone due to node mobility or the unpredictable nature of the wireless channel. On the other hand, in geometric routing, a mobile node only needs to know the destination's location, location of neighbor nodes in its locality (radio range) and its own location in order to make a sensible routing decision. Since the position of a node's neighbors is conveyed through periodic broadcast messages, this information is readily available at each node.

However, it is well known that greedy geometric schemes suffer from the local maxima problem, in which the metric to be optimized cannot be improved further at an intermediate node with respect to its neighbors. In such cases, packets are dropped at the intermediate node even if a perfectly valid path exists between the source node and the destination node. To recover from the error caused by greedy forwarding, face routing on a planar graph extracted from the static wireless network was considered in [9], [10]. The wireless network is modeled as a geometric *unit disk graph* (UDG), in which the Euclidean co-ordinates of the mobile node represents a vertex in the plane, and an edge exists between two vertices if the Euclidean distance between the vertices is less than the unique node transmission range R. In general, the UDG is not planar, but known techniques [11], [12] can be used to extract a sub-graph from a connected UDG which is both planar and connected. Face routing on the planar graph then guarantees the delivery of packets in a static wireless network.

A key obstacle in geometric routing is that a destination node's current location needs to be discovered before packets can be routed via position based routing. The problem of managing locations in a distributed manner such that the location of any node can be discovered prior to routing is known as the *location management* problem, and has been studied exhaustively by researchers. Several efficient solutions for this problem are available in literature. However, we note that certain techniques that divide the terrain into logical unit regions and delegate location service duty to nodes located in the unit regions require significant node densities to operate correctly. Under low node density conditions, location update/query packets will reach *empty server regions*, and these packets will ultimately be dropped by the forwarding nodes since they cannot be delivered to their desired destinations. In such cases, nodes that are registered to these regions will be temporarily disconnected from the network, even though there are perfectly valid paths that connect them to the ad hoc network.

To our knowledge, there has not been any prior study that analyzed the effect of node density on the performance of location management protocols for mobile ad hoc networks. In this paper, we propose a novel proxy based scheme for location management to overcome the problem caused by insufficient node density, whereby a server in a nearby region is delegated the responsibility of location management for the empty region. A related protocol was introduced in [13], but may encounter protocol incorrectness due to race conditions during distributed operation. Thus, even if the notion of a proxy is simple, the inherent difficulty in applying a distributed proxy mechanism in ad hoc networks makes this problem interesting and challenging.

The rest of this paper is organized as follows: Section 2 briefly describes the role of location management in geometric routing and the empty server region problem caused by irregular node distribution or low node density. We propose a proxy based enhancement that can be used with current location management protocols to tackle the empty server region problem in Section 3. We describe our simulation study and comment on the results in Section 4 and conclude this work in Section 5.

2 Location Management and the Effect of Node Density

One of the main components of geometric routing is location management, with which the current location of the desired destination is discovered before geometric routing can begin. To be deemed scalable with respect to geometric routing, the overhead due to the location management protocol must be kept minimal so that the performance of the routing protocol is minimally affected. Many location management schemes have been proposed in literature, and a general outline of a class of these schemes is as follows:

- Divide the terrain into well ordered unit regions. Regions may be flat [14], [15], or aggregated for the purpose of routing scalability [16], [17], [18], [19]. Each node selects one region as its *server* region. The mapping between a node and its server region is unique so that other nodes who wish to know a node's server region can easily do so using the unique mapping function.
- Each node carries out the *update* (or registration) phase, in which update packets, containing the current location of the node, are geometrically routed to server regions. The location of the server region is indicated by a unique point inside the region (such as the lower left corner or the center). The update phase can be triggered by a timer (periodic) or by node mobility (crossing grid regions). In a sub-category of these techniques ([14], [17], [18]), when a node that resides in the server region receives the update packet for the first time, it carries out a region wide *geocast* to update all other nodes that are resident in that region of the updating node's current location. Thus, mobile nodes that are currently resident in a server region form the location *servers* for all the nodes registered to that region.
- When a node moves into a new unit region, it also carries out a *maintenance* phase, in which it requests nodes already present in that region to forward new location information that it must store as part of its server duty.
- Finally, when a source node needs to find the location of a destination node, it queries the destination node's server region using a *query* packet. The query phase is terminated by the *response* phase, in which the first node to receive the query will respond with the latest known location of the destination node. Data is then routed to the destination node by the source node using this location and a geometric routing protocol.

As noted in [20], the class of protocols described in [14], [17], [18] were found to operate efficiently under high network density conditions, with greedy forwarding being chosen as the position based routing protocol. In greedy forwarding, an intermediate node chooses its next hop from amongst its neighbors who is closest to the destination's location than itself. If no such neighbor is found, then the packet is dropped. Under high density conditions, greedy forwarding usually finds a candidate neighbor, and most packets are routed correctly to their respective destinations.

However, in a sparse network, both the location management scheme as well as greedy forwarding becomes inefficient and can be problematic. More specifically, since most of the proposed location management schemes delegate the location service task to particular unit regions in the topography, it is imperative that there is *at least* one node in the location server region and that this region be accessible to the network. Under low density conditions, unit regions may not contain any nodes. We denote such a condition, in which a location server is devoid of member nodes by the term *empty server region*. Intuitively, the probability that a unit region becomes empty becomes significant with decreasing network density and high node mobility.

In addition, greedy forwarding can fail due to the occurrence of local maxima during the forwarding process. Hence, even if a valid path exists to the location server, the packet will be dropped by the intermediate node at which the local maxima occurs. Note that routing protocols that route along the faces of planar graphs such as Greedy–Face–Greedy (GFG) [9] and the Greedy Perimeter Stateless Routing (GPSR)[10] or depth first search based approach such as the Geographic Routing Algorithm (GRA)[21] can be used to overcome this problem. Thus, the bottleneck in position based routing in sparse networks seems to be the incorrect operation of the location management scheme, and needs to be rectified for efficient geometric routing.

3 Proxy based Location Management

As a novel solution, we propose to use a proxy–based location management scheme to address the empty server region problem. If a server region R_E is empty, then the duty of that server needs to be delegated to a *proxy* server R_P under the proxy based scheme. Although the concept of proxies is simple, the task of distributed proxy selection and management is non trivial in ad hoc networks due to the inherent tradeoff between complexity and control overhead. Consider the following two schemes: in the first scheme, a node mapped to an empty server region R_E simply selects the first non–empty grid as its proxy server R_P, and a network–wide broadcast is carried out to make all nodes aware of the new server. All future updates and queries are routed directly to R_P. In the second scheme, after R_P is selected as a proxy for R_E, no network–wide broadcast is done, but instead all location management packets routed towards R_E are redirected to R_P via the proxy selection scheme. While the first scheme is conceptually simpler, it incurs significant overhead. On the other hand, the second scheme is resource efficient, but is more complicated. For example, if some time later R_P itself becomes empty, then the protocol has to readjust the mapping without creating *race conditions* that adversely affect the protocol operation.

In the following section, we outline our proxy based location management scheme. We assume that there is a connected planar graph defined on the non–empty regions of the terrain (referred to as an overlay graph) and a routing protocol such as GTA [22] that can traverse the overlay graph faces. The routing protocol can detect empty regions when regular packet forwarding fails, and switches from the regular forwarding mode to the face routing mode to get around empty regions. We also assume that the empty region is within a single face and that the graph faces remain stable with respect to packet routing time (graph face changes that occur in between the time two packets are routed do not affect the proper routing of packets by the protocol). We note that there is a MAC protocol that can reliably transmit a packet. All protocol packets from a node are uniquely identified by a sequence number. We use the term location server loosely to indicate any node who is a member in an unit region, since all nodes in an unit region carry out the task of location management.

3.1 Proxy Selection

Proxy selection refers to the set of actions that need to be taken when update packets from mobile nodes to an empty server region need to either set up a proxy server (if one has not yet been set up) for that region, or be routed to a proxy server (if a proxy server has already been defined). The proxy selection phase for region R_E is started by an unit region R when greedy forwarding of an update packet to the location server region R_E fails in R. It enters this update into a temporary database, indexed by the destination region R_E, in the case that R may have to be the proxy server for R_E later on. R then retransmits the update in face routing mode along the face intersected by the straight line connecting the centers R and R_E. A flag is turned on in the header to indicate that the packet is to be routed via face routing, along with the grid identifier R to indicate where the face route mode was initialized. A proxy *timer* is also started for this entry whose timeout indicates that the packet either resumed the regular forwarding mode or reached its intended destination (either R_E or a proxy that was already set up for R_E). The value of the timeout is set to the time required to traverse the terrain perimeter. When the timeout occurs, all entries in the temporary database are deleted.

R also snoops into location update packets in face routing mode, and if R notices that it has to forward an update packet to any of the grids in its temporary database in face routing mode, it checks the face route start point (unit region identifier) of the packet, as recorded by the routing protocol. If the Euclidean distance between the closest point of that region to the center of the empty server region is smaller than its own, or if the distances are equal but the identifier of the start point is greater than its own, then it stops the timer, deletes the temporary database and continues the packet's routing. This rule avoids potential race conditions that may occur during proxy selection as indicated by Figure 1. If two location updates for empty server region R_E arrive at regions R_{10} and R_8 simultaneously, there is a possibility that we may have *two* proxy servers for R_E. However, R_8 has a lower identifier than R_{10}, and gives up its candidacy to be a proxy for R_E. Packets that loop around the graph faces are handed up by the routing protocol to the location service layer. If all copies of a previously retransmitted update looped around at R, it means that R_E is indeed empty, and that a proxy has not yet been assigned for R_E. R then assumes itself to be the proxy server for R_E, stops the set up process, and copies all entries for R_E from its temporary database into its proxy table. All future updates to the empty server region R_E are entered into the proxy table of the proxy server R.

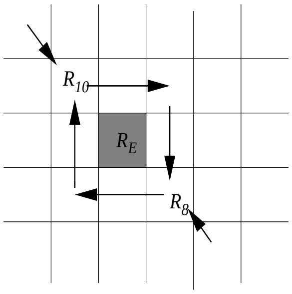


Fig. 1. Potential race condition during proxy selection

Based on the above discussion and the observation that an empty server region R_E lies inside an interior face or outside the exterior face of the grid graph, we have the following theorem and its informal proof.

Theorem 1: In a connected network, the selection phase creates exactly one proxy grid R_P for an empty server region

R_E and lies in the graph face containing R_E.

Proof: Face routing guarantees that the retransmitted update packets will visit all edges of the face containing the empty server region, and select one of the adjacent grids closest to R_E that lies on the graph face. Moreover, if the update packet started its journey via face routing due to a temporary void, then regular forwarding resumes at an intermediate node that continues to route the packet to either the server region, or to the face containing the empty server region. Finally, race conditions that may create multiple proxies for R_E are avoided by either the distance rule or the unit region identifier rule. ■

3.2 Proxy Delegation and Maintenance

Proxy delegation is required when a node is about to leave a proxy server region R_P empty and move into a new region R_x or if such a node is about to be shut down (either by an user or due to low battery power of the device). If a location or proxy server becomes empty due to mobility, then a new proxy server has to be selected which is closest to the original empty region R_E and which lies within the face containing R_E. To this end, the last node to move out of R_P (when two or more nodes move out simultaneously, the tie can be broken by using node addresses) starts the proxy delegation phase by transmitting a *delegation* message in face route mode with R_E as the destination and R_x as the grid where face route mode was initiated. If the message loops at R_x, then R_x is delegated as the new proxy server for R_E and the message is discarded. On the other hand, if R_x notices that the message returned with a closer grid $R_{x'}$ to R_E than itself, it forwards all the entries for R_E to $R_{x'}$, and delegates $R_{x'}$ to be the new proxy server for R_E. In case that the proxy delegation phase was initialized due to low battery power, the process of server selection is carried out by excluding R_P as a potential candidate.

Similarly, proxy maintenance is required when a region R_x, which was hitherto empty, now contains a node u which just moved into it from an adjacent grid. If R_x was a server sometime before, then u must claim all location entries for which R_x was a server prior to it being empty. Similarly, if R_x partitions/closes a previously close/open face in the planar graph, then u must claim all entries for an empty server region along the graph face which R_x just closed, from its proxy. This can be accomplished as follows. Similar to the proxy selection phase, node u starts the proxy timer and transmits a proxy *maintenance* message along all its graph edges in face route mode. All proxies promiscuously listen to maintenance messages, and a server responds to it with a proxy *response* if this grid is currently a proxy for R_x or if R_x is closer to any of the grids that it is currently a proxy for, than itself. Moreover, a node in R_x caches any update, query or maintenance message that it forwards in face routing mode if it is currently seeking a proxy response, indicated by an active timer. In the event that a proxy response is received, the grid is updated with the new location entries, and all previously cached messages are responded to, if possible.

It is worth noting that the proxy delegation and maintenance phases preserve the face containing the empty server R_E such that its proxy R_P is also along the same face.

3.3 Location Discovery

A node y that wishes to find a destination node x's current location sends a location query to x's known server R_x, by using the unique mapping between x and R_x. The query is then routed until it reaches either R_x, or its proxy R_E. The server that receives the query responds with a location response with node x's current location. We have the following theorem and its informal proof to establish the correct operation of the proxy location management scheme.

Theorem 2: Assuming a reliable MAC protocol, and a routing protocol that can visit all unit regions along the face of a planar overlay graph, the location query is guaranteed to return the last known location of a queried node in a connected network.

Proof: Since all phases of the proxy management result in the proxy server lying along the face containing the empty server, the query will be responded to as long as the query packet reaches the face under consideration. Similar to an update packet, a query packet is forwarded until it reaches the graph face containing the location server (and hence its proxy). Thereafter, face routing guarantees that the packet visits the appropriate server who can respond to the query. A race condition could have occurred if a query had passed through a region which was in the process of executing the maintenance phase. However, since any region which is currently waiting for a maintenance response caches every packet that passes through it, the query will be answered eventually. ■

4 Numerical Study

To test the performance of our proxy management scheme, we implemented our proxy enhancements on SLURP [14], a simple and flat location management protocol in GloMoSim [23]. The location management layer was built in to the TCP/IP protocol stack that operated in conjunction with IP as the network layer protocol. Server selection in SLURP is based on a random function which maps the node address to one of the grids in the terrain. Main data structures in the location management layer consist of i) a *location table* and ii) a *neighbor table*. When a location server node receives a location update packet from a node, the current location of that node is updated in its location table. A periodic broadcast protocol enables each node to realize its local connectivity, and records it in the neighbor table to assist in geographic routing.

We used GTA (Grid Traversal Algorithm) [22] as the routing protocol for SLURP with the proxy enhancement. GTA is a face routing algorithm which routes on a planar overlay graph (an overlay graph is defined on the unit regions such that the unit regions form the vertices of the graph and edges exist

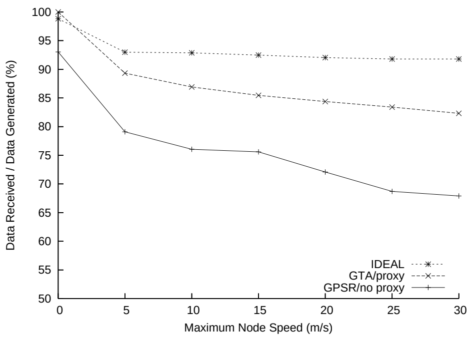


Fig. 2. Fraction of successfully received data

between the vertices if they are directly connected through radio nodes), and is better suited for the proxy operations. Since our objective is to test the effectiveness of our proxy enhancement, we compared the scheme against a combination of GPSR (as the routing protocol) and SLURP without using the proxy enhancement. GTA has an adjacency vector table to keep track of adjacent vertices, and a list of next best hops to reach each adjacent vertex. GPSR implementation was borrowed from the NS–2 implementation of the protocol at [24] and verified against published results by running the protocol using an ideal location management layer. The location of a destination node is known *a priori* using an *ideal* location management scheme, and serves as an upper bound for the performance of a practical location management protocol. The RNG scheme [12] was used to create the planar graph for perimeter routing in GPSR, since this scheme yields a less densely connected graph and leads to better performance of the routing protocol.

We ran our simulations on a 2000x2000 m terrain consisting of 70 mobile nodes (average density of 1.75×10^{-5} nodes/m^2), in which the unit region is a $250m \times 250m$ square region. Although this is not a very sparse network, choosing scenarios that are even sparse may lead to frequent network disconnections and possibly meaningless results. For the simulation scenario, 1000 Constant Bit Rate (CBR) connections were randomly generated, with each session sending one packet with a 512 byte data payload. A session terminates successfully if the location discovery phase returns the correct location so that the data sent to the said location successfully reaches the destination before the simulation ends. Simulation parameters for the scenario are shown in Table I. Each plot point presented in the next section is an average of seven simulation runs.

Figure 2 shows the fraction of data packets that were correctly received by destination nodes using both GPSR and GTA. The ideal location management plot serves as an upper bound for the maximum performance achievable by either routing protocol, since the exact location of a destination node is known as soon as the data packet arrives at the network layer. Clearly, the proxy based scheme is effective in tackling the empty server region problem as shown by the increase in the fraction of data packets received by either routing protocol over that of GPSR without the proxy enhancement. As the average node speed increases, link changes occur more frequently, and fluctuations in the planar graph causes temporary loops, resulting in packet drops. Since the location of the destination needs to be found before data can be forwarded, the fraction of successfully received data packets is directly proportional to the success of the location discovery phase. As shown by Figure 3(b), the success ratio of queries is slightly more for GTA, indicating a more resilient proxy discovery as well as more stable planar graph for GTA. Thus, GTA combined with our proxy enhancement performs better than GPSR without proxy in terms of the fraction of data packets successfully received.

Figure 3(a) shows the average end–to–end delay experienced by data packets. When the network is static, local maxima in greedy forwarding causes packets to be routed in face route mode most of the time, causing additional delay. However, node mobility removes some of the voids that were present in the static network, causing more packets to be forwarded greedily. Additionally, loops due to mobility cause face traversal to be unsuccessful. In general, successful face traversal also produces longer paths than greedy routing. Thus, the measurement does not take all unsuccessful long paths produced by face traversal into account which finally leads to shorter end-to-end delays and explains why there is a drop in data delay for both the plots. While RNG based routing is quite effective in GPSR, the 2–hop adjacency vector based routing and reduced control overhead (see Figure 5(b)) assists GTA to obtain a slightly lower average end–to–end delay for data packets. The increased delay towards the end of the plot is due to a higher fraction of successfully received data.

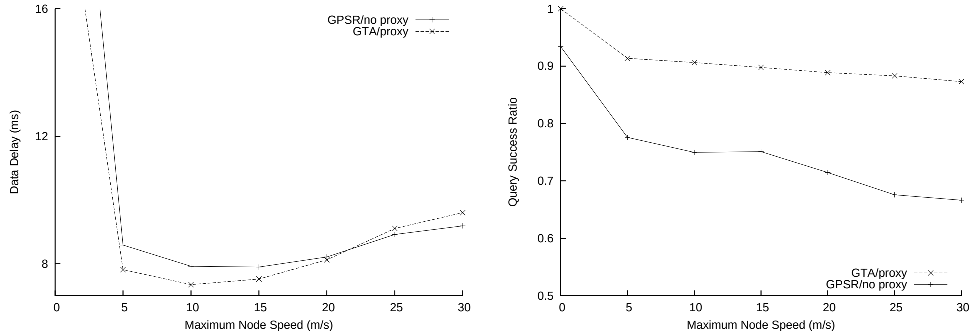
Figures 3(b), 4(a) and 4(b) show the performance of the location discovery process using the proxy scheme over the regular location management scheme. We note from Figure 3(b) that not all the queries are responded to in either schemes. Although face routing guarantees packet delivery when the graph faces remain stable and connected, this is no longer true when increased node mobility changes faces of the graph. Thus, packets that are in transit start looping using face routing in GTA as well as GPSR if graph faces change, and are dropped eventually. Location discovery can fail either when the query itself is dropped or when a previous update from a node was dropped, and the server failed to respond to the query correctly. Thus, query success ratio is below 100% even for GTA. However, as shown by the relative performance in Figure 3(b), the proxy scheme fares better in responding to location query packets than GPSR without proxy, which fails to answer queries for nodes who have empty servers. Additionally, GTA is effective in finding proxy servers using the overlay subgraph, and is able to route queries faster, as indicated by the delay in hops as well as seconds in Figures 4(a) and 4(b).

Figure 5(a) shows the effectiveness of the routing protocol in routing location update packets to their respective servers. Since the regular scheme does not have proxy servers standing by for empty servers, packets to empty servers simply loop

TABLE I

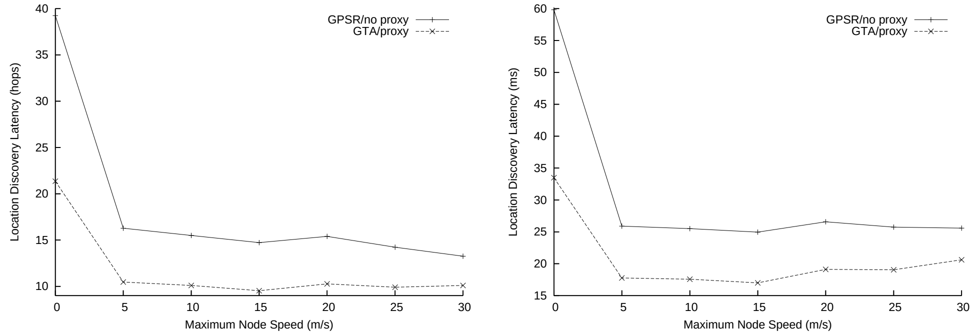
SIMULATION PARAMETERS

Simulation Time	900 sec	Mobility Model	Random Waypoint
Simulation Area	2000×2000m	Maximum Speed	0-30 m/sec
Unit Grid Size	250m	Minimum Speed	0 m/sec
Number of Nodes	70	Pause Time	0 sec
Transmission Range	350m	Traffic Type	Random CBR
Transmission Speed	54 Mbps	Number of Connections	1000
MAC Protocol	IEEE 802.11g	Data Payload	512 bytes
Beacon Interval	1 sec	Buffer Size	1000 packets



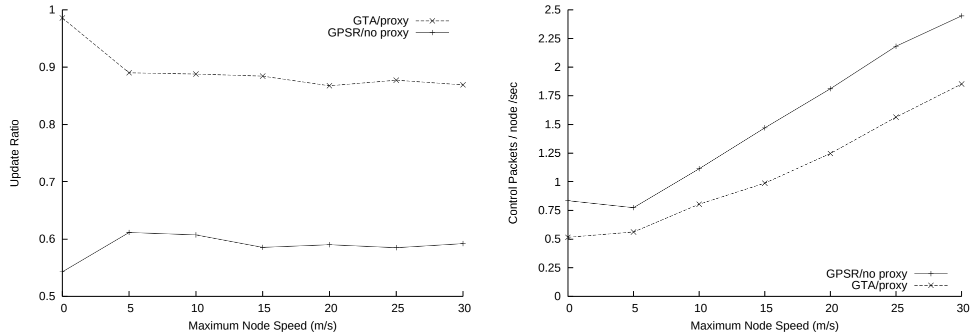
(a) Average end–to–end data delay (b) Query success ratio

Fig. 3.



(a) Location discovery latency (hops) (b) Location discovery latency (msec)

Fig. 4.



(a) Update ratio (b) Average control overhead

Fig. 5.

around before being dropped eventually. The update ratio is quite low in this scheme, indicating the potential breakdown of the location management protocol. However the proxy based scheme is able to recover from the empty server region problem, and update packets are typically routed to their respective proxy servers. Since graph faces do not remain constant in practice and changes with node mobility, a change in a graph face causes packets that are face routed to loop around that face, and are eventually dropped when their time–to–live hits the limiting value. Thus, even in the GTA scheme, update ratio is below 100%.

Figure 5(b) shows the control overhead in number of packet transmissions incurred per node per second. Since packets loop around in the absence of proxy servers, GPSR without proxy suffers from increased resource wastage. Also, since routing is based on edges of the RNG graph, control packets take longer paths and transmissions to reach their respective destinations. GTA performs better in terms of control overhead, since it is able to route packets to servers/proxy servers using shorter paths and with lower overhead.

5 Conclusion

As wireless access becomes ubiquitous, providing location based services to mobile users will become more of the norm than the exception, and user location information will be required for providing such services. Geometric routing takes advantage of location information to facilitate routing, and many such proposals have been described in literature. Location management is a key factor for geometric routing in ad hoc networks, in which location servers have to be selected to keep track of mobile nodes. In this paper, we have shown that current location management protocols that use unit region based approach suffer from the empty server region problem, which becomes pronounced when the network is sparse. We have presented a proxy based location management enhancement by delegating the task of empty server regions to adjacent non–empty unit regions.

While the concept of proxies is simple, distributed proxy selection and management can be quite complicated in ad hoc networks, and we have shown that our scheme avoids most of the pitfalls encountered while designing such a protocol. The proxy enhancement on SLURP combined with GTA, a face routing protocol that routes on a planar overlay subgraph, outperform SLURP without proxy/GPSR, a conventional location management scheme using a known face routing protocol that routes on a connected planar subgraph extracted from the unit disk graph. Our scheme is localized, resource efficient and can be easily incorporated into current location management protocols to combat the empty server region problem successfully in sparse ad hoc networks.

6 References

[1] G. G. Finn, “Routing and addressing problems in large metropolitan-scale internetworks,” *ISI Research Report ISU/RR-87-180*, March 1987.

[2] I. Stojmenovic and X. Lin, “Geographic distance routing in ad hoc wireless networks,” *University of Ottawa Tech Report TR-98-10*, December 1998.

[3] H. Takagi and L. Kleinrock, “Optimal transmission ranges for randomly distributed packet radio terminals,” *IEEE Transactions on Communications*, vol. 32, no. 3, pp. 246–257, 1984.

[4] T. Hou and V. Li, “Transmission range control in multihop packet radio networks,” *IEEE Transactions on Communications*, vol. 1, no. 34, pp. 38–44, 1986.

[5] E. Kranakis, H. Singh, and J. Urrutia, “Compass routing on geometric networks,” *Proc. 11 th Canadian Conference on Computational Geometry*, pp. 51–54, 1999.

[6] J. Broch, D. B. Johnson, and D. A. Maltz, “Dynamic Source Routing in ad hoc wireless networks,” *Mobile Computing, Kluwer Academic Publishers*, pp. 153–181, 1996.

[7] C. Perkins and P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computers,” in *ACM SIGCOMM’94 Conference on Communications Architectures, Protocols and Applications*, 1994, pp. 234–244.

[8] C. E. Perkins and E. M. Royer, “Ad hoc On-Demand Distance Vector Routing,” *Proceedings of the 2nd IEEE Workshop on Mobile Computing Systems and Applications*, pp. 90–100, February 1999.

[9] P. Bose, P. Morin, I. Stojmenovic, and J. Urrutia, “Routing with Guaranteed Delivery in Ad Hoc Wireless Networks,” *Wireless Networks*, vol. 7, no. 6, pp. 609–616, 2001.

[10] B. Karp and H. T. Kung, “GPSR : Greedy perimeter stateless routing for wireless networks,” *Proceedings ACM/IEEE MobiCom*, pp. 243–254, August 2000.

[11] K. Gabriel and R. Sokal, “A new statistical approach to geographic variation analysis,” *Systematic Zoology*, vol. 18, pp. 259–278, 1969.

[12] Toussaint G.T, “The relative neighborhood graph of a finite planar set,” *Pattern Recognition*, vol. 12, pp. 261–268, 1980.

[13] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, “GHT: A geographic hash table for data-centric storage in sensornets,” *In Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA)*, September 2002.

[14] S. Woo and S. Singh, “Scalable Routing Protocol for Ad Hoc Networks,” *Wireless Networks*, vol. 7, pp. 513–529, January 2001.

[15] I. Stojmenovic, “Home agent based location update and destination search schemes in ad hoc wireless networks,” *Advances in Information Science and Soft Computing, WSEAS Press*, pp. 6–11, 2002.

[16] J. Li, J. Janotti, D. S. J. D. Couto, D. Karger, and R. Morris, “A Scalable Location Service for Geographic Ad Hoc Routing,” *Proceedings ACM/IEEE MobiCom*, pp. 120–130, August 2000.

[17] C. Cheng, H. Lemberg, S. Philip, E. van den Berg, and T. Zhang, “SLALoM: A Scalable Location Management Scheme for Large Mobile Ad-hoc Networks,” *Proceedings of IEEE Wireless Communications and Networking Conference*, March 2002.

[18] S. Philip, J. Ghosh, and C. Qiao, “Performance analysis of a multilevel hierarchical location management protocol for ad hoc networks,” *Elsevier Computer Communications*, vol. 8, no. 10, pp. 1110–1122, 2005.

[19] Y. Xue, B. Li, and K. Nahrstedt, “A scalable location management scheme in mobile ad-hoc networks,” *Proc. of the IEEE Conference on Local Computer Networks*, 2001.

[20] S. Philip, J. Ghosh, S. Khedekar, and C. Qiao, “Scalability analysis of location management protocols for mobile ad hoc networks,” *Proceedings of the IEEE WCNC*, 2004.

[21] R. Jain, A. Puri, and R. Sengupta, “Geographical routing using partial information for wireless ad hoc networks,” *IEEE Personal Communications*, vol. 8, pp. 48–57, February 2001.

[22] Sumesh J. Philip, “Scalable Location Management for Geographic Routing in Mobile Ad hoc Networks,” *Computer Science Tech Report TR-2005-21, SUNY at Buffalo*, 2005.

[23] X. Zeng, R. Bagrodia, and M. Gerla, “Glomosim: a library for parallel simulation of large-scale wireless networks,” *Proceedings of the 12th Workshop on Parallel and Distributed Simulations – PADS ’98*, May 1998.

[24] B. Karp, “Greedy perimeter stateless routing (GPSR),” *http://www.icir.org/bkarp/gpsr/gpsr.html*.