Int. Computer Symposium, Dec. 15-17, 2004, Taipei, Taiwan.

A Reliable Location Service for Mobile Ad Hoc Networks

Chih-Hsun Chou Kuo-Feng Ssu

Department of Electrical Engineering
National Cheng Kung University
Tainan, Taiwan 701

Abstract*-Location service support is required for mobile nodes using geographic ad hoc routing. The mobile nodes need location information of destinations for sending packets. Due to mobility or failure of the nodes, the location queries may not succeed. This paper presents a reliable location service (ReLS) mechanism that employs dual servers to maintain location information of mobile nodes. With the mechanism, the number of failed queries, including no response and erroneous results, can be reduced. ReLS has been implemented and evaluated with the ns2 simulator. The experimental results show that ReLS improves service availability and location query latency compared to the Grid Location Service (GLS).*

Keywords*: Mobile ad hoc networks, location service, geographic routing, fault tolerance.*

1 Introduction

Geographic routing supports a fully any-to-any communication pattern without explicit route establishment. The basic assumption of geographic routing is the availability of a positioning device such as a GPS receiver at each mobile host [1, 2]. Mobile hosts can use the geographic location information to routes packets towards the destination. Therefore, geographic routing is appropriate for large scale networks.

Since the geographic routing is based on the coordinates, not the address, a mobile node cannot communicate with the destination node without knowing where the target is. Geographic routing has to be augmented with the location service that can provide each node's up-to-date location. The Grid Location Service (GLS), a well-known scheme, provides a scalable and elegant solution but did not address the issue of node failure [3].

In order to tolerate the node failure, a reliable location service (ReLS) is developed for the geographic ad hoc routing. ReLS utilizes a dual-server approach to managing location information. If one location server fails, the other will take over the service. Additionally, ReLS improves the location update and query performance.

A geographic forwarding scheme based on Cartesian routing is used for performance measurement [4]. Several varying scenarios were used for evaluation. Simulation results show that the geographic routing with ReLS provided the higher successful query rate than that with GLS. Furthermore, ReLS reduced the delay during location query.

2 Related Work

As described in the introduction, geographic routing consists of the geographic forwarding and the location service that maps the identifer of a mobile node to its position. Existing geographic forwarding algorithms can be mainly classified into two manners, beacon-based and contention-based. With beacon-based routing, a node must periodically broadcast beacon messages to exchange location information with its neighbors [4]. Data packets thus can be forwarded closer to the target. For instance, GPSR is a famous beacon-based forwarding mechanism [5]. DREAM requires that each node periodically flood its location information over the whole network so the complete location database can be built [6]. Although the beacon-based schemes achieve rapid forwarding, the route maintenance may consume large energy.

With contention-based schemes, mobile nodes do not maintain route information. Instead, a contention process is performed to select the next-hop. The contention process uses suppressing strategies to avoid the collision and to ensure that only one node is elected as the next relay. GeRaF randomly chooses a relaying node using a region-based priority suppression [7, 8]. Three suppression mechanisms were provided by CBF according to forwarding efficiency and suppression characteristics [9, 10]. The contention-based mechanisms typically involve larger transmission delay due to the necessary contention process for each relaying. LAR is

---

This research was supported by the Taiwan National Science Council (NSC) under contracts NSC 92-2213-E-006-042 and 93-2213-E-006-019.

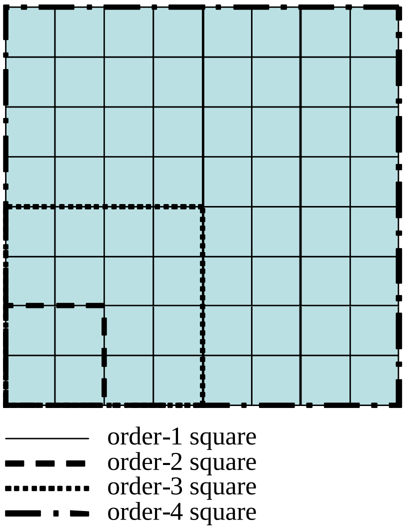

Figure 1. The hierarchical grid structure of ReLS.

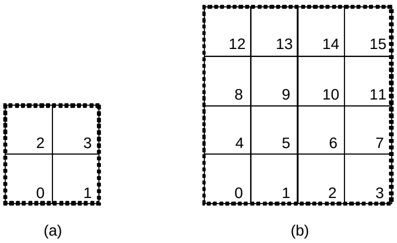

Figure 2. The square ID definition in an order-2 and order-3 square.

another type of forwarding algorithm that requires no beacon [11]. A route request in LAR is flooded in the direction to the target node. A node responses to the route request depends on whether it is in the region that leads to the destination. ReLS can be attached to either a proactive or a reactive scheme.

Several researches have been proposed for providing a location service for geographic ad hoc routing. They can also be divided into reactive and proactive schemes. In a reactive manner, location information is acquired using flooding when a node want to communicate with another node. With a proactive method, location information is maintained by periodically location updating. RLS provides a location service reactively that a RLS node must flood a location request and wait the answer from the destination node before communication [12]. The drawback of reactive scheme is that the flooding operation may cause heavy traffic loads to the network. Additionally, an unbounded location discover latency can be resulted especially in a large scale network. Examples for proactive location services are DREAM, Homezone, and GLS. DREAM nodes periodically exchange location information with other nodes in the network so the complete location database can be created. Homezone adopts the concept of a virtual Homezone where position information for a node is stored [13]. A hash function is used for determining the Homezone of a node, where a node's ID can be hashed to a center of Homezone with a constant radius R. GLS provides a scalable and elegant method for equally distributing location database among all nodes in the network. Our approach is similar in spirit, though ReLS employs dual servers structure for providing higher service availability. In addition, ReLS provides the new location query and update scheme for a mobile node to rapidly locate another node.

3 Geographic Forwarding

Each node is assumed to have a unique permanent identifier. To obtain the geographic location information, the GPS receiver is attached to each node. Each node maintains a routing table for 2-hop neighbors' location information. The table contains a neighbor's ID, location, speed, and timestamp. A node periodically broadcasts beacon messages that includes a list of the location information of itself and all the neighbors so the neighboring information can be updated according to the beacons. Once a node wants to communicate with another node, it attaches the geographic coordinates of the intended target to the packet header. Then the node consults its routing table to choose the node closest to the position of the destination and forwards the packet to the node. The packet will be continuously forwarded to the next hop until it reaches the destination or the TTL of the packet expired.

A well-known problem in geographic forwarding is *dead-end*. When a packet is not able to find a node for the next hop, nor has reached the destination, the process reaches a *dead-end*. Some schemes such as GPSR have been proposed for dealing with this problem [5]. Our approach employs the solution of GPSR. When a dead-end occurs, the current relay must follow the right hand rule to perform an expanding ring search until a closer node is found.

4 Reliable Location Service

4.1 System Models and Assumptions

ReLS adopts the hierarchical grid structure based on GLS. Where the surface of the earth is arbitrarily partitioned into a hierarchy of grids with squares of increasing size, as shown in Figure 1. The smallest square represents an order-1 square. An order-2 square consists of four order-1 squares, and so on. An order-n square can participate only one order-(n+1) square. Furthermore, each order-1 square is

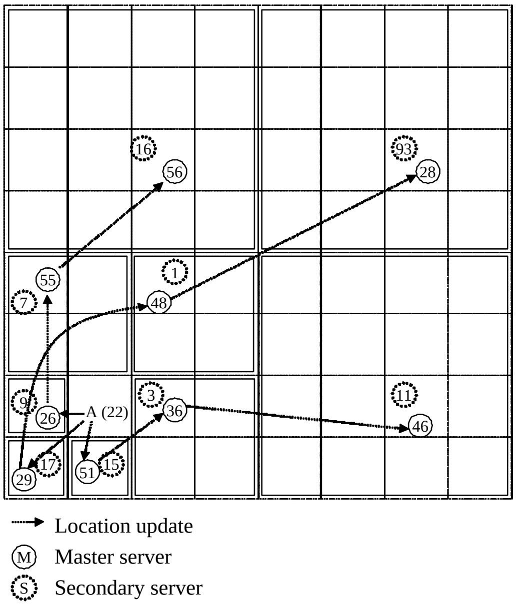

Figure 3. An example of location server selection.

given a square ID for each order of squares above level-1. Figure 2(a) and 2(b) respectively shows the definition of a square's order-2 and order-3 ID. An order-n square ID can also be known according to the same rule.

In our approach, each node is equipped with a GPS receiver. With the position information obtained from the GPS receiver, a node can know which square it resides in. Each node equips the 2Mbps IEEE 802.11 radio where the transmission range (γ) is 250 meters. The length of the edge of order-1 square (ε) is set to 100. The value γ is larger than $\sqrt{2} \cdot \varepsilon$ so each node can directly communicate with other nodes in an order-1 square. Actually, the definition of such a structure can be instead of any other hierarchical partition of the space. GRID provides an analysis on selecting appropriate grid size for grid structure [14].

4.2 Location Server Selection and Update

In ReLS, a node's location information is maintained by some location servers distributed throughout the network. These location servers are automatically elected by the location update operation. There are two cases that a node must update its location information: (1) when a node joins the network, (2) once a node moves a particular threshold distance ω since the last update. The update operation includes the following three steps. First, a node A sends location update messages to its neighbor order-1 squares in the same order-2 square. Secondly, the node received the update message compares the ID of A with itself. The node with the least ID greater than A in the order-1 square becomes the master

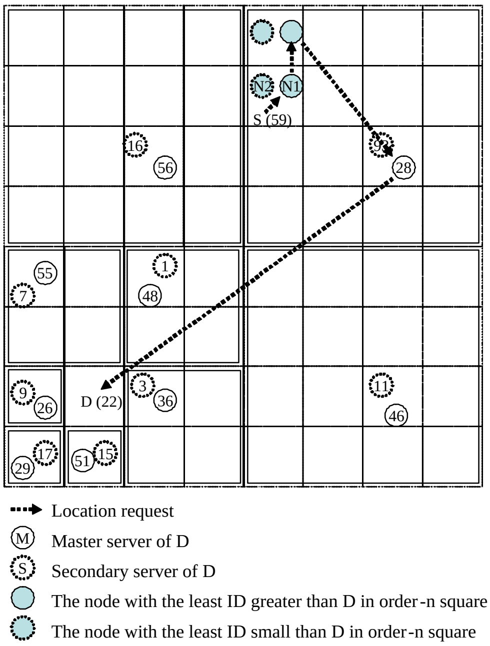

Figure 4. An example of location query.

order-1 location server of A, and the node with the least ID smaller than A in the order-1 square turns into the secondary order-1 location server.[1] Thirdly, the selected order-1 master location servers forward the update messages to the order-1 squares with the order-2 square ID equals to the remainder of A's ID divided by $(2^2)^{2-1}$ of other order-2 squares in the same order-3 squares (called order-2 location square of A). Similarly, the elected order-n master location servers forward the message to the order-(n+1) master location servers.

Figure 3 shows an example of the details of the server selection. A node chooses three master servers and three secondary servers for each order of the grid hierarchy. In each of the three order-1 squares, node A selects the nodes closest to itself in ID space as its master/secondary order-1 servers. In this case, thus, node A chooses 26, 29, 51 as its master order-1 location servers, and selects 9, 17, 15 as its secondary order-1 location servers. Then 26, 29, 51 takes the responsibility on selecting the order-2 servers in order-2 squares. Thus, 26, 29, 51 respectively selects node 55, 48, 36 as the master order-2 servers and 7, 1, 3 as the secondary order-2 servers. The same election process occurs in higher order squares.

---

[1] The ID space is considered to be circular, for example, 13 is less greater than 25 compared with 19 and 25 is less smaller than 13 in contrast to 19.

4.3 Location Query

Once a node wants to communicate with another node, it has to perform a location query for obtaining the location information of the destination node. Figure 4 shows how a node S (59) finds the location of the destination node D (22). To perform a location query, S sends a request to the node (*N1*) with the least ID greater than D in its order-1 square. At the same time, the node (*N2*) with the least ID smaller than D also can receive the request by overhearing. If *N1* or *N2* contains the location information about D, that means *N1* or *N2* is a location server of D. In the example, no information about D is found neither in *N1* nor in *N2*. Thus *N1* forwards the request to the node with the least ID greater than D in the order-1 square with the order-2 square ID equals to the remainder of D divided by $(2^2)^{2-1}$ of its order-2 square. Similarly, the querying process is continuously executed in higher order squares until the request reaches the location server of D or the TTL of the request expired. When the request reaches the location server of D, the server forwards the request to D. After receiving the request, D replies the position of itself to S. Thus S can directly communicate with D by geographically routing packets to the position. On the other hand, if S gets no reply after the TTL expires. S reinitiates the query operation periodically using binary exponential back-off to increase the TTL.

5 Discussions

5.1 Caching

During forwarding, the location information of source node and even destination node are attached to each data packets. Thus a node can acquire location information without any extra cost from these passing-by packets and store these information in a location cache. In ReLS, each node maintains a local location cache with α entries.[2] When the cache is full, the oldest entry should be replaced by the new one. Once a location request is received, a node also checks its local location cache. If the desired location information of the request is recorded in the cache, the node directly replies the information to the requester. This scheme helps to reduce the latency of location request.

Table 1. Parameters for Simulations

simulator	ns-2.1b9a
simulation time (sec)	300
radio transmission range (m)	250
simulation area (m^2)	1000×1000
mean number of nodes	200, 250, 300, 350, 400
location update threshold (ω)	100, 200
mobility model	random waypoint
max moving speed (m/s)	10, 30, 50
pause time (sec)	30

5.2 Empty Square

ReLS works well in dense networks. However, when the number of nodes is insufficient to fill out all squares, empty squares will make a strike against to the location update and query operation of ReLS. To prevent the empty square problem, ReLS provides exception handling for update and query operation. The details are described in the following.

During updating, if an intermediate node finds no node resides in sender's location square, it is called that the location updating meets an empty square. In this situation, the intermediate node temporary turns into a master location server of sender and forwards the update message to select the higher order location server. When a node joins the empty square. the temporary server will pass the location information belong to the square to the new node.

When an intermediate node cannot find any node in the location square of sender during location querying, the node must request all its 2-hop neighbors to check if any node keeps the location information for the target node. If at least one node responds, the intermediate node forwards the location request to one of the responded nodes. Otherwise, the intermediate node just forwards the request to the higher order location square of destination node.

6 Experimental Results

6.1 Simulation Setup

ReLS and GLS are implemented for comparison using the ns2 simulator (ver 2.1b9a) [15]. The simulation area is 1000 * 1000 square meters. The nodes are placed at uniformly random locations in the area. Each simulation runs for 300 seconds of simulation time. The simulation involves only location updating and querying without any data traffic. Each node initiates 20 location queries to random destination node starting at 30 seconds and the query interval is based on poisson distribution with mean λ = 10. Two values are selected for the location update threshold distance ω: 100 and 200 meters.

---

[2] There is a tradeoff on selecting an appropriate value for α. A large α brings heavy storage cost mobile nodes, but reduces the query latency. Contrariwise, a little α causes light overhead on storage but increases the query latency. In the simulation, the α is arbitrarily set to 10.

Table 2. Average Location Query Failures

Query Failures Caused by Reason (2)			
	GLS	ReLS	
max. moving speed	# of query failures	# of query failures	decreasing ratio
10 (m/s)	155	128	17.42%
30 (m/s)	168	140	16.67%
50 (m/s)	153	136	11.11%

Query Failures Caused by Reason (3)			
	GLS	ReLS	
max. moving speed	# of query failures	# of query failures	decreasing ratio
10 (m/s)	268	124	53.73%
30 (m/s)	272	148	45.59%
50 (m/s)	278	121	56.47%

The random waypoint model is used for node movement. When a node arrives the destination, it will stop moving for a predefined pause time and then move to another position. Each run randomly selects 20% nodes abrupt departing the network for 30 seconds. Each data point presented is an average of three simulation runs. The simulation parameters are summarized in Table 1.

6.2 ReLS Results

The reasons for location query failures in GLS can be classified into three main classes: (1) geographic forwarding fails since at some intermediate node there is no choice for the next-hop node, (2) the TTL of packets expired, and (3) no location server can be found or the founded location server does not contains the location information of the intended node. ReLS provides dual servers structure for preventing the occurrence of Reason (3). The definition of location square ID in ReLS helps a node efficiently forwards the request to the correct position, so the failures caused by Reason (2) can be decreased. In addition, the caching mechanism prevents both the appearances of Reason (2) and (3). The location query failure reducing rate of ReLS compared to GLS was analyzed in Table 2. 300 node participated in each simulation, and update threshold (ω) was set to 100 meters. Thus the the total number of queries was about 6000. The results showed that 15% Reason (2) failures and 52% Reason (3) failures can be saved. Since 62% of query failures are caused by Reason (3) and 29% are induced by Reason (2), ReLS reduces about 37% of total query failures.

Figure 5 shows the success rate for ReLS and GLS location queries with the maximum moving speed is 10 m/s. Because number of query failures are reduces, ReLS provides a higher service availability than GLS. As mentioned before, ReLS works well in a dense network. Though ReLS provides a lower success rate in the loose network, it still outperforms GLS. The location update threshold distance ω is also an important factor for service availability. When ω is 100, a node updates its location server much often so the query success rate is raised. However, more update packets will be generated in this case (see Figure 6). ReLS brings a unobvious increasing of traffic load when the network size grows.

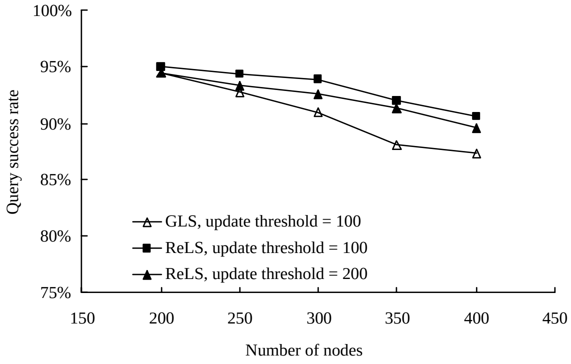

Figure 5. The average successful query rate.

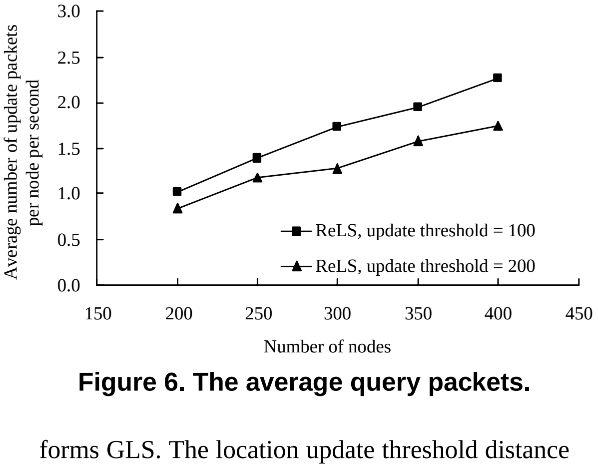

Figure 6. The average query packets.

Table 3 shows how the average distance that a location query travels in hops between the source and the destination. The simulation included 300 nodes with the update threshold (ω) of 100 meters. Loca-

Table 3. Average Length of Query Path

	GLS	ReLS
max. speed	query path length	query path length
10 (m/s)	9.31	8.14
30 (m/s)	9.56	7.89
50 (m/s)	9.39	8.02

tion query in GLS is continuously forwarded to the next node with the least ID greater than the destination node, so the length of query path cannot be expected. ReLS uses a simple hash function to map the node's ID to the specified location squares so the query path length can be bounded in a limited area. With the mechanism, the location query can be rapidly sent to the destination node. The caching mechanism also reduces the delay of location query. Thus ReLS provided a shorter location query latency than GLS.

7 Summary

This paper presents a reliable location service for geographic ad hoc routing. ReLS employs the dual-server structure for tolerating the location server failure and also provides a caching mechanism for fast query process. The performance for location update and query was also improved. ReLS was successfully implemented with the ns2. The simulation results show that ReLS prevented about 36% of location query failures compared to GLS.

References

[1] G. Dommety and R. Jain, "Potential Networking Applications of Global Positioning Systems(GPS)," Tech. Rep. TR-24, Department of Computer Science, The Ohio State University, Apr. 1996.

[2] E. D. Kaplan, *Understanding GPS: Principles and Applications*. Artech House, Boston, 1996.

[3] J. Li, J. Jannotti, D. S. J. De Couto, D. R. Karger, and R. Morris, "A Scalable Location Service for Geographic Ad Hoc Routing," *Proc. ACM/IEEE MobiCom*, pp. 120–130, Aug. 2000.

[4] G. G. Finn, "Routing and Addressing Problems in Large Metropolitan-Scale Internetworks," Tech. Rep. ISI/RR-87-180, Information Sciences Institute, University of Southern California, Mar. 1987.

[5] B. Karp and H. T. Kung, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," *Proc. ACM/IEEE MobiCom*, pp. 120–130, Aug. 2000.

[6] S. Basagni, I. Chlamtac, V. R. Syrotiuk, and B. A. Woodward, "A Distance Routing Effect Algorithm for Mobility (DREAM)," *Proc. ACM/IEEE MobiCom*, pp. 76–84, Oct. 1998.

[7] M. Zorzi and R. R. Rao, "Geographic Random Forwarding (GeRaF) for Ad Hoc and Sensor Networks: Multihop Performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 337–347, Oct.-Dec. 2003.

[8] M. Zorzi and R. R. Rao, "Geographic Random Forwarding (GeRaF) for Ad Hoc and Sensor Networks: Energy and Latency Performance," *IEEE Transactions on Mobile Computing*, vol. 2, no. 4, pp. 349–365, Oct.-Dec. 2003.

[9] H. Füßler, J. Widmer, M. Käsemann, M. Mauve, and H. Hartenstein, "Contention-Based Forwarding for Mobile Ad-Hoc Networks," *Ad Hoc Networks Journal (Elsevier)*, vol. 1, no. 4, pp. 351–369, Nov. 2003.

[10] H. Füßler, J. Widmer, M. Mauve, and H. Hartenstein, "A Novel Forwarding Paradigm for Position-Based Routing (with Implicit Addressing)," *Proc. of IEEE 18th Annual Workshop on Computer Communications*, pp. 194–200, Oct. 2003.

[11] Y. B. Ko and N. H. Vaidya, "Location-Aided Routing (LAR) in Mobile Ad Hoc Networks," *Proc. ACM/IEEE MOBICOM*, pp. 66–75, Oct. 1998.

[12] M. Käsemann, H. Füßler, H. Hartenstein, and M. Mauve, "A Reactive Location Service for Mobile Ad Hoc Networks," Tech. Rep. TR-02-014, Department of Computer Science, University of Mannheim, Nov. 2002.

[13] S. Giordano and M. Hamdi, "Mobility Management: The Virtual Home Region," Tech. Rep. TR-SSC/1999/037, EPFL, Lausanne, Switzerland, Oct. 1999.

[14] W. H. Liao, Y. C. Tseng, and J. P. Sheu, "GRID: A Fully Location-Aware Routing Protocol for Mobile Ad Hoc Networks," *Telecommunication Systems*, vol. 18, no. 1-3, pp. 37–60, Sept. 2001.

[15] K. Fall and K. Varadhan (editors), *The ns Manual (ns Notes and Documentation)*. The VINT project, www.isi.edu/nsnam/ns/ns-documentation.html, Feb. 2002.