International Journal of Smart Home
Vol.8, No.3 (2014), pp.33-48
http://dx.doi.org/10.14257/ijsh.2014.8.3.04

A Group mobility oriented History information based Light Location Service

Xinyu Yang, Linqi Li and Xiaojing Fan

Department of Computer Science and Technology
Xian Jiaotong University, P.R. China
yxyphd@mail.xjtu.edu.cn,lilinqi@gmail.com, fanxiaojing@cn.fujitsu.com

Abstract

In a large number of location service protocols, the load for long-distance periodic updates takes up a big portion in the total load. So it's possible to effectively reduce the overall load of a protocol through the drop of update load as to further enhance the scalability of the protocol. In this paper, with the use of mobility features of groups on network nodes, it proposes to utilize the location service framework which is based on historical information together with clustering mechanism to curtail the information amount saved by single nodes in the local location database, optimizing individual node's memory overhead and the initialized load of location service protocol. Both higher service success ratio and lower load are enabled in the massive mobile Ad Hoc network.

Keywords: *Group Mobility, Location Service, Clusters, MANET*

1. Introduction

In a large number of location service protocols, the load for long-distance periodic updates takes up a big portion in the total load. So it's possible to effectively reduce the overall load of a protocol through the drop of update load as to further enhance the scalability of the protocol. The complicated mechanism for location update and request is vulnerable to the mobility of network nodes, in particular, in the large-scale network, it's rather difficult to maintain the consistency of information with complex hierarchies and grid structures. When the service success rate is decreased, it's more likely to increase the maintenance overhead. Mobility characteristics of groups are widely found in MANET application circumstances like post-disaster rescue and battlefield communication. Current location service protocols are designed with less focus on the dynamism of groups and inaccessibility of the guiding role by mobile group information in the location service, restricting the availability of location service protocols in massive situations like that.

At present, a variety of location service protocols generally employ flooding or local flooding to achieve the cyclic update of location information. So in the network whose nodes have strong mobility, in order to fully ensure the accuracy of information stored in the server (hereafter location server referred to server), the traditional location service inevitably utilizes extensive and periodic flooding or local flooding, resulting in plenty of control loads. The control loads of location service limit too much the success and accuracy ratio. Moreover, inaccurate information regarding the location will bring down the packet delivery rate of position-based routing. Packet conflict and collision caused by excessive load will weaken the performance of location routing. Group mobility are intensively seen in the application scenarios of massive mobile Ad Hoc

ISSN: 1975-4094 IJSH

networks [8], such as Disaster relief and battlefield communications. In such scenes, nodes need to collaborate in groups. Mobile nodes are organized in groups and nodes in the same group have to coordinate mutually to maintain the locations adjacent to them according to the similar movement mode. Nodes in the same group move as a whole and that the location changes in a similar trajectory.

With the use of group mobility features of network nodes, it proposes to utilize the location service framework which is based on historical information together with clustering mechanism to rebate the information amount saved by single nodes in the local location database, optimizing individual node's memory overheads and the initialization load of location service protocols. Both higher service success ratio and lower load are enabled in the large-scale group movement network.

2. Network Model

Before detailed description of algorithm mechanism of the protocol, it's necessary to introduce the MANET model based on by the protocol. G-HLLS protocol adopts MANET featuring the same node and single channel. The model has following properties:

(1) The network area is two-dimensional surface, *i.e.*, all network nodes move randomly in a given two-dimensional plane area;

(2) Each node has the only identifier (ID);

(3) Each node is assigned with a wireless transceiver device, of which the transmission radius is R and communication parameters like bandwidth is completely same;

(4) The area for direct communication by nodes is considered as a circular plane region, with node as the center and radius R, in other words, if the Euclidean distance between any two nodes is shorter than R, they can directly communicate without the aid of any intermediate routing node;

(5) Two nodes which can communicate directly are considered neighbor nodes; the network topology is a set of vertices by all mobile nodes; links between the neighbors form the connected graph of edge sets;

(6) The number of a node's neighbors is the degree of the node;

(7) Each node can acquire its accurate location coordinates through global positioning system (GPS).

In the design of G-HLLS protocol, network nodes are divided into cluster heads and members. Therefore, the location information includes two types. The position information regarding a node as head has the four elements: coordinates, speed, timestamp, as well as member table, which can be expressed like:

$HPMj(i) < Lji,Vji,Tji,Mji >$

HPMj(i) is the tetrads of location information about a head node i saved by any node j in the network, where Lji is the coordinate of i when j records the information; Vji is the movement speed of i; Tji is the time stamp when j takes the record; Mji refers to member table of head i, which has IDs of all member nodes in the cluster.

The location information regarding a member node in the cluster includes location coordinates and time stamp, of which the expression is:

MPMj(i) < Lji,Tji >

MPMj(i) stands for the location information of a member node i reserved by any node j in the network, where Lji is the location coordinate of the node i when j makes the record; Tji means the accurate time for recording the information.

Definition of the Group Model

For the group movement in G-HLLS protocol, all network nodes are classified into several groups, in each of which a head node exists. Nodes in the same group move together. Head nodes determine the direction and velocity of movement. Members in the group move at the same speed and towards the same direction as the head. That's why nodes in the same group locate with a similar track. It can be defined as:

(1) Head nodes: To execute Random Waypoint model, it's required, starting from the original location, to select randomly a target position in the network area and a motion rate within [1m/s, Vmaxm/s]; then nodes move towards the target position at the selected speed. After the target position is accessed, it's necessity to wait till all members in the group arrive at the intended location before the choice of next target is made.

(2) Common nodes: Starting from the initial place to randomly choose a point as the target location in the circular region where the target position of head is center and radius λR, with the speed of the head as movement rate, shifting towards the selected location. When they go close to the target, the next target won't be chosen before all nodes in the same group get to the preset target. Moreover, it's necessary to update periodically the cluster head of common nodes.

3. G-HLLS Protocol

3.1. Overview of the Idea

G-HLLS protocol updates the location with the dependence on HELLO packet. As well, the core of location request mechanism tracks destination nodes in the way as what's based on the historical trace information. But due to few amount of information maintained by each node, the mechanism for location update and request needs improvement based on HLLS algorithm [1]. Figure 1 depicts the main idea of G-HLLS protocol. The protocol adopts integrative the location service and clustering mechanism oriented towards group movement, but the research concern is not clustering and maintenance techniques. Therefore, G-HLLS protocol applies the simpler clustering and maintenance mechanisms put another way, nodes in the same movement group form a cluster, of which the head node is considered as the one of the cluster, responsible for collecting and maintaining location information about its members. The cluster head can be regarded as agent of all its members, *i.e.*, nodes outside the cluster look on merely the whole cluster where the head is taken as agent, no care for everything inside the cluster. The database for the local location of each node stores only the information regarding the location of each head and the update for that depends mainly on HELLO packet. Hence, it's required to build across the whole network the distributed database for the location information of cluster heads and such information has gradient time stamp, as seen in Figure 1 (gray nodes). The darker the gray becomes, the more recently nodes update the information. Thus, for the location request for nodes of the cluster head, it can be tracked by direct dependence on historical location information in the distributed database mentioned above. Yet, with regards to the location request for a member node in the cluster, it's converted to the track of the head's nodes of the cluster where it belongs to. Then the request packet is delivered by the cluster node to the member one. But the member may change the cluster. So in the process of location request for a member node, it's advisable to verify at times whether the node has left the original cluster and joined in a new one, which is called dual-pointer track. Figure 1 shows the location request path of a node Dn, which omits those nodes not involved with the request. Dn which originally belongs to the cluster where the head is C1 joins in the one with C2 as head. The source node Sn sends a request for the location of Dn. According to the local historical information that Dn belongs to the cluster where C1 belongs, the request should be offered to the head C1. Then, the request packet is transmitted with the use of target track strategy based on historical information. A node Pn met the head C2 in the location L0 and learnt that Dn joined the cluster which C2 belongs to. When Pn passes the packet, it won't send it to C1 but to the historical location L0 of C2. The node Pn like that is renamed as cluster-pointer node, which signals the change information about the head of destination nodes. For the stage from Pn to C2, the location request packet is forwarded still with the destination track strategy which bases on the historical information. The node which involves in the transmission of the request package and can indicate the coordinates of the destination cluster head's new location is appalled cluster-head pointer node. Following the instruction of that node, the location request packet is delivered to the head C2 and then to a destination node Dn by C2. Dn returns the current location information through greedy routing to the source node Sn. The location request and response is finally completed.

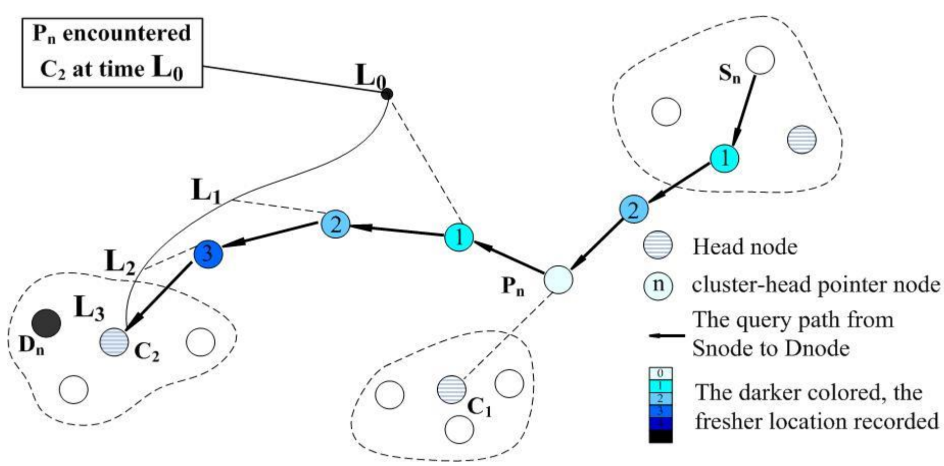

Figure 1. Overview of G-HLLS

G-HLLS protocol includes three stages: Firstly creating LLDB on each node by carrying out the initialization mechanism, then updating the regular location of LLDB; when source nodes require for the destination location information, the location request and response mechanism is triggered.

3.2. Protocol Initialization

G-HLLS protocol initialization mechanism aims to ensure the establishment of local location database (LLDB) for all nodes of the network, which has the information with regards to all nodes' ID and location of the cluster head. The location information has four elements: location coordinates, speed, timestamp and member table. So nodes of every cluster head should spread information of its own across the whole network. To achieve that, G-HLLS adopts multicast which bases on the idea in infectious diseases algorithm for low load and reliability. But the location information of member nodes does not get involved with the reliable multicast.

During the initialization of G-HLLS protocol, each node, whether of cluster head or member, collects the location information about the head by means of the HELLO packet disseminated in the direct neighboring area, put another way, if HELLO packet sent by head's nodes is received, the information contained is inserted into LLDB. When the number of new entries in LLDB exceeds Ngossip, the new added ID of cluster head should form a table as Gossip information to be bundled in the HELLO packet for propagation. The special HELLO packet is called HELLO-GP. When the adjacent node receives HELLO-GP, it examines if there is cluster head's ID which is not saved in local LLDB. If yes, it will use unicast to deliver MREQ packet to the source node HELLO-GP, asking for the missing information about the cluster head. Once HELLO-GP receives MREQ, it will use unicast to return MREP, returning the required location information about the cluster head.

G-HLLS initialization algorithm is a type of distributed technique. Each node across the entire network performs similar algorithmic steps. With any node as example, we'll introduce the process of G-HLLS initialization:

Step1. Check periodically if the new records of node i in the LLDB reach Ngossip. If not, go to Step2;

Step2. If it arrives in the next HELLO cycle, HELLO packet will be produced, which contains the information about its ID and the current location and is sent out in the way of one-hop broadcast;

Step3. Use ID of cluster head's nodes in the new records to form abstract table; set the counter of new records 0 and generate HELLO-GP packet, included its own ID, current location information and abstract table; when the next HELLO packet arrives, they are sent out through one-hop broadcast;

Step4. When HELLO packet is received, if its source node is the one of cluster members', then stop; if the source node of HELLO packet is the one of cluster head's, and the head's ID and the location tetrad are new records, then add them to LLDB and change the figure of the counter to 1; if they are not new, use them to update LLDB;

Step5. When HELLO-GP is received, examination is made over the abstract table; if there are entries about the cluster head excluded in LLDB in the abstract table, MREQ which includes the abstract table of the missing cluster head is generated, and sent in a unicast way to the source node of HELLO-GP;

Step6. When MREQ is obtained, check the abstract table of the missing cluster head; find from the local LLDB all requested cluster head's IDs and the location tetrads to produce MREP packet, including the obtained information about the cluster head; return MREP in a unicast way to the source node of MREQ packet;

Step7. When MREP is received, input all cluster head's IDs and location tetrads which are included in the MREP packet to the local LLDB and update the accordingly entries on the timer.

During the initialization of G-HLLS protocol, information for spread refers only to that regarding the location of cluster head. Accordingly, the number of HELLO-GP, MREQ and MREP control packets which are in need decrease significantly, thus to the remarkable reduction of the loads for the initialization of G-HLLS protocol. If LLDB of each node has complete information about the cluster head, then G-HLLS protocol initialization ends and nodes stop transmitting HELLO-GP, MREQ and MREP. For the daily maintenance of network after initialization, only common HELLO package will be utilized.

3.3. Location Update Mechanism

The location update mechanism for G-HLLS protocol consists of two parts: local location update and global location update. The former is confined inside the cluster, i.e. the cluster head is responsible to maintain the location information of its members. The latter's task is to update LLDB of each node, *i.e.*, to update the location information of the cluster head's nodes throughout the network.

(1) Local location update

As specified by the definition of group movement model, each group member, *i.e.*, node of cluster member, has knew the target location and motion speed of the cluster head when it begins to choose the next moving target; and all nodes move close to the moving target with uniform motion in a straight line. With the formula about uniform linear motion, cluster member's node can get the location coordinates of its cluster head at any time point. In this regard, the task of local location update is simply to ensure real-time awareness of the location information of all its members.

Local location update should occur inside every cluster. Nodes of cluster members regularly send member location update packet (MLUP) in a unicast way to those of the cluster head. MLUP has ID and location information two-tuples like current location coordinates and timestamp of members. Its routing is enabled through greedy forwarding method. When the location update packet is received by nodes of the cluster head, ID and location information included in it will be imported or the local member table updated.

(2) Global location update

The task of this mechanism is to update in real time LLDB of each node in the ensemble network. G-HLLS protocol makes use of the location information saved in HELLO packet, DATA packet and LRP to make piggyback update of LLDB. LRP packet, location reply packet, is returned by the requested destination node through greedy routing in a unicast way to the location request source node. The information included in LQP packet about the source node is the most recent location information of the requested destination node. So in the routing process, it's possible to perform piggyback update of the location information of the node mentioned above. LLDB keeps only the information about cluster head. From that point, only HELLO packet, DATA packet and LRP packet which carry the information about the location of the head can participate in the location update. We'll illustrate in the following passages.

In G-HLLS protocol, any node will perform the global location update under these three circumstances:

● When a node receives HELLO packet and if the source node is that of the cluster head, the head's ID and location information tetrads carried by HELLO packet will be employed to update simultaneously the local LLDB and neighboring table (NT); if the source node is a cluster member, members' ID and location information two-tuples carried by HELLO packet will be utilized to update NT;

● When a node receives DATA packet and if at least source or destination node is that of the cluster head, the head's ID and location information tetrads will be used to update the local LLDB; then, DATA packet is sent out; if both the source and destination nodes are the cluster's members, none is updated and DATA packet is forwarded;

● When a node receives LRP packet and if the source information is the cluster head's node, ID and location information of the cluster's node will be applied to update the local LLDB; then LRP packet is transmitted; if the source information is not the cluster head, there will no update and LRP is transferred.

LLDB's update algorithm is to use the cluster head's ID in HELLO/DATA/LRP packet to search the local LLDB and find out corresponding entries of the ID, comparing the timestamp of cluster head's location information in the packet and the detected one. If the timestamp in the packet is more recent, the location information of the cluster head in the packet is used to replace the detected local records. Otherwise, do nothing.

3.4. Location Request and Response Mechanism

G-HLLS protocol initialization and location update mechanism creates the distributed location database of cluster head's nodes in LLDB across the entire network, in which the location information presents gradient timestamps. For that database, the tracking mechanism based on historical information can be directly used to enable the location request of cluster head's nodes. For the location request of its member nodes, it can be finished by the cluster head as agent and converted to the request of the head. To achieve the goal, it's necessary to ask the source node to send location query packet (LQP), which is delivered to the requested destination node; when such a node receives LQP, location reply packet (LRP) will be sent and the current location information will be returned to the request source node. As well, a pair of the request source and destination nodes is taken for example. Snode is any request source node in the network. Dnode is the requested destination node and also the one of a cluster member. The algorithm for Dnode's location request has the following steps:

Step1. Snode generates LQP packet and puts its own ID and location information into LQP as the information of source node; use Dnode's ID to search LLDB and discover entries about the location of the cluster head to which Dnode belongs; input Dnode's ID, its cluster head's ID and the location tetrads to LQP; leave blank the field of the location information about the destination node; send LQP to the destination head; take advantage of the location of cluster head to reckon the next hop of routing according to the greedy forwarding mechanism;

Step2. When LQP packet is received by any node i, Dnode's ID will be used to search for the local LLDB and review if Dnode updated the cluster head;

If yes, ID and the location information tetrads of Dnode's new cluster head will be entered into LQP; then compute the next hop according to greedy forwarding mechanism and LQP is transferred to the new head;

If not, compare the timestamp for the information of Dnode's cluster head in LLDB and that for the information of the cluster head in LQP;

If the former one is more recent, insert the location information tetrads of Dnode's cluster head to LQP, which is found from LLDB; then evaluate the next hop with the greedy forwarding mechanism and continue to send LQP to the destination cluster head;

If the timestamp in LQP is more recent, use the location tetrads of the cluster head in LQP to update entries in LLDB, and then estimate the next hop by the greedy forwarding mechanism before continuing to dispatch LQP to the destination cluster head;

Repeat Step2 till the node signaled by ID field of the destination cluster head in LQP receives LQP; then go to Step3;

Step3. Once the node signaled by ID field of the destination cluster head in LQP receives LQP, use Dnode's ID to retrieve the local member table and check whether it is the cluster head to which Dnode belongs;

If yes, fill up the location information field of the destination node in LQP with the location information two-tuples of Dnode in the member table; send directly LQP to Dnode through greedy routing;

If not, revert to Step2;

Step4. When Dnode receives LQP packet, LRP packet is produced; with its ID and the current location information two-tuples input to LRP packet as the source information; import Snode's ID and the location information carried by LQP to LRP as the destination information; then utilize the greedy forwarding mechanism to get the next hop; send LRP to Snode;

Step5. When any node i receives LRP, Dnode, as the node of cluster member, does not perform piggyback update but reckon the next hop with the use of the greedy forwarding mechanism; send LRP out and repeat Step5 till LRP is obtained by Snode;

Step6. When Snode gets LRP, it has the location information two-tuples of Dnode brought by LRP; by the moment, the location request and response is finished.

From the above steps in the algorithm of G-HLLS protocol, in the process of the location request of cluster members, LQP is led by double pointers. One pointer indicates the historical location updated by the destination cluster head under the G-HLLS location request mechanism. Besides, that mechanism needs cluster-pointer nodes so that it is indicative of the change of the belonged cluster of the destination node. Under the guide of dual pointers, LQP is finally transmitted to the requested destination node, which itself reacts to the location request.

4. Comparative Analysis of Protocols

From the demonstration of G-HLLS protocol, it's observed that only the location information of the cluster head's node gets involved in the protocol initialization and global location update. Compared with HLLS protocol, the maintenance information of single nodes is declined and nodes' memory overheads are cut down. For the discussion in the following, we assume N nodes in the network.

In HLLS protocol, each node should retain the location information of all other nodes in the network. Hence, in the local database, records about all nodes should be included and the storage overheads is O(N).

In G-HLLS protocol, we set the size of each cluster in the group mobile model a. The location information of all cluster heads' nodes are preserved in the local LLDB, with the number of entries N/a. The cluster member's nodes need to maintain only LLDB, but the cluster's have to maintain both LLDB and member table (MT). MT has the location information of all members in the cluster, with the number of records a. Thus, we have the expression for the average number of entries about the location information to be maintained by every single node in the G-HLLS protocol, like (1).

$$\frac{\frac{N}{a}\left(N-\frac{N}{a}\right)+\left(\frac{N}{a}+a\right)\frac{N}{a}}{N}=\frac{N}{a}+1 \tag{1}$$

The node's average storage overheads in G-HLLS is O(N/a+1).

In G-HLLS protocol, only nodes in the hometown area play the role of a location server. Each node saves the location information records on one or more nodes whose geographic locations take place within their hometown. The number of nodes in the hometown will change along with the status of network, which determines the robustness of location service. As a consequence, under the circumstances that both the regional area and the number of nodes included in the hometown are fixed, the average storage overheads of nodes in G-HLLS protocol is O(c)[11].

In GREASE protocol, the average entries of location information stored by each node are substantially susceptible to the mobility features of network nodes. Each node is responsible for keeping the location information of other nodes which ever came across with them. Each node encounters very few others that the memory overheads become less. For the more intense dynamics of the network as a whole, nodes can meet together fully that the memory costs become greater. In the case of maximum overheads and each node meeting one another and saving mutually the location information, the average memory overheads of nodes in GREASE protocol is O(N). But factually, the greater the overheads become, the higher rate of success the location service achieves.

Double Circle protocol has the same qualities to GREASE in terms of the memory overheads. It's also influenced by nodes' mobility. The range of transmission of each node's location information is determined by its displacement distance. When the distance is greater than R2i, where, R is the transmission radius of a node and i a positive integer, the update of location information should be executed for nodes in the circular region with the current location as center and radius R2i+1. If network's mobility is intense and as time goes by, the location information of each node will eventually be able to distribute across overall nodes. Thus, in the worst case, the average memory overheads of nodes in Double Circle protocol is O(N). The average memory overheads for each location service are outlined in Table 1.

Table 1. Comparison Table Storage Overhead Location Services

Location services	G-HLLS	HLLS	GLS	GHLS	GREASE	Double Circle
The average storage overhead	$O(N/a+1)$	$O(N)$	$O(\log(N))$	$O(c)$	$O(N)$(worst case)	$O(N)$ (worst case)

In that table, it's evident that GLS and GHLS protocols have lower memory costs, with G-HLLS moderate, and HLLS, GREASE and Double Circle have the highest. However, such overheads are little affected by the performance of service protocols. On the contrary to several other protocols, lower load and higher success rate of the location service by HLLS and G-HLLS enable them with better scalability and relatively favorable comprehensive performance.

5. Experimentation and Simulation

Comparative analysis is conducted about the performance of those protocols based on OPNET simulation, to prove that G-HLLS protocol has outstanding extendibility in the large-scale group movement MANET.

5.1. Description of Simulation Parameters and Assessment Parameters

For the sake of fair comparison, we used these parameters. For all simulation experiments, the network protocol in MAC layer adopts IEEE 802.11b DCF (Distributed Coordination Function), with channel bandwidth 1Mb/s and wireless transmission radius 200m for all nodes. During the simulation, each node sends HELLO packet every 2s. UDP packet is employed for data service. 30% nodes are randomly selected from the network to join in data service. Every second, those nodes send a 1024bit-big packet and change arbitrarily the destination nodes every 5s. Nodes involved with data service will send the location request for the destination nodes as required. For the group movement model used in the simulation test, the speed of the cluster head's nodes is randomly selected from [1m/s, 10m/s], and each cluster has on average eight nodes. For every simulation, it lasts 700s, of which the first 50s is for warm-up, i.e. no data service happens in [0, 50s]. For the simulation experiment in the same scene and with the same parameters, random seeds are chosen to repeat the experiment for ten times. So every data point in the following diagram refers to the mean value concluded from the ten random simulation results. Table 2 summarizes all parameters for the simulation. In addition to the setting of those data, in order to investigate comparatively the scalability of the aforesaid protocols running in the network of different scale, where nodes range from 500 to 1600. Also the network density is kept the same, i.e. the average degree of nodes is 7. The length of foursquare network area needs to vary along with the size of network.

Table 2. Simulation Parameters

Simulation parameters	values	Simulation parameters	values
MAC layer protocol	IEEE 802.11b DCF	random purpose replacement frequency	5 s
Wireless channel bandwidth	1 Mb/s	mobility model	group mobility model

Wireless transmission radius	200 m	speed variation interval	[1 m/s, 10 m/s]
HELLO packet broadcast period	2 s	average node degree	7
packet type	UDP	simulation time	700 s
Participation rate	30%	simulation times	10
Data traffic	1024 bit/s		

For the experiment, there are two major parameters as follows:

(1) Average rate of success in the location request: It is ratio of the number of location response packets which are successfully received against that of location request packets which are dispatched. For the justice, the location request is not retransmitted so that the success rate of location service is all of the first time.

(2) Protocol overheads: It means the total number of location service protocol control packets which are transmitted per second by all nodes in the network. In G-HLLS protocol, HELLO, MREQ, MREP, MLUP, LQP and LRP are embraced.

With the advantage of the two parameters, we'll compare the performance of the proposed protocol with GREASE, GHLS and HLLS in different size of network.

5.2. Simulation Results and Analysis

(1) Success rate of location service to change with network scale

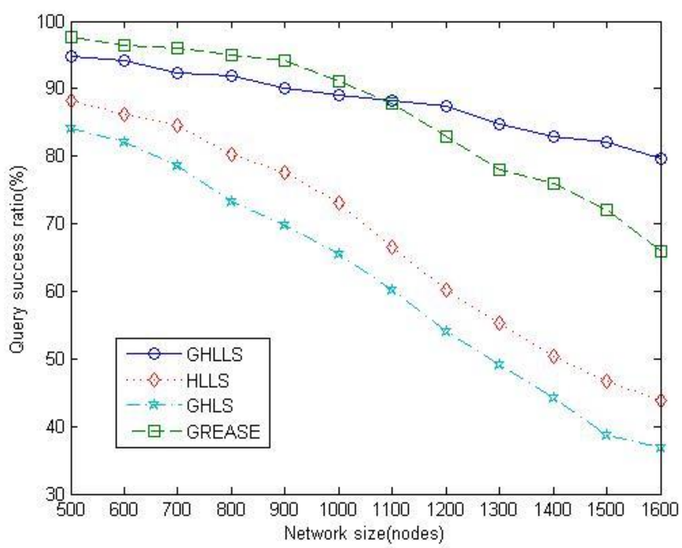

Figure 2. Query Success Ratio vs. Network Size

Figure 2 shows the change of the location service success ratio along with the size of network. As the network size becomes larger, the location service in G-HLLS and HLLS protocols have lower success rate, with the former's rate descending more slowly than the latter's. When the network has less than 1000 nodes, G-HLLS protocol has lower success rate than HLLS, because when the size is not big enough to 1000 nodes, the negative impact by the initialized load on HLLS protocol is not obvious. HLLS protocol can still establish the complete local location database. That's why its success rate keeps still higher. After initialization, besides the global location update with HELLO packet, G-HLLS has to make local update in the cluster, so that its regular load becomes bigger than HLLS'. Moreover, with smaller amount of information in the local location database in G-HLLS than HLLS, the dispersal of the location information is not so complete as in HLLS protocol. All those account for the lower success rate of location service in G-HLLS than HLLS in terms of the network in small sizes. Yet, as the network scales up to possess 1000 nodes, the problem is exposed with too big initialization load in HLLS protocol. Congestion and conflict result in incomplete local location database, i.e. nodes may lose the initial location information of other nodes, and for the reason of wide network, those nodes will not meet for a long time and the location is not updated. Missing too much information of the location will give rise to the degressive success rate of location service. But, since the information to be saved in the local location database in G-HLLS protocol declines remarkably than HLLS, and the initialization load is not too heavy, so the success rate turns down slowly and can be maintained at the acceptable level.

GHLS protocol chooses the location server in a planar hash way. The hometown area where the server locates is the network center. As the network aggrandizes, the load will be becoming out of balance. The load in the hometown area is huge. Overload and conflict will make easily lose the location update/request packet sent to the hometown. Consequently, the success rate of location service drops.

In small-scale network, GREASE has high success rate of such service. But, when the network enlarges, such rate falls rapidly. The main reason lies in the insufficient distribution of the location information. In small-size network, even if nodes move at a moderate velocity, most node pairs can meet as neighbors and update mutually the location information, which can be spread fully to achieve a higher rate of the successful location service. However, in large-scale network, nodes which are distributed geographically far away won't come across nor update the location information. The problem deteriorates with the incomplete scatter of the location information and that the success rate of location service drops quickly.

(2) Overheads of location service protocols to change with network scale

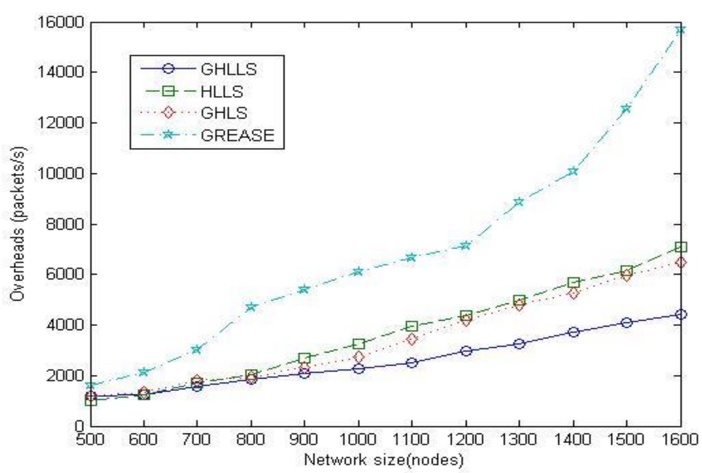

Figure 3. Overhead vs. Network Size

Figure 3 depicts the variation of the load of location service protocols with the scale of network. As the network size becomes bigger, the load of all location service protocols tends to increase. When the size of network is smaller than 600, HLLS protocol's load is smaller than G-HLLS for the two reasons. First of all, the initialization load of HLLS protocol is not greatly big; secondly, G-HLLS should update the local location, which causes the heavier regular maintenance load of G-HLLS than HLLS. But with increasing size of the network, HLLS protocol's load ascends more quickly than G-HLLS. In the large-scale network, G-HLLS protocol loads less, for the reason that the initialization load of HLLS protocol grows rapidly with the expansion of network. During the initializing stage, the quantity of information distributed by G-HLLS protocol is remarkably smaller than HLLS that the load is smaller too. GHLS protocol needs to maintain more daily control packets, such as location update packet and HANDOFF packet. Therefore, it should load more. GREASE protocol introduces regional flooding mechanism. In the large-scale network, so many nodes join in the flooding that the load ascends instantly, reducing the performance of location service.

6. Related Works

Studies started earlier on the location service protocol and have gone further, in two categories like flooding-based and group-based. Early research achievements were mostly based on flooding, where DLS [5] is the typical. DSL was designed specifically for the location service in support of DREAM protocol. Each node called up all nodes in the network as the location server of itself, while the location update mechanism kept to the well-known distance effect so as to limit as much as possible the location update load.

Recent researches are made with focus on the group type, where GLS [6], XYLS [7] and GHLS[4] are representative. From them, many of the current location service protocols have learnt a lot. GLS adopts the principle of Hierarchical hash function to divide network area in grids. According to the given hash function, each node recalls different nodes in different grids as the location server. The same rule is followed to perform the location search. The latest research result MLS can be regarded as the extension of GLS, where nodes across several levels are convoked as the location server. However, the location information saved in each server is no more concrete but a pointer to the server at a lower level.

Based on planar hash function, GHLS employs the idea of home agent. Each node chooses its hometown through hash function mapped onto the plane domain of network. Nodes in the hometown assume as home agent the task of being location servers. GrLS [8] still has its core on the home agent idea by GHLS. The innovation is the consideration of group movement features of network nodes and the combination of location service with group management, which includes group location management and single-point location management. For the group location management, the location server is selected for all group leaders in the defined group home area, in which, the local overloads may appear. The more recent finding Phero-Trail LS [9, 10] has its new points: with the qualities of sensor equipped aquatic swarm, SEA swarm, the location service is fulfilled in the three-dimensional space. Nodes in the 3D space are mapped onto the two-dimensional plane and then are summoned as the location server. The procedure for the location update and tracking still follows client/server mode. Region-based location service management protocol, RLSM [11], adopts message merge for the location update and request to curtail control loads for the enhanced scalability. On the whole, the kernel of group-location-based server is to investigate the enlistment algorithm of the location server. That is also the fundamental difference among those protocols. Excellent enlistment algorithm can ensure higher rates of location service success and accuracy even when nodes recruit fewer location servers.

7. Conclusions

To adapt to the group movement features in the extensive MANET, the paper proposed a lightweight location service protocol oriented towards group motion, G-HLLS. By integrating lightweight location service framework and clustering mechanism, the new protocol considered group mobility and performed the location service protocol in layers. The cluster head's nodes are responsible for the update of local location and confine the location update of plentiful members of the cluster to a small area, which effectively cut down the memory overheads of the local location database. In the location request stage, the cluster head's nodes, as agent of the requested members, carried out the double-pointer location track based on historical information. OPNET simulation experiment was conducted for the comparative assessment of performance of G-HLLS and three other location service protocols. G-HLLS protocol was validated to enable a higher rate of successful location service with lower memory costs and control loads. It demonstrated excellent performance in the large-scale group mobile network.

Acknowledgment

The work was supported in part by the following funding agencies in China: the New Century Excellent Talents in Universities of China under grant NCET-09-0649, the Ph.D. Program Foundation of Ministry of Education of China under grant 20100201110027. The Program for Industrial Development of Shaanxi under grant 2011K06-40. Any opinions, findings and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the funding agencies.

References

[1] X. Yang and X. Fan, "HLLS: A History information based Light Location Service for MANETs", Elsevier Computer Networks, vol. 56, **(2012)**, pp. 731-744.
[2] B. Karp and H. T. Kung, "GPSR: greedy perimeter stateless routing for wireless networks", Proceedings of the 6th Annual ACM International Conference on Mobile Computing and Networking (MOBICOM), Boston, USA, **(2000)**.
[3] E. Lee, F. Yu, S. Park, J. Jung, S.-H. Kim and S. Noh, "General sink location service based on circle and line paths in wireless sensor networks", Proceedings of the IEEE Wireless Communications and Networking Conference (WCNC), Sydney, Australia, **(2010)**.
[4] S. M. Das, H. Pucha and Y. C. Hu, "On the Scalability of Rendezvous-based Location Service for Geographic Wireless Ad Hoc Routing", Elsevier Computer Networks, vol. 51, no. 13, **(2007)**, pp. 3693-3714.
[5] S. Basagni, I. Chlamtac, V. R. Syrotiuk and A. W. Barry, "A Distance Routing Effect Algorithm for Mobility (DREAM)", Proceedings of The 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MOBICOM). Dallas, USA: ACM Press, **(1998)**.
[6] JLJJD SJ, R. K. De Couto David and R Morris, "A Scalable Location Service for Geographic Ad Hoc Routing", Proceedings of The 6th Annual ACM International Conference on Mobile Computing and Networking (MOBICOM). Boston, USA: ACM Press, **(2000)**.
[7] I. Stojmenovic and B. Vukojevic, "A Routing Strategy and Quorum Based Location Update Scheme for Ad Hoc Wireless Networks", Ottawa: University of Ottawa, **(1999)**.
[8] H. Cheng, J. Cao and H. Chen, "GrLS: Group-Based Location Service in Mobile Ad Hoc Networks", IEEE Transactions on Vehicular Technology, vol. 57, no. 6, **(2008)**, pp. 3693-3707.
[9] L. F. M. Vieira, U. Lee and M. Gerla, "Phero-Trail: a Bio-inspired Location Service for Mobile Underwater Sensor Networks", Proceedings of the 14th Annual International Conference on Mobile Computing and Networking (MOBICOM). San Francisco, USA: ACM Press, **(2008)**.

[10] L. F. M. Vieira, U. Lee and M. Gerla, "Phero-Trail: A Bio-Inspired Location Service for Mobile Underwater Sensor Networks", IEEE Journal on Selected Areas in Communications, vol. 28, no. 4, **(2010)**, pp. 553-563.
[11] H. Saleet, O. Basir, R. Langar and R. Boutaba, "Region-Based Location-Service-Management Protocol for VANETs", IEEE Transactions on Vehicular Technology, vol. 59, no. 2, **(2010)**, pp. 917-931.
[12] M. Grossglauser and M. Vetterli, "Locating Mobile Nodes with EASE: Learning Efficient Routes from Encounter Histories Alone", IEEE/ACM Transactions on Networking, vol. 14, no. 3, **(2006)**, pp. 457-469.
[13] M. Grossglauser and M. Vetterli, "Locating Nodes with EASE: Last Encounter Routing in Ad Hoc Networks through Mobility Diffusion", Proceedings of the 22th Annual Joint Conference of the IEEE Computer and Communications (INFOCOM). San Francisco, USA: IEEE Society, **(2003)**.
[14] K. Amouris S. Papavassiliou and M. Li, "A Multi-Zone Position Based Routing Protocol for Wide Area Mobile Ad-hoc Networks", Proceedings of the 49th IEEE Vehicular Technology Conference (VTC). Houston, USA: IEEE Society, **(1999)**.
[15] H. Saleet, O. Basir, R. Langar and R. Boutaba, "Region-based location-service-management protocol for VANETs", IEEE Transactions on Vehicular Technology, vol. 59, **(2010)**, pp. 917-931.
[16] Y. Yu, G. Lu and Z. Zhang, "Enhancing location service scalability with HIGH-GRADE", Proceedings of the 1st IEEE International Conference on Mobile Ad-hoc and Sensor Systems (MASS), Fort Lauderdale, USA, **(2004)**.
[17] M. Abdulla and R. Simon, "Characteristics of common mobility models for opportunistic networks", Proceedings of the 2nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks, Chania, Greece, **(2007)**.
[18] K. Abrougui, R. W. N. Pazzi and A. Boukerche, "Performance evaluation of location-based service discovery protocols for vehicular networks", Proceedings of the IEEE International Conference on Communication (ICC), Cape Town, South Africa, **(2010)**.

Authors

Xinyu Yang, received the Diploma in Computer Science and Technology from the Xi'an Jiaotong University of China in 2001 and the Bachelor, Master, and PhD. degrees from Xi'an Jiaotong University in 1995, 1997, and 2001. He has held positions with the Department of Computer Science and Technology at the Xi'an Jiaotong University, where he is a professor teaching in the Department of Computer Science and Technology. His main technical interests lie in the areas of network security, wireless communication and mobile ad hoc networks.

Linqi Li, received the B.S and M.S degree in computer science and technology from Northwestern Polytechnical University, in 1999 and 2004 respectively. He have attended the China 863 project ''Research on Routing in heterogeneous mobile ad hoc networks''. He is now pursuing his Ph.D. degree in Xi'an Jiaotong University. His current research interests include routing and location service in mobile ad hoc network

Xiaojing Fan, received the B.S degree in computer science and technology from Xi'an University of Posts and Telecommunications, in 2008, and her M.S degree in from Xi'an Jiaotong University in 2011. She have attended the China 863 project ''Research on Routing in heterogeneous mobile ad hoc networks''. She is an IEEE student member, and her major research interest lies in the area of routing and location service in mobile ad hoc networks.

