

An Enhanced Map-Based Location Service for VANET

Deling Huang^{1,*}, Chang Su² and Yusong Yan³

¹ College of Software Engineering, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

² Information Center, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

³ College of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, Sichuan, China

*Corresponding author

Abstract—Position-based routing is a promising technology for VANET. And location service is the fundamental component for the success of position-based routing. A map-based location service algorithm is proposed in this paper. It uses a predefined hash function to map a node to a certain location, where location servers are recruited. Hence any node in the network can find location servers of others, then obtain the position of others. This kind of server deployment scheme makes a distributed location service in the network, the system is subsequently more convenient and robust. Simulation results exhibit that the proposed algorithm improves the query success ratio and reduces the network traffic when compared with state of the art location service scheme.

Keywords—location information; position-based routing; VANET; location update; distributed

I. INTRODUCTION

Vehicular Ad Hoc Network (VANET) is a special kind of Mobile Ad Hoc Network (MANET), which takes advantage of wireless networks and transportation systems. Vehicles in this network can communicate to each other without centralized authority. Hence kinds of information can be shared in the network in a scalable manner. However, topology of VANET changes frequently due to the high speed of nodes, routing for VANET is still a challenge. Numbers of routing protocols have been proposed, and they're categorized into two categories: topology-based and position-based. Studies have shown that position-based routing protocols are more stable and outperform topology-based ones.

Position-based routing protocols forwards packets on the basis of the position information of the ultimate destination and the direct neighbors. Usually intermediate nodes between the source and the destination will be selected mainly according to their location relative to the destination. Researchers also take into account some other information to help selecting the intermediate forwarding nodes. They proposed several protocols for VANET, considering some other factors when choosing the intermediate node, to improve the routing performances. They seldom suspect the accuracy of the location information of the ultimate destination, which is a prerequisite to any position-based routing algorithm. However, acquiring the accurate location of nodes out of one's radio range is challenge in VANET where nodes are always moving fast.

The rest of the paper is organized as follows. Section 2 explains the location service for position-based routing protocols. In section 3, we details our design scheme of location service mechanism based on a hierarchical partition of digital street maps. Section 4 verifies the efficiency of our proposed algorithm through the simulation results. In section 5, we conclude this paper and present the future work.

II. RELATED WORK

The fundamental component for the success of position-based routing is location service. Several algorithms have been proposed, location information can be stored centralized or distributed, in fixed infrastructure or in mobile vehicles themselves. Studies[1][2] have shown that distributed and infrastructure-free ones are more robust and suitable for networks consist of fast moving nodes.

The location service proposed in Distance Routing Effect Algorithm for Mobility (DREAM)[3] does not rely on prior fixed infrastructure. It uses flooding to spread location information of all the nodes in the network. Each node broadcasts its position information with varying frequency and range. The broadcasting frequency is decreased when the nodes' mobility is decreased or the flooding range is increased. Hence every node have some position information of the others in the network. Nodes store this information in a local table called location table, in case that it communicates with one of them. However, a node may find no entry in its location table about the node it would like to communicate with. Under this condition, partially flood as recovery procedure is activated again. As a result, the main disadvantage of DREAM is network congestion caused by flooding position information or location queries in dense networks. The Reactive Location Service (RLS) [4] uses flooding only in request message when a nodes needs to communicate with another node in the network. It refrain from periodically broadcasting position information in DREAM. But the location request is flooded to all nodes in the network, which may also result in network congestion.

Grid Location Service (GLS)[5] is also a decentralized location service without need of prior deployed fixed infrastructure, which does not need flooding. It partition the world into a hierarchy of grids. Four order-n squares make up an order-(n+1) square. The grid-based partition is arbitrary on the requisite that the hierarchical partition of the space is balanced and without any overlap. In order to avoid any node

to be a bottleneck, the location servers are spread evenly. No such node has a priority to be a location server, and every node could be a location server for others. Each node in the network recruits a location server in every level grid of the hierarchy. The rule is choosing the one with the least ID greater than the node itself to be its location server. Hence the location servers for a node are distributed in the whole network whereas are in ahierarchy architecture. The far the grid is away from the node, the sparse the location serves are deployed. When a source needs to send a packet to another node, it looks up in its location table to find an entry matching the ID of the destination. If there is no such entry, the source will initiate a location query to the node, which is chosen using the same computing method when a node is recruiting a location server. The request will be answered if the node has the required message, or it will be forwarded hop-by-hop until it eventually received by a location server of the destination. The disadvantage of GLS is the update or query is forwarded through a chain. If any node in this chain failed, the update or request message would lost. To avoid this problem, HLS introduces the concept of responsible cells for improving the robust.

Hierarchical Location Service (HLS)[6]partitions the network in cells. And all nodes in the network know the partitioning. One of the prerequisites is any two nodes in the same lowest-level cell must communicate to each other. The cells are organized hierarchically into regions of different levels. A few cells forms a level-1 region, and a few level-n regions forms a level-(n+1) region. HLS recruits location servers in a set of responsible cells. A node N selects one responsible cell in each level of the hierarchy. The selecting algorithm is mainly relying on a hash function based on characteristics of the responsible cells set and the node ID of node N. HLS exploits two methods to update location information, either direct or indirect. When a node wants to communicate with another node T out of its radio range, it computes the responsible cells which T would choose if it is in the same level-1 region, and sends its request to it. Any node in the candidate cell who received the request packet first can answer if it has the require information. Or else it will broadcast the request to all nodes within the same cell to waiting for the answer. If none of the nodes in the cell has the information, the request would be sent to the responsible cell of the next level. HLS also provide a mechanism if a responsible cell happens to have no node in it. However HLS has employed a soft state approach to overcome location server failures, which may result in unnecessary location information updates, and consequently more protocol overhead.

The above mentioned and most existed location service are designed for MANET, only a few literatures have worked for VANET and haven't been well adopted. Map-Based Location Service(MBLS)[7] is a well-known algorithm especially designed for VANET. It assume that vehicles in the discussing network have GPS devices and digital maps on board. Vehicles can get their own geographic positions though GPS devices, and street topology from Digital maps. The geographical area is partitioned in the same way as GLS does. A predefined hash function is introduced to map a node to a waypoint at each level of the hierarchy. Then a certain node in the road intersect

is chosen as a location server. Though this hash function, all nodes can find the location servers of others, then acquire the position information of others, which can be used for position-based routing protocol. However, some problems still exist in MBLS. As a result, the vast majority of position-based routing protocols proposed for VANET uses location service designed for MANET directly, which may degrade the routing performance[8][9]. Our work is to enhance MBLS, making it more efficient and capable to provide location service for other position-based routing protocol in VANET.

III. ENHANCED MAP-BASED LOCATION SERVICE

The same assumption is introduced with Enhanced Map-based Location Service (EMLS), that vehicles in the discussing network have GPS devices, digital map sand Geographic Data File on board. Vehicles can obtain their own geographic positions though GPS devices, and get way points for selecting location server from street topology from digital maps. Waypoint, where streets intersects, are represented by unique ID. EMLS introduced the same geographic partition used in MBLS as shown in figure I. The entire network area is covered by level-3 square, which is divided into four level-2 squares, which is further divided into four level-1 squares.

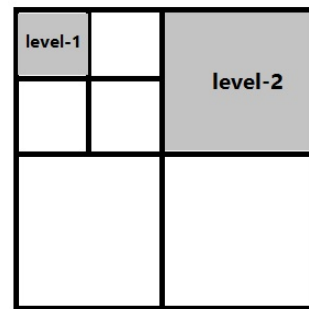


FIGURE I. GEOGRAPHIC PARTITION

After partition, we improve the following aspects of a location service.

A. Location Server Selection

The location servers are selected at each hierarchical level square. A mapping function $W_i = (A + N_{id}) \parallel B$ is used to compute the waypoint ID for a given node in MBLS. Where A is the first waypoint ID, and B is the last waypoint ID of that level square. N_{id} is the ID of node which is being mapped. N_{id} is in a ID space having no relationship with that of waypoints. Hence W_i may sometimes beyonds the waypoint ID boundary formed by A and B. We redesign a mapping function as:

$$W_i = A + [N_{id} \text{MOD} (B-A)] \tag{1}$$

A location update is sent to that waypoint. Three of the nodes around this waypoint are recruited as location server in this level square. A competition is performed among the nodes around this waypoint. Nodes present within a radius R, which is half of the radio range, around this waypoint take part in this competition. The top three with the lowest velocity win the competition. All the velocity information of other candidate nodes can be got from neighbor table stored in the nodes

themselves. This information backup mechanism overcomes the risk of location update or query failure due to a certain server failure of MBLs.

As soon as a node has become the location server of node in level- i square, it will compute the waypoint W_i for that node in level- $(i+1)$, following the same procedure until location servers of level-3 are settled. Meanwhile, other candidate nodes will stop contending for being the location server when they hear three such packets.

We use the same hand-off strategy as is used in MBLs.

B. Location Update

Vehicles are always moving, they have to send their position information to their location server from time to time to keep the accuracy. The more frequent update is, the more accurate the location information is, but the heavier the network load is. There must be a tradeoff between network load and information validity.

Thanks to the moving pattern of vehicles on street is subject to the layout of the street. The location can be predicted unless the vehicle crosses a waypoint. Hence update happens only when the node moves into another road segment or a threshold time is reached. We set a timer to minimizing the quantity of update in case the street is too short and might lead to unnecessary update.

Location update packets are designed as RECORD (N_{id} , Lo , La , W_j , Ts), N_{id} is the node id, Lo and La are the longitude and latitude of the node respectively, W_j is the waypoint ID to which the vehicle is heading and Ts is the timestamp when the packet is generate. Lo and La are added in the packet instead of the two waypoint IDs of the street, in MDLS, vehicle is moving along. This is because they provide more accurate position information, and the road segment information can still be obtained from the digital map the node pre-stored. Speed information is deleted from update packet in MBLs, due to it can be computed by the location server through equation 2.

$$\text{speed} = \frac{\sqrt{(x_{t2}-x_{t1})^2+(y_{t2}-y_{t1})^2}}{T_{s1}-T_{s1}} \quad (2)$$

where (x_{t2}, y_{t2}) is the present location of the load at t_{t2} , (x_{t1}, y_{t1}) is the location of the node at time t_1 , T_{s2} is the timestamp of time t_2 , T_{s1} is the timestamp of time t_1 .

After a location server in level-1 received an update packet, it rebuilds it as RECORD (N_{id} , Level, W_i , Ts), and forward it to the waypoint in level-2. Where Level represent the current level location server is working for, W_i is the waypoint ID the location server is around. The same procedure is performed by the location servers at level-2 until the location servers at level-3 is updated. In our approach, location servers in level-2 and lever-3 only keep the information about the waypoint been chosen in the next lower level. This change from MBLs can reduce the protocol overhead, and can also decrease the upload

frequency. Servers in level-2 and level-3 only need to be updated when the waypoint in the next lower level has changed.

C. Location Queries

The location queries are sent by a vehicle to location servers to acquire the position information of a target vehicle, which couldn't be found in the location table.

If a source vehicle S wants to send message to a destination vehicle D without its position, three possible scenario are:

If S and D are within the same level-1 square, S will compute the location servers of D using the hash function presents in section A, and will certainly get the position of vehicle D.

If S and D lie in different level-1 squares, but same level-2 square. S won't get a reply from D's location servers in level-1, and the request packet is forward to servers in level-2, which will further forward the query to the exact waypoint in the level-1 square D lies in. Location servers around that waypoint will answer S.

The worst situation happens as shown in figure II. S and D are in different level-2 squares. The queries are forwarded sequentially as arrow 1 to arrow 5, and finally get the reply (arrow 6). Query packet takes two more forwards than in MBLs. But this only happens at the beginning of a certain communication, which may avoid dozens of location update packets. When S receives the location information of D, it computes the probable position of D at that moment. Then S caches the location record of D for a period as defined in MBLs.

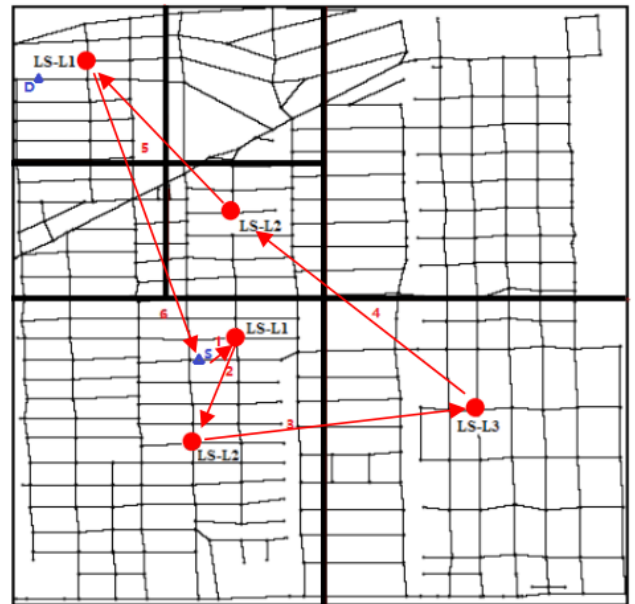


FIGURE II. S AND D ARE IN DIFFERENT LEVER-2 SQUARES

IV. SIMULATION

We use NS-2 to analyze the efficiency of the proposed solution for location service in VANET. A prediction-based

routing protocol[10] is used for position-based routing. We use a real map download from TIGER database from the U.S. Census Bureau. Medium access control (MAC) is IEEE 802.11p, with a radio range of 260m. The mobility traces were generated using VANETMobiSim with the micro-mobility controlled by IDM_IM. Table 1 illustrates the simulation setup.

TABLE I. SIMULATION PARAMETERS

Parameter	Value
Number of vehicles	150 ~ 300
Transmission Range	260m
Area size	2500m × 2500m
Max speed	10 ~ 25 m/s
Degree of geographical partition	3
Simulation Time	300 s

We studied the query success rate under different conditions, comparing with MBLS and DMBS[11].

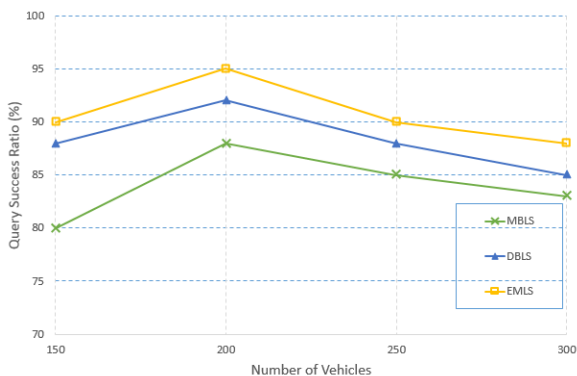


FIGURE I. QUERY SUCCESS VS NODE DENSITY

In figure III, the query success ratio of EMLS overcomes the other two, as its protocol overhead is always smaller than the other two, consequently smaller congestion possibility. And the upload strategy makes the number of packets is always much more smaller than the other two, hence the query success ratio is less susceptible to vehicular density.

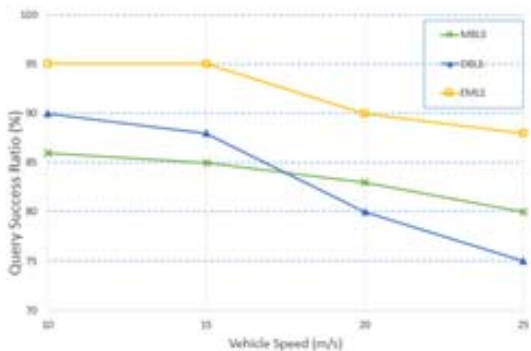


FIGURE II. QUERY SUCCESS VS NODE SPEED

In Figure IV, the query success ratio drops for all the three protocol with the increasing moving speed, which is due to the packet loss caused by the routing. The ratio for DBMS drops largely than the other two, because of the location server selecting schedule it employs. It depends on the density of the intersection, which changes a lot if the vehicles are in high

speed. So the query packet may lead to a different waypoint from the one location information upload to.

V. CONCLUSION

This work aims at improving the accuracy and query success ratio of the location service in vehicular ad-hot networks. The simulation results have proved the proposed scheme has improved the query success ratio in different node densities and moving speeds compared with some other schemes. We are currently extending this work. We will try to give the quantitative analysis on the size of hierarchical level in the future.

ACKNOWLEDGMENT

This research was financially supported by the National Natural Science Foundation of China (Grant No. 61309032, No.61272400), Program for Innovation Team Building at Institutions of Higher Education in Chongqing (Grant No.KJTD201310), China Postdoctoral Fund (No. 2014M562282), the Project Postdoctoral Supported in Chongqing(No. Xm2014039), Scientific and Technological Research Program of Chongqing Municipal Education Commission (KJ1400431, KJ130508).

REFERENCES

- [1] Mauve, M., J. Widmer, and H. Hartenstein, "A survey on position based routing in mobile ad hoc networks". IEEE Network, 2001. 15(6): pp. 30-39.
- [2] Camp, T., J. Boleng, and L. Wilcox. "Location information services in mobile ad hoc networks. in Communications", 2002. ICC 2002. IEEE International Conference on. 2002.
- [3] Basagni, S., et al. "A distance routing effect algorithm for mobility (DREAM)." in ACM/IEEE International Conference on Mobile Computing and NETWORKING. 1998.
- [4] Kasemann, M., et al., "A Reactive Location Service for Mobile Ad Hoc Networks." Arbeitspapier, 2003.
- [5] Li, J., "A scalable location service for geographic ad hoc routing." Proc Acn/ieee Mobicom, 2001: pp. 120-130.
- [6] Kie, et al., "Hierarchical location service for mobile ad-hoc networks." Acn Sigmoblie Mobile Computing & Communications Review, 2004. 8(4): pp. 47-58.
- [7] Boussejra, M., et al. "Map-Based Location Service for VANET." in Ultra Modern Telecommunications & Workshops, 2009. ICUMT '09. International Conference on. 2009.
- [8] Huang, D. and Y. Yan, "A contention-based routing protocol for VANET." Telkomnika (Telecommunication Computing Electronics and Control), 2016. 14(1): pp. 319-325.
- [9] Bilal, S.M., C.J. Bernardos, and C. Guerrero, "Position-based routing in vehicular networks: A survey." Journal of Network and Computer Applications, 2013. 36: pp. 685-697.
- [10] Huang, D., et al., "Prediction-based geographic routing over VANETs." Revista Tecnica de la Facultad de Ingenieria Universidad del Zulia, 2016. 39(2): pp. 157-164.
- [11] Brahmi, N., et al. "An improved map-based location service for vehicular ad hoc networks." in 6th Annual IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, WiMob'2010, October 11, 2010 - October 13, 2010. 2010. Niagara Falls, ON, Canada: IEEE Computer Socie