RESEARCH　　　　　　　　　　　　　　　　　　　　　**Open Access**

Geographic routing based on predictive locations in vehicular *ad hoc* networks

Yanmin Zhu[1,2*], Ruobing Jiang[1], Jiadi Yu[1], Zhi Li[1] and Minglu Li[1]

Abstract

Many geographic routing algorithms have been proposed for vehicular *ad hoc* networks (VANETs), which have the strength of not maintaining any routing structures. However, most of which rely on the availability of accurate real-time location information. It is well known that vehicles can be intermittently connected with other vehicles. Thus, in such networks, it is difficult or may incur considerable cost to retrieve accurate locations of moving vehicles. Furthermore, the location information of a moving vehicle available to other vehicles is usually time-lagged since it is constantly moving over time. Fortunately, we observe that the short-term future locations of vehicles can be predicted. Based on the important observation, we propose a novel approach for geographic routing which exploits the predictive locations of vehicles. Thus, we have developed a prediction technique based on the current speed and heading direction of a vehicle. As a result, the request frequency of location updates can be reduced. In addition, we propose two forwarding strategies and three buffer management strategies. We have performed extensive simulations based on real vehicular GPS traces collected from around 4,000 taxis in Shanghai, China. Simulation results clearly show that geographic routing based on predictive locations is viable and can significantly reduce the cost of location updates.

Keywords: Vehicular *ad hoc* networks; Geographic routing; Predictive locations; Trace-driven simulations

1 Introduction

The combination of vehicles and wireless communication has created a promising technology of vehicular *ad hoc* networks (VANETs) [1]. A wide range of appealing applications can be developed with VANETs, from driving safety [2] to urban monitoring [3]. Compared to static sensor networks [4-6], a network of mobile vehicles can cover larger area and provide useful sensing information about the surrounding environment. VANETs exhibit some unique characteristics. First, vehicles have high mobility and the topology of a VANET can be changing quickly over time. The connection between vehicles is frequently disrupted. Second, being spread over a vast area, the vehicles may be unable to form a connected network. Third, the vehicles are distributed with different densities. These characteristics together suggest that most existing designs and solutions for traditional network systems,

such as MANETs and the Internet, cannot be directly applied.

Efficient routing is essential to VANETs, which determines how a data packet can be delivered from a vehicle to another. Many geographic routing algorithms have been proposed, which share the central idea that a packet destined to a remote vehicle is always forwarded towards the direction of the destination. By following such a routing strategy, the packet can eventually reach the destination node. Most of existing geographic routing algorithms, however, take it for granted that the location information of the destination vehicle is accessed in real time. Nevertheless, such assumption is impractical in the real world considering the cost may be incurred.

A vehicle must maintain its location information over time, and make it available by some means, for example, publishing it to a location server that is accessible to other vehicles. For a vehicle to get the location information of another vehicle, it has to establish a connection with the provider of location information. The data service through GPRS of wireless cellular systems is a practical way for location update and retrieval. Both reporting location information to the location server and retrieving

*Correspondence: yzhu@sjtu.edu.cn
[1] Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
[2] Shanghai Key Lab of Scalable Computing and Systems, Shanghai 200240, China

location information from the location sever incur considerable cost. For example, a volume of 1 MBytes costs about US$1 with China Mobile [7]. Although frequent location update and retrieval can provide more accurate location information, a higher cost is hence incurred. *As a result, it is impractical to assume that location information can be retrieved in real time. In addition, the location of a vehicle maintained at another vehicle is usually time-lagged.*

It is well known that in geographic routing, the location information of the destination is crucial for making decisions of forwarding a packet. If the location information is time-lagged and deviates from the real location, the forwarding decision may result in failure of packet delivery and thus the performance of data delivery of the VANET is degraded.

To address the problem mentioned above, we propose a novel approach for geographic routing which exploits the predictive locations of vehicles. We have developed a prediction technique based on the current speed and heading direction of a vehicle. As a result, the request frequency of location updates can be reduced. In addition, we propose two forwarding strategies and three buffer management strategies. We have performed extensive simulations based on real vehicular traces collected from around 4,000 taxis in Shanghai, China. Simulation results clearly show that geographic routing based on predictive locations is viable and can significantly reduce the cost of location updates.

The rest of the paper is structured as follows. Section 2 describes the problem background. Section 3 presents the design details of our proposed geographic routing based on predictive locations. In Section 4, we present the performance results of simulations driven by real vehicular traces. In Section 5, we review the related work. Finally, we conclude the paper and introduce future work in Section 6.

2 Network model and local service
In this section, we first introduce the system background with which this routing protocol is developed.

2.1 Network model
We consider a set of moving vehicles, denoted by V. Two vehicles can communicate with each other only when they are close to each other. Here, we do not assume a fixed communication range because in an urban environment, there are obstacles like high buildings which may block the radio propagation from a vehicle to other vehicles.

In this paper, we focus on unicast, i.e., each packet m is originated at one vehicle $s_m \in V$ and has one destination d_m. There is one time-to-live (TTL) beyond which the reception of this packet is useless. Each vehicle may generate packets over time.

2.2 Practical location service
There are many ways for providing location information for a VANET, by which one is able to query about the current location of a vehicle. In the prototype, we have implemented a practical location service which introduces only a reasonable cost.

We describe the location service in the following. Each vehicle has a GPRS channel available with the cellular system. Each vehicle reports, through the general packet radio service (GPRS) channel, its location and speed information every T seconds ($T \in [60, 180]$). There is a location service which hosts all location information of the vehicles, which is connected to the Internet. For a vehicle j, it can retrieve the current location of vehicle i, by querying the location server. Such query can be initiated any time on demand. Note that before the queried location information reaches vehicle j, it must experience a certain delay introduced by the cellular system and the Internet.

3 Design
In this section, we first give the overview of the whole design, then present the technique for predicting future locations of a vehicle. We next describe the forwarding strategies and finally describe the strategies for buffer management.

3.1 Overview
Our geographic routing algorithm based on predictive locations consists of three components, including *location predictor*, *packet forwarding*, and *buffer management*.

The component of location predictor predicts the future location of a vehicle given the location of the vehicle at a previous time instant (i.e., time-lagged location). The component of packet forwarding determines the strategy for forwarding packets. And the component of buffer management determines the strategy of replacing packets when the buffer is full.

3.2 Predicting future locations
We device a location predictor that estimates the current location of a vehicle based on its history location and velocity. It is unavoidable that there is a time gap between the time instance of retrieval and that of the location update. Suppose that node i reported its location P_{t_0} and instant velocity $\vec{v_{t_0}}$ to the database at time t_0. At time t, node j requests for the location of node i. Let Δt denote the time gap, i.e., $\Delta t = t - t_0$. Then, the predictor assumes that node i does not change its velocity (both direction and magnitude). Thus, as illustrated in Figure 1, the estimated current location p'_t of node i is
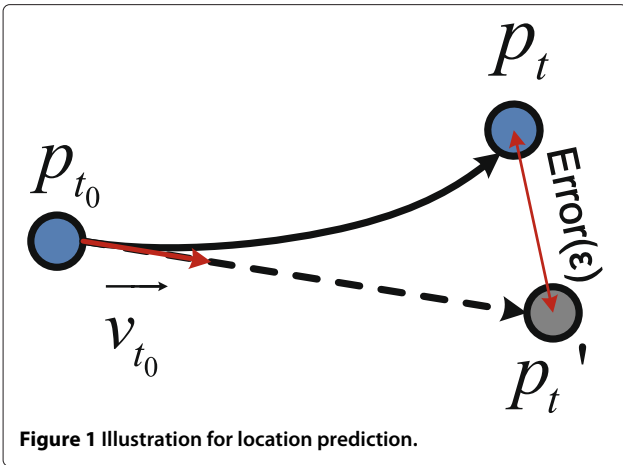
$$p'_t = p_{t_0} + \vec{v_{t_0}} \times \Delta t. \tag{1}$$

Figure 1 Illustration for location prediction.

Figure 2 Error distribution as a function of delay and speed in the prediction.

There is an error ε between the real location p_t and the estimated location p'_t,

$$\varepsilon = p_t - p'_t. \tag{2}$$

To show the performance of this location predictor, we have conducted simulations based on real vehicle traces collected from around 4,000 taxis. Equipped with a GPS receiver, a taxi records its instant speed, direction of heading, and position (in longitude and latitude) every 15 to 30 s, shown in Table 1.

Applying the predicator on the traces, we can compute the real error of each prediction. In Figure 2, we plot prediction error ε against update delay Δt and speed $|\overrightarrow{v}|$. We can find that the error varies for different update delays and different speeds. In general, when the update delay is larger, the error is larger, and when the speed is higher, the error is larger.

When making a prediction, knowing the prediction performance measured as prediction error is important to each vehicle. However, as the vehicle does not know the real location of the target node, it is impossible for it to evaluate the prediction error. To circumvent this issue, we propose a method to approximate this error using the speed and the delay which can both be measured by the vehicle. From Figure 2, we have observed that the error is approximately proportional to the speed of the target vehicle and the update delay. Using linear regression, we obtain the following approximation of the prediction error,

$$\varepsilon = \frac{1}{2} \times |\overrightarrow{v_{t_0}}| \times \Delta t. \tag{3}$$

It is apparent that the time gap Δt has a large impact on prediction error. In general, a smaller *Deltat* results in a smaller error which is better for packet routing. There
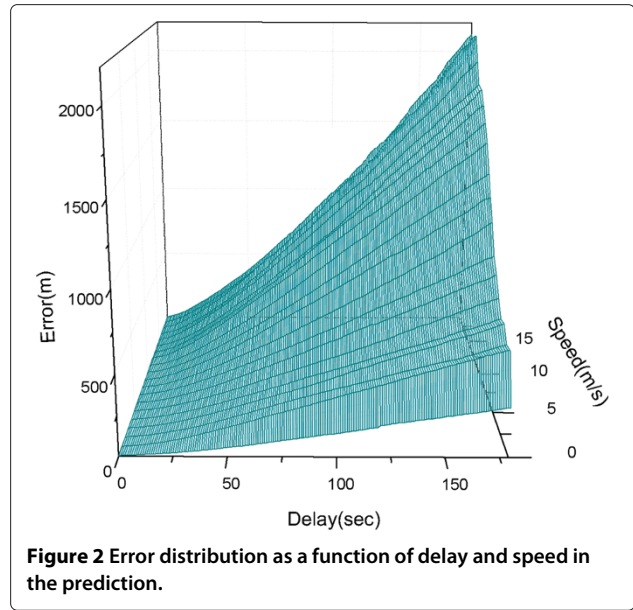
Table 1 The format of each vehicle trace record

TaxiID	Speed	Direction	Long.	Lat.	Timestamp

is a clear tradeoff between cost of location update and routing performance. When the target node is responsible for determining the update interval, the source node has no control on selection of the time gap. When the time gap can be engineered, it should be such selected that the forwarded packet is likely to reach the target node.

The impact of prediction errors varies in different scenarios. Intuitively, when the target vehicle is further away from the predicting vehicle, the impact of a fixed prediction error becomes relatively smaller. We propose a metric called *reliability r* to quantify this relative impact of a given prediction error,

$$r = 1 - \frac{\varepsilon}{p(j) - p_{t_0}}, \tag{4}$$

in which, $p(j)$ is the current position of the predicting node j. In essence, this metric normalizes the prediction error with the relative distance between the predicting node and the destination node. With such a metric, we can better select packets to forward. Intuitively, it is more desirable to select those packets with better prediction accuracy.

3.3 Forwarding packets
The greedy forwarding strategy has two subcomponents: *neighbor selection* and *packet selection*.

Once a new packet is injected into the VANET, the source node queries the location server for the current location information of the destination node, including its physical position, speed, and the corresponding timestamp. The source node appends this information to the packet header. Note that vehicles may update their location information to the location server in a fixed period or

in a unpredictable interval. Relay nodes keep updating this information if the location information of the destination node is updated at the location server.

As a result, when making the forwarding decision, a relay node makes a decision based on the location of the destination node embedded in the packet header.

For the next hop selection, our greedy forwarding algorithm forwards a packet to the neighbor that is closest to the destination node, as shown in Figure 3. Each vehicle maintains the set of active neighbors. This is achieved by each node sending periodic beacons and listening to such beacons. A node is added as a new neighbor if it is not in the current set. An existing neighbor node is removed if no beacon of the node has received beyond a given time threshold. As shown in Figure 3, node C has two neighbors. Node N has the shortest distance to the destination node D out of the two neighbors. Thus, according to greedy forwarding strategy, C forwards the packet to node N.

Another key issue is about which packet among all packets in the buffer should be forwarded first. Due to the limited bandwidth and short contact duration, only a limited number of packets can be transferred during a contact. That means the relay node cannot forward all packets in this contact. Thus, the selection of packets to forward is of great importance.

We design two strategies to sort the packets for forwarding:

Nearest first strategy (NFS). The packet that is closest to its destination takes the highest priority to be forwarded. To sort the packets, each packet m is assigned a priority score,

$$G(m) = r(d_m)/\beta(m), \qquad (5)$$

where $\beta(m)$ is the residual distance of the packet (i.e., the distance between the relay node to the destination d_m).
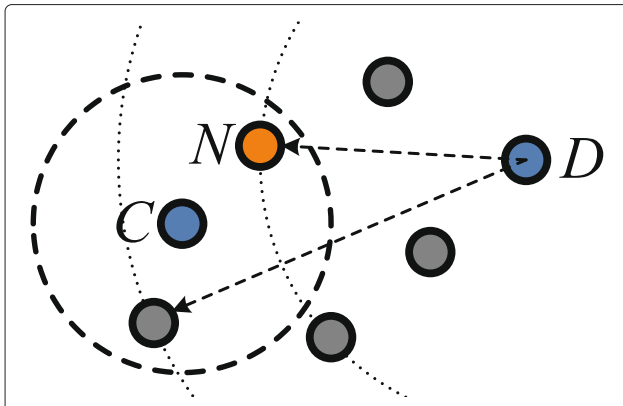

Figure 3 Next hop selection. Node C wants to forward a packet to node D. The packet is forwarded to node N that is the nearest to D in the set of C's neighbors.

On deciding which packet to forward, a relay node forwards packets in the decreasing order according to the priority score. This strategy favors those packets that are closer to their destinations.

Maximum jump strategy (MJS). The packet that makes the maximum jump is forwarded first. The packet with the maximum jump is the one with the maximum reduction on the distance to its destination if this packet is forwarded to the next hop. To compute the priority score of each packet, we use the reliability as weight (which is defined in Equation 4),

$$G(m) = r(d_m) \times f(m), \qquad (6)$$

where $f(m)$ is the maximum forwarding step. The packets are selected to forward in the decreasing order of $G(m)$. This strategy gives more importance to the packets that have traveled a long distance.

3.4 Managing buffers

A vehicle has a limited buffer, and hence, we have to consider buffer management. When the buffer of a vehicle is full, we need a strategy to discard some existing packets in order to make space to accommodate new packets. We implement three buffer replacement strategies:

- Residual distance strategy (RDS): the packet with the maximum residual distance to the destination node is replaced first. Such packets are expected to linger in the system for longer time and consumes more network resources.
- Elapsed time strategy (ETS): the packet with the maximum time elapsed from departure from the source node is replaced first with the observation that such a packet is less likely to reach the destination soon.
- Rejection strategy (RJS): no replacement is made. Upon reception of a new packet, the relay node responds to the sender with an indication of full buffer. The sender then selects another neighbor as the next hop.

4 Evaluation

We compare the performance of the proposed geographic routing based on predictive locations (GRPL) with Greedy Perimeter Stateless Routing (GPSR) [8] which always queries for the most update location of every target vehicle and thus incurs a high cost of location update. We have conducted extensive trace-driven simulations based on a large dataset of real vehicular traces. As introduced in Section 3, the dataset of traces is collected from more than 4,000 taxis running in the urban area of Shanghai, China. The traces had been collected in November 2006 and span a duration of 15 days.

4.1 Simulation setup and methodology

We adopt the simulation setting parameters as shown in Table 2. Two vehicles can communicate with each other when they are both in the communication range of each other. The source-destination pairs are randomly selected. Two types of traffic conditions are studied. In a light workload scenario, 6,000 pairs of source and destination nodes are selected. In the heavy workload scenario, 12,000 pairs are selected. Each source node generates one packet for its destination and each packet has single copy. For each simulation setting, we compute the average of 10 runs.

In performance evaluation, we focus on the following four performance metrics.

- *Delivery ratio*: the ratio of successfully delivered packets to the total packets to transfer.
- *Average delay*: the average delay of all the packets that are successfully delivered.
- *Transmission cost*: the ratio of the number of packet transmissions to the number of successfully delivered packets. This metric indicates the average number of transmissions needed to successfully deliver a packet.
- *Location update cost*: the number of total location updates over the number of successfully delivered packets. This metric indicates the average number of location updates needed to deliver a packet.

We compare our GRPL protocol with GPSR which adopts the first-in-first-out (FIFO) packet selection strategy and the RJS buffer management strategy.

4.2 Impact of location update frequency

We study the performance of GPSR and our GRPL routing with different packet selection strategies under different location update period. The performance of GRPL varies according to different update frequencies while GPSR keeps the same performance as it always requests locations when needed.

Figure 4 plots the delivery ratio of three routing protocols. GRPL-MJS has the best delivery ratio when the location update period is shorter than about 70 s. In Figure 5, the average delivery delay of three protocols are plotted. Our GRPL-NFS routing has the best performance, and when the update period is 50 s, its delay is 20 and 30 s

Table 2 Parameters used in the simulations

Parameter	Value
Communication range	200 m
Packet size	256 KB
Buffer size	8 MB
Bandwidth	2 Mbps

Figure 4 Delivery ratio vs. location update period.
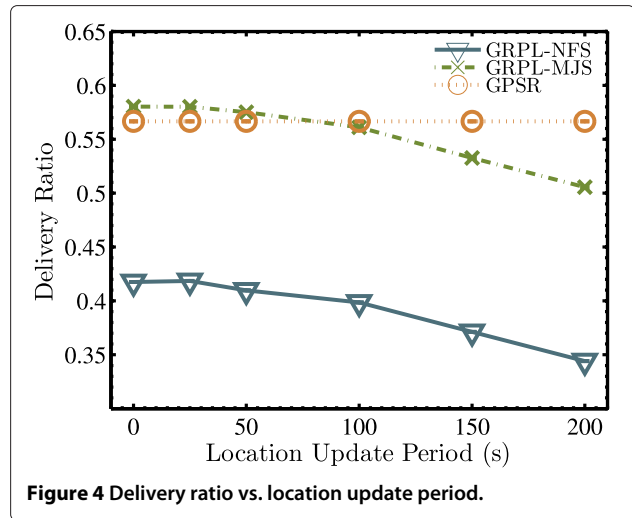
lower than that of GRPL-MJS and GPSR, respectively. Because GRPL-NFS gives the priority to packets that are near their destinations, packets with lower delay are more likely to be delivered. Figure 6 shows the average transmission cost of three protocols. GRPL-MJS and GPSR have similar performance. But GRPL-NFS has a higher transmission cost mainly because of its low delivery ratio. In Figure 7, the number of location updates of our two protocols decrease when the update period increases. When the location update frequency is very high, e.g., once every 2 s, GPSR has the least number of requests. This is because GPSR uses FIFO packet selection strategy which does not need the destination location of each packet. However, for our GRPL routing, both GRPL-MFS and GRPL-MJS need destination locations of all the packets to select one to forward. When the update period increases, the number of location updates for our two routing strategies decrease
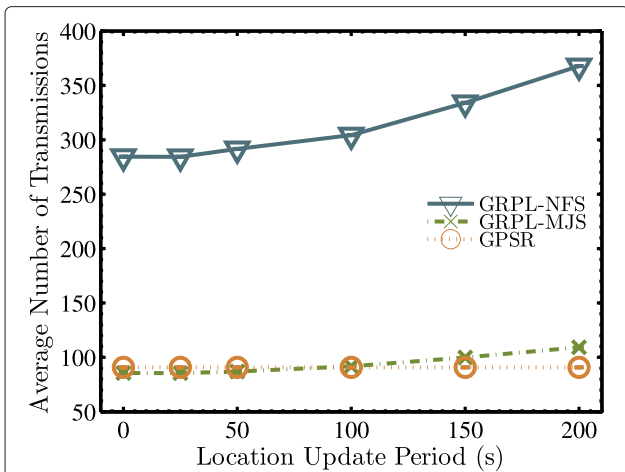

Figure 5 Average delay vs. location update period.
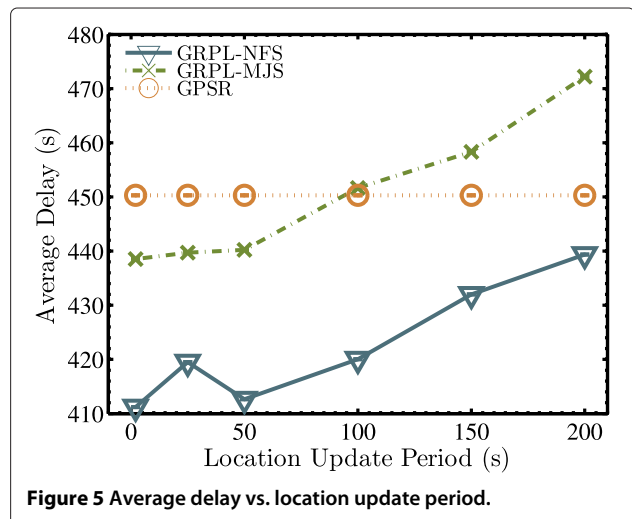
Figure 6 Average number of transmissions (per delivered packet) vs. location update period.
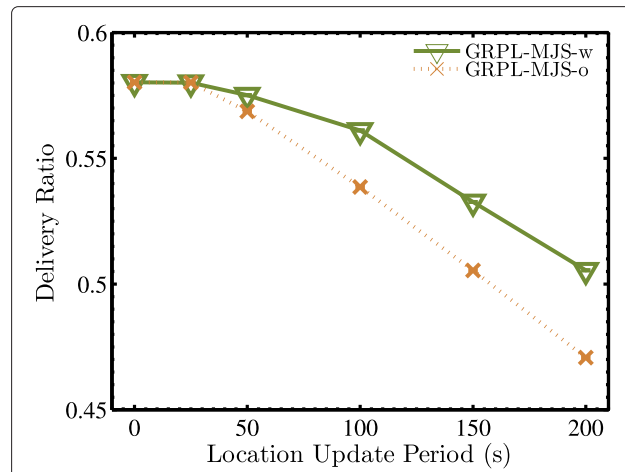

Figure 8 Delivery ratio of GRPL-MJS with/without prediction vs. location update period.

sharply. For example, when the update period is longer than 25 s, the average number of updates of GRPL-NFS and GRPL-MJS are 64% and 73% lower than that of GPSR, respectively.

4.3 Effect of location prediction

We study the effectiveness of location prediction by evaluating the delivery ratio and location update efficiency for GRPL-MJS with and without location prediction. As shown in Figures 8 and 9, the protocol with location prediction has higher delivery ratio and lower transmission cost than the protocol without location prediction when the location update period increases. When the location update period is longer, the error between the previous updated position and the current position of a node becomes larger. Thus, location prediction can be used to improve the packet delivery performance.

4.4 Effect of reliability weight

Applying the reliability weight filters poor location prediction in making a forwarding decision. We study the effect of applying such a reliability weight by evaluating the performance of our GRPL-NFS and GRPL-MJS strategy with and without applying reliability weight. Figure 10 shows the evaluation results of packet delivery ratio. For GRPL-MJS and GRPL-NFS, applying the reliability weight contributes a performance improvement of 9.5% and 8.2%, respectively, over the condition without the reliability weight.

4.5 Effect of buffer management

We study the effect of buffer management under different buffer replacement strategies. In this set of simulations, the routing protocol is the combination of GRPL-MJS with location prediction weighted by reliability. In addi-
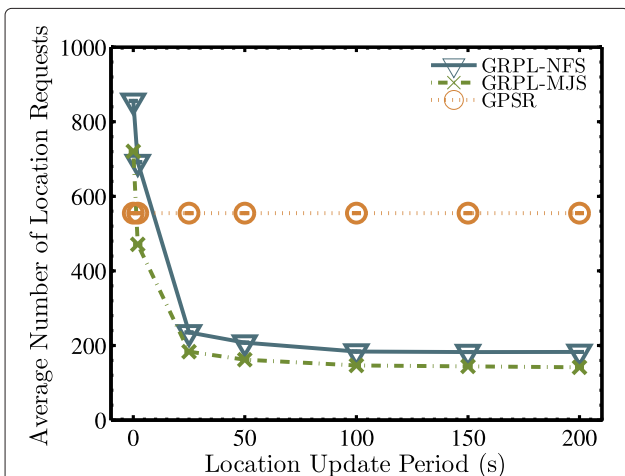

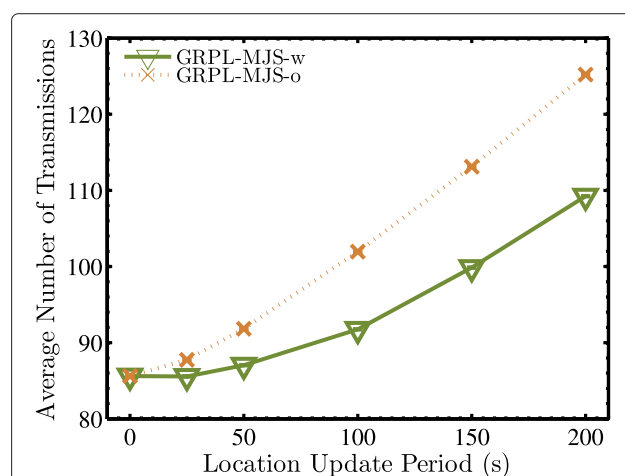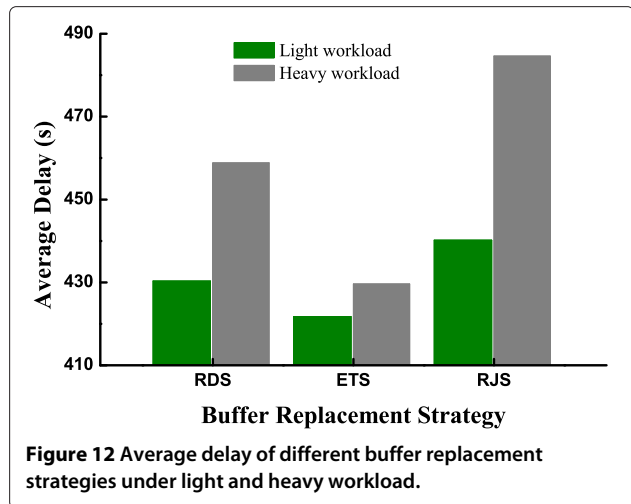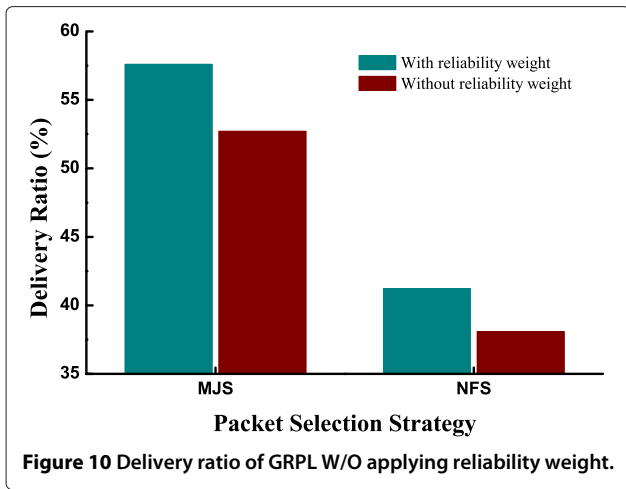**Figure 7 Average number of location requests vs. location update period.**

Figure 9 Average number of transmissions of GRPL-MJS with/without prediction vs. location update period.

http://www.itrans24.com/landing1.html
www.trans24.ir
آنلاین: سفارش ترجمه این مقاله
تماس: (۰۲۱) ۴۰-۶۶۵۷۲۲۳۸
trans24

Figure 10 Delivery ratio of GRPL W/O applying reliability weight.

Figure 12 Average delay of different buffer replacement strategies under light and heavy workload.

tion, we also investigate the impact of traffic load on routing performance. A VANET relies on intermittent connections of nodes, and thus, the aggregate capacity is limited. The traffic load qualifies the total transmission demands from the vehicles. When the traffic load is light, each vehicle generates one packet per minute, and when the traffic light is heavy, each vehicle generates ten packets per minute. The location update period is set as 50 s. The performance results of four metrics are shown in Figures 11 and 12 and Table 3.

In Figure 11, the delivery ratio of GRPL-MJS with three buffer replacement strategies, namely, RDS, ETS and RJS are shown. Under light workload, the performance of three strategies are similar. While under heavy workload, the delivery ratio has a decrease of 19.5%, 27.7% and 19.1% for RDS, ETS and RJS, respectively. ETS has the biggest decrease as it drops the oldest packet when the buffer is full. However, the older packets in the network are more likely to reach their destinations soon. Figure 12 presents

the average delay of three strategies. ETS has the shortest delay under both light and heavy workloads because it gives priority to young packets with shorter delay. The RJS has the longest average delay under both workloads mainly because its high delivery ratio.

Table 3 gives the performance results of transmission cost and location update cost. Both transmission cost and location update cost of ETS and RJS increase as the workload increases, while RDS has a lower cost for both performance metrics as the workload increases. The reason for this result is that although the delivery ratio decreases under heavy workload, the number of delivered packets actually increases. As a result, RDS has an efficient performance in terms of transmission and location update.

5 Related work
A lot of research efforts have been made to develop routing protocols that are suitable for VANETs. In general, the existing algorithms can be classified into three classes.

5.1 Flooding-based routing
These protocols deliver data by broadcasting many copies. Epidemic routing [9] is a typical one among these protocols. It is firstly proposed for delay-tolerant networks (DTN) [10]. Lately, researchers borrow and use it in VANETs since VANETs share with DTN in frequent
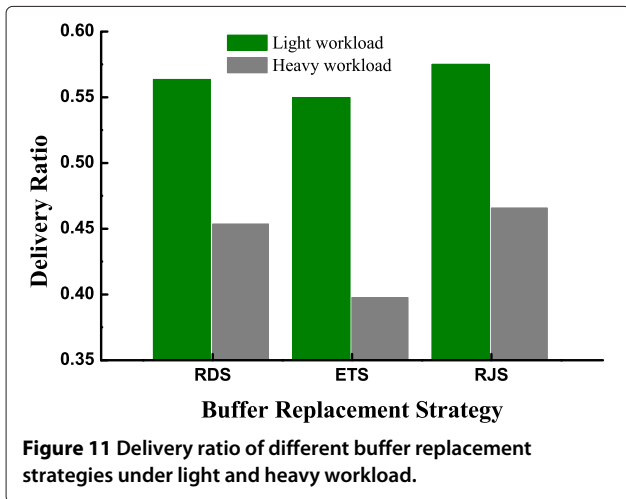

Figure 11 Delivery ratio of different buffer replacement strategies under light and heavy workload.

Table 3 Performance of different buffer replacement strategies under light and heavy workload

Replacement strategy	Transmission cost		Location update cost	
	Light	Heavy	Light	Heavy
RDS	84.13	80.36	135.90	130.68
ETS	86.98	94.40	149.31	174.73
RJS	87.05	92.55	161.82	209.83

disruptions. When two nodes contact, they exchange the metadata of packets they carried and then transfer to each other the packets that the other one does not have. This process is analogous to disease infection. It is assumed that a node has unlimited buffers and bandwidth. This scheme is efficient when little information is on the hand. However, when the buffer size and bandwidth become limited, its performance drops quickly. A lot of variations try to solve the problem of flooding storm by controlling TTL or forwarding times.

5.2 Geographic routing

In contrast to conventional topology-based routing, geographic routing protocols use location information that can be obtained via location sensing device (e.g., GPS). There have been several geographic routing algorithms for vehicular networks [11].

GPSR [8,12] is a typical location-based greedy forwarding algorithm. A node forwards messages to the farthest neighbor within its communication range towards the destination. Some algorithms, such as GSR [13] and GPCR [14], determine a sequence of intersections to connect the source and the destination using the Dijkstra algorithm. CAR [15] discovers a path between the source and the destination by sending path discovery packets based on controlled flooding. Then, the destination node will reply a packet carrying the path information. In [16], the authors introduce a location service to locate the destination node and implement a routing protocol that is similar to GPSR [8]. It is assumed that the location service can provide accurate location information, which is impractical in the real world.

In [13], a method is proposed to generate a routing path on the digital map that has the shortest length. In [17,18], the authors propose a greedy routing protocol, which takes into account the length and traffic condition of a particular road.

Location information is essential for routing in VANETs and many real world applications built on VANETs. As a first step, each vehicle is equipped with a sensing device. The sensing device provides location, velocity, and taxi occupancy. Location and velocity information is measured by a GPS receiver. Thus, the location is in the format of a tuple of longitude and latitude.

Many interesting applications have been investigated, for example, traffic monitoring [19] and file sharing. For a file sharing application, a routing protocol is necessary, with which a vehicle needs to send data to another vehicle. In this paper, we consider that there are n vehicles $V = \{1, 2, 3, \cdots, n\}$. Each vehicle is assigned a unique ID. We consider that a vehicle i may want to send data to another vehicle j, $(i, j \in V)$.

A few algorithms try to make use of road traffic information to find a better path of road sequence for data delivery. A-STAR [20] relies on a statistically rated map, which includes bus route information to assign weights to different roads. GyTAR [17] considers real-time traffic information instead of statistical traffic information for each road. A leader vehicle is selected to gather road traffic within each small fixed-area cell in the road system. RBVT [21] takes advantage of real-time traffic information through route discovery packets which are sent by source nodes to find paths with high connectivity to the destination. Route reply (RR) packets with the intersection sequence are sent back by destination nodes.

5.3 Opportunistic routing

Opportunistic routing focuses on opportunities that two nodes contact with each other. It is believed that the movement of vehicles follows certain laws. MaxProp [22] tries to figure out the probability that two nodes contact with each other from history information and then determine the most likely delivery path to forward packets. In [23], a routing algorithm based on MobySpace is proposed. In this space, the mobility pattern of a node i are described by a vector $V_i = (p_1, p_2, \cdots, p_n)$, where p_j denotes the probability that node i appears at location j. Two nodes that often appear at the same location would probably meet again, when they could exchange data. Thus, forwarding packets to the node that is nearest to the destination in the MobySpace is likely to deliver them successfully.

6 Conclusions

This paper has presented a practical geographic routing protocol for VANETs. The design of such a routing is in response to the unrealistic assumption widely made by previous geographic routing protocols that the accurate location of the destination node can be obtained in real time. Such idealized assumption makes existing routing protocols inappropriate in the real world. The proposed practical routing makes use of a location predictor that estimates the location of the destination node by using its previous speed and location information. In addition, the routing design also considers the practical constraints posed by short contact duration of vehicles on the move. Trace-driven simulation results show that the proposed protocol improves data delivery in VANETs and has a huge reduction of destination location updates than existing routing protocols.

This paper has adopted a relatively simple predictor for estimating the future location of a vehicle. However, the focus of this paper is to investigate the impact of time-lagged location information on the routing performance. GPSR is a classic routing algorithm which uses only most update location information. Comparing to GPSR, the performance gain of our routing scheme can show the significance of time-lagged location information and the

necessity of predicting locations when only time-lagged locations are available. We would like continue to improve location predictors and routing strategies, and carry out more performance comparison with more recent work of routing algorithms in VANETs.

Competing interests

The authors declare that they have no competing interests.

References

1. S Yousefi, M Mousavi, M Fathy, Vehicular ad hoc networks (VANETs) challenges and perspectives, in *ITS Telecommunications Proceedings, 2006 6th International Conference on* (IEEE, 2006), pp. 761–766
2. J Yin, T ElBatt, G Yeung, B Ryu, S Habermas, H Krishnan, T Talty, Performance evaluation of safety applications over DSRC vehicular ad hoc networks, in *Proceedings of the 1st ACM International Workshop on Vehicular Ad Hoc Networks* (ACM, 2004), pp. 1–9
3. U Lee, B Zhou, M Gerla, E Magistretti, P Bellavista, A Corradi, Mobeyes: smart mobs for urban monitoring with a vehicular sensor network. IEEE Wireless Commun. **13**(5), 52–57. (IEEE, 2006)
4. M Li, Y Liu, Underground coal mine monitoring with wireless sensor networks. ACM Trans. Sensor Netw. **5**(2), 10. (ACM, 2009)
5. M Li, Y Liu, Iso-map: energy-efficient contour mapping in wireless sensor networks. IEEE Trans. Knowl Data Eng. **22**(5), 699–710. (IEEE,2010)
6. M Li, Y Liu, L Chen, Nonthreshold-based event detection for 3d environment monitoring in sensor networks. IEEE Trans. Knowl Data Eng. **20**(12), 1699–1711. (IEEE, 2008)
7. Shanghai Division, China Mobile. http://www.10086.cn
8. B Karp, H Kung, GPSR: Greedy Perimeter Stateless Routing for wireless networks, in *ACM MOBICOM* (ACM, 2000), pp. 243–254
9. A Vahdat, D Becker, Epidemic routing for partially connected ad hoc networks. tech. rep., Technical Report CS-200006. Duke University, 2000
10. K Fall, A delay-tolerant network architecture for challenged internets, in *Proc. the Conference on Applications, Technologies, Architectures, and Protocols for Computer Communications* (ACM, 2003), pp. 27–34
11. F Li, Y Wang, Routing in vehicular ad hoc networks: a survey. IEEE Veh. Tech. Mag. **2**(2), 12–22. (IEEE, 2007)
12. P Bose, P Morin, I Stojmenović, J Urrutia, Routing with guaranteed delivery in ad hoc wireless networks, in *Proc. ACM International Workshop on Discrete Algorithm and Methods for Mobile Computing and Communications* (ACM, 1999)
13. C Lochert, H Hartenstein, J Tian, H Fussler, D Hermann, M Mauve, A routing strategy for vehicular ad hoc networks in city environments, in *IEEE Intelligent Vehicles Symposium* (IEEE, 2003), pp. 156–161
14. C Lochert, M Mauve, H Füßler, H Hartenstein, Geographic routing in city scenarios. ACM SIGMOBILE Mobile Comput. Commun. Rev. **9**(1), 69–72. (ACM, 2005)
15. V Naumov, T Gross, Connectivity-aware routing (CAR) in vehicular ad-hoc networks, in *IEEE INFOCOM* (IEEE, 2007), pp. 1919–1927
16. H Füßler, HF Uler, H Hartenstein, M Mauve, M Käsemann, D Vollmer, M Mauve, MK Asemann, Location-based routing for vehicular ad-hoc networks (MobiCom poster). ACM SIGMOBILE Mobile Comput. Commun. Rev. **7**, 47–49. (ACM, 2003)
17. M Jerbi, R Meraihi, SM Senouci, Y Ghamri-Doudane, GyTAR: improved greedy traffic aware routing protocol for vehicular ad hoc networks in city environments, in *Proc. International Workshop on Vehicular Ad Hoc Networks* (ACM, 2006), pp. 88–89
18. M Jerbi, SM Senouci, T Rasheed, Y Ghamri-Doudane, Towards efficient geographic routing in urban vehicular networks. IEEE Transaction on Vehicular Technology. **58**(9), 5048–5059. (IEEE, 2009)
19. H Zhu, Y Zhu, M Li, L Ni, SEER: metropolitan-scale traffic perception based on lossy sensory data, in *IEEE INFOCOM* (IEEE, 2009), pp. 217–225
20. B Seet, G Liu, B Lee, C Foh, K Wong, K Lee, A-STAR: a mobile ad hoc routing strategy for metropolis vehicular communications, in *IFIP NETWORKING* (IFIP, 2004), pp. 989–999
21. J Nzouonta, N Rajgure, G Wang, C Borcea, VANET routing on city roads using real-time vehicular traffic information. IEEE Transactions on Vehicular Technology. **58**(7), 3609–3626. (IEEE, 2009)
22. J Burgess, B Gallagher, D Jensen, B Levine, Maxprop: routing for vehicle-based disruption-tolerant networks, in *Proc. IEEE INFOCOM* (IEEE, 2006)
23. J Leguay, T Friedman, V Conan, DTN routing in a mobility pattern space, in *Proceedings of the 2005 ACM SIGCOMM Workshop on Delay-Tolerant Networking* (ACM, 2005), pp. 276–283