Predictive Directional Greedy Routing in Vehicular Ad hoc Networks

Jiayu Gong, Cheng-Zhong Xu, and James Holle
Department of Electrical & Computer Engg.
Wayne State University, Detroit, Michigan 48202
{jygong, czxu, j.holle}@wayne.edu

Abstract

VANETs (Vehicle Ad hoc NETworks) are highly mobile wireless ad hoc networks targeted to support vehicular safety and other commercial applications. Conventional routing protocols in MANETs (Mobile Ad hoc NETworks) are unable to fully address the unique characteristics in vehicular networks. On the other hand, some characteristics in VANETs, like mobility constraints and predicable mobility, can benefit routing in vehicular networks. It is known that vehicles tend to move more regularly. We propose a new position-based routing strategy with the consideration of nodes moving direction for VANETs, called DGR (Directional Greedy Routing). Considering the fact that vehicles often have predicable mobility, we propose PDGR (Predictive Directional Greedy Routing) to forward packet to the most suitable next hop based on both current and predicable future situations. We evaluate the performance of the solutions via simulations with realistic mobility models in open environments. Simulation results show that our solutions outperform existing ones in terms of packet delivery ratio, end-to-end delay, and routing overhead.

1. Introduction

Vehicular Ad hoc Networks (VANETs) are based on short-range wireless communications (e.g., IEEE 802.11) for the use in road safety and many other commercial applications. The Federal Communications Commission (FCC) has allocated 75 MHz in 5.9 GHz band for licensed Dedicated Short Range Communication (DSRC) for vehicle-to-vehicle and vehicle-to-infrastructure communications. It is expected that more vehicles would be equipped with computing and wireless communication devices in the near future. We assume that vehicles should be equipped with wireless communication devices, GPS, digital maps, and optional sensors for reporting vehicle conditions. Vehicles exchange information with other vehicles as well as road-side infrastructures within their radio ranges. They together form a special Mobile Ad Hoc Networks (MANETS). Its characteristics can be summarized as high dynamics, mobility constraints, predicable mobility, and large scale.

In practice, many applications in VANETs need the support of multi-hop communication. For example, a moving vehicle may want to query a data center located several miles away through VANETs. Multi-hop communication needs routing algorithms. Routing in VANETs is complicated by the characteristics of VANETs, e.g., high dynamics . High dynamics in a large scale network will lead to uneven network density, which varies by time and location. It means the network might be sparsely connected in one area but densely connected in others. On the other hand, some characteristics of VANETs, like mobility constraints and predicable mobility, provide the opportunities to facilitate routing in VANETs.

The existing routing schemes for VANETs, such as [6], [7], [10], [11], [12], improve the performance of routing by considering the fact of mobility constraints and predicable mobility. But there are still limitations. For example, GSR[6] and GPCR[7] employ greedy forwarding based on a pre-selected path. They neglect the case that there are not enough nodes for forwarding packets when the traffic density is low. A-STAR[10] relies on a bus route to identify an anchor path with high connectivity. This approach guarantees to find an end-to-end connection even in the case of low traffic density. But the routing path may not be optimal because it is along the anchor path. The delay can be large. MDDV[11] employs trajectory-based forwarding while considering the traffic density. It assumes the traffic density is static. This may lead to a suboptimal routing path with large delay if the traffic density is varying by time. VADD[12] is designed specifically for sparse networks. This approach uses direction priority instead of a pre-select path to forward packets toward destination. It selects next hop based on the preferred direction and location information in current situation. It predicts the directions of vehicles' movement. But it doesn't predict the future environment change.

This paper studies the routing problem in VANETs in general case, including both dense and sparse net-

27th International Conference on Distributed Computing Systems Workshops (ICDCSW'07)
0-7695-2838-4/07 $20.00 © 2007 IEEE

works. Specifically, when a source vehicle sends a packet to a destination, the routing scheme should be able to efficiently route the packet with few hops and small delay. We propose Directional Greedy Routing (DGR) due to the fact that the nodes movement in VANETs is more regular. By choosing the node moving toward the destination with greedy forwarding, we can reduce the number of hops. We further enhance the approach by making use of the predicable mobility to predict the future environment change. Such prdicable mobility information can be derived from the traffic pattern and street layout. By using current and predicable future information, a more predictive decision can be made when choosing next hop. We refer to this approach as Predictive Directional Greedy Routing (PDGR). Both DGR and PDGR are direction-guided instead of using a pre-selected path. They can forward packets according to the vehicles' movement direction as well as location information. Carry and forward is applied here because high dynamics in VANETs introduces more opportunities for mobile vehicles to meet others intermittently during moving. Simulation results show our proposed strategies outperform existing solutions in terms of packet delivery ratio, data packet delay and routing overhead. Moreover, PDGR can outperform DGR slightly in our test cases.

The rest of this article is organized as follows. In Section 2, the related studies on VANETs routing is presented together with the uniqueness of VANETs. In Section 3, we propose Directional Greedy Routing (DGR) and Predictive Directional Greedy Routing (PDGR). Section 4 evaluates the proposed routing schemes. Finally, we conclude this paper in Section 5.

2. Background and Related Work

In this section, we will survey the existing routing schemes in both MANETs and VANETs in vehicular environments. Before this, we briefly summarize the characteristics of VANETs related to routing.

2.1. VANETs Characteristics

In the following, we only summarize the uniqueness related to routing of VANETs compared with MANETs.

Geographical constraints: Instead of random movement in MANETs, the movement of nodes in VANETs is constrained by the layout of roads. The radio range for VANETs is several hundred meters, typically 250 to 300 meters. In freeway scenario, the nodes can communicate with others in the radio range. But in city environment, there would be radio obstacles because of buildings. Each pair of nodes can communicate directly when they have a "line-of-sight" to each other within the radio range.

High dynamics: In VANETs, vehicles will join and leave the network much more frequently than MANETs since the radio range is small compared with the high speed of vehicles (typically, the radio range is only 250 meters while the speed for vehicles in freeway will be 30m/s). This indicates the topology in VANETs changes much more frequently.

Predicable mobility: Because of the regularity of the road layout, the vehicles mobility can be predicted based on the speed and direction as well as the properties of roads accurately in most cases.

Partitioning and large scale: Because of different traffic density, in some areas, perhaps there is no vehicle that can forward the packets to the destination. That means the network is partitioned into several parts. In addition, VANETs can extend in a large area as long as there is road available.

Mobility models: A realistic mobility model for VANETs can make the simulation results much more accurate. The random waypoint mobility model used in MANETs for analyzing routing behavior is not appropriate for VANETs.

2.2. Routing in MANETs

The routing protocols in MANETs can be classified by their properties. On one hand, they can be classified into two categories, proactive and reactive. Proactive routing (e.g., OLSR [1]) is a table-driven approach. They maintain routing information about the available paths in the network even if these paths are not currently used. The main drawback here is that the maintenance of un-used paths may occur a significant part of the available bandwidth if the network topology changes frequently. Reactive routing (e.g., DSR [4], AODV [8]) is an on-demand approach. They maintain only the routes that are currently in use, thereby reducing the burden on the network while only a small subset of all available routes is in use at any time. VANETs is high dynamic so that routing in VANETs should be reactive.

On the other hand, the routing protocols can be classified into topology-based and position-based (geographic) approaches. Topology-based routing (e.g., AODV [8]) only considers topology connection of the nodes. The drawback is its large latency. To overcome this limitation, position-based (geographic) routing (e.g., GPSR [5]) has been proposed. This kind of routing protocols requires the information about the physical position of the participating nodes. The routing decision is based on the destination's position and the forwarding node's position. Position-based routing does not require the establishment or maintenance of routes.

2.3. Routing in VANETs

Following are a summary of representative VANETs routing algorithms.

GSR(Geographic Source Routing): Lochert et al. in [6] proposed GSR, a position-based routing with topological information. This approach employs greedy forwarding along a pre-selected shortest path. The simulation results show that GSR outperforms topology-based approaches (AODV and DSR) with respect to packet delivery ratio and latency by using realistic vehicular traffic. But this approach neglects the case that there are not enough nodes for forwarding packets when the traffic density is low. Low traffic density will make it difficult to find an end-to-end connection along the pre-selected path.

GPCR(Greedy Perimeter Coordinator Routing): To deal with the challenges of city scenarios, Lochert et al. designed GPCR in [7]. This protocol employs a restricted greedy forwarding procedure along a pre-selected path. When choosing the next hop, a coordinator (the node on a junction)is preferred to a non-coordinator node, even if it is not the geographical closest node to destination. Similar to GSR, GPCR neglects the case of low traffic density as well.

A-STAR (Anchor-based Street and Traffic Aware Routing): To guarantee an end-to-end connection even in a vehicular network with low traffic density, Seet et al. proposed A-STAR [10]. A-STAR uses information on city bus routes to identify an anchor path with high connectivity for packet delivery. By using an anchor path, A-STAR guarantees to find an end-to-end connection even in the case of low traffic density. This position-based scheme also employs a route recovery strategy when the packets are routed to a local optimum by computing a new anchor path from local maximum to which the packet is routed. The simulation results show A-STAR achieves obvious network performance improvement compared with GSR and GPSR. But the routing path may not be optimal because it is along the anchor path. It results in large delay.

MDDV (Mobility-Centric Data Dissemination Algorithm for Vehicular Networks): To achieve reliable and efficient routing, Wu et al. proposed MDDV [11] that combines opportunistic forwarding, geographical forwarding, and trajectory-based forwarding. MDDV takes into account the traffic density. A forwarding trajectory is specified extending from the source to the destination (trajectory-based forwarding), along which a message will be moved geographically closer to the destination (geographical forwarding). The selection of forwarding trajectory uses the geographical knowledge and traffic density. MDDV assumes the traffic density is static. Messages are forwarded along the forwarding trajectory through intermediate nodes which store and forward messages opportunistically. This approach is focusing on reliable routing. The trajectory-based forwarding will lead to large delay if the traffic density varies by time.

VADD (Vehicle-Assisted Data Delivery): To guarantee an end-to-end connection in a sparse network with tolerable delay, Zhao and Cao proposed VADD [12] based on the idea of carry and forward by using predicable mobility specific to the sparse networks. Instead of routing along a pre-select path, VADD chooses next hop based on the highest pre-defined direction priority by selecting the closest one to the destination. The simulation results show VADD outperforms GPSR in terms of packet delivery ratio, data packet delay, and traffic overhead. This approach predicts the directions of vehicles' movement. But it doesn't predict the environment change in the future.

2.4. Discussion

High dynamics of VANETs suggests routing in VANETs should be reactive. Recent comparative studies also showed position-based routing has advantages over topology-based approaches [3]. When applied in VANETs, traditional position-based strategies, like GPSR, have limitations in the aspects of too many hops, routing loops, wrong directions, etc. So new schemes are needed to improve performance. These schemes should be reactive, position-based, and specifically designed for VANETs with consideration of their unique characteristics.

There are still limitations with the existing VANETs routing strategies, as we aforementioned. Additionally, one common defect for GSR, GPCR, A-STAR, and MDDV is that they take no consieration of predicable mobility in VANETs. The predicable mobility in VANETs makes it possible to choose a more efficient next hop. Simply forwarding the packets by a greedy mechanism may not be the best choice if we take into account the velocity and direction of vehicles. In the above routing strategies, only VADD uses predicable mobility for forwarding packets. But it only predicts the directions of vehicles' movements. No prediction is made for the forwarding node's neighbor information in the near future. It may cause more hops and longer delay in routing.

3. Directional Greedy Routing and the Predictive Extension

In this section, we will present the DGR algorithm and its predictive extension. We assume every vehicle has a device to communicate with one another within its radio range. Also it has static digital maps and GPS (or DGPS) installed to get its accurate geographical location. When it wants to send packets to a destination, the destination location is known in advance. We also assume each vehicle has the knowledge of its own velocity and direction.

3.1. Directional Greedy Routing (DGR)

Directional Greedy Routing is based on greedy forwarding under the consideration of nodes movement. It consists of the following two forwarding strategies.

Position First Forwarding: Given the preferred forwarding direction of a packet, the position-first strategy tries to find the closet node towards destination as the next hop. Simple geographical greedy forwarding algorithm can be used here. Though this greedy forwarding algorithm intends to reduce hops and end-to-end delay, in some cases it will have negative effect. For example, node A in Figure 1 is trying to forward a packet to the destination which is along its moving direction while node B is moving in the opposite direction, quite near A. If we simply adopt geographical greedy forwarding, A will forward the packet to B. But after B receives this packet and wants to forward further, it will find A is the suitable next hop. Thus a routing loop occurs. It will lead to more hops to destination and increase end-to-end delay. This infers by adopting Location First Probe strategy alone cannot make routing efficient enough.

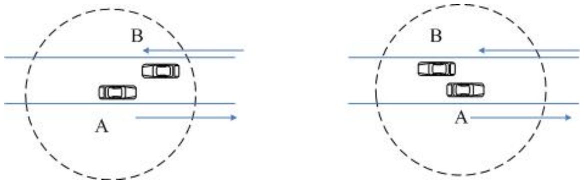


Figure 1: A scenario for routing loop.

Direction First Forwarding: The direction-first strategy will select the nodes moving toward destination. Among those nodes, the one closet to the destination will be chosen as next hop. This scheme intends to reduce routing loops in the forwarding process. But another problem arises when we look into the example in Figure 2. Node A and B are moving towards the destination while C is moving in the opposite direction. A and B are very close. B is closer to destination. C is the closet one to the destination among these three nodes. When A wants to forward a packet to the destination, it can choose B as next hop if only direction-first scheme is used. This may cause more hops and delay.

Our routing approach is designed for the general case in VANETs, which means it is able to perform well in the extreme cases discussed above. It motivates us to take both position and direction into consideration when choosing next hop. To make a tradeoff between the merits of position-first and direction-first forwarding, we propose a mathematical model in (3.1) to reflect the relationship between these two factors. The next hop is selected by calculating weighted score W_i:

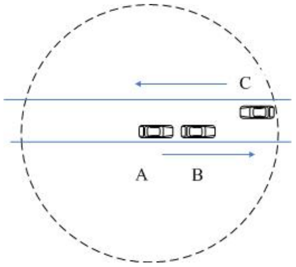


Figure 2: A dilemma in direction first forwarding.

$$W_i = \alpha(1 - D_i/D_c) + \beta\cos(\overrightarrow{v_i}, \overrightarrow{p_{i,d}}).$$

Here, α and β are the weight for these factors and $\alpha + \beta = 1$; D_i is the shortest distance from node i to destination; D_c is the shortest distance for forwarding node to destination; D_i/D_c is the closeness of next candidate hop; $\overrightarrow{v_i}$ is the vector for the velocity of node i; $\overrightarrow{p_{i,d}}$ is the vector from the position of node i to the position of destination; $\cos(\overrightarrow{v_i}, \overrightarrow{p_{i,d}})$ is the cosine value for the angel made by these two vectors.

At one moment, there might be several vehicles which can be closer to destination if they move in their current motion direction. To distinguish the priority of these vehicles according their moving direction, we introduce the cosine value mentioned above. A large cosine value implies a vehicle can still approach the destination closer and closer even after it travels quite a long period of time along its current direction. By applying this mechanism, we can tell which vehicle may move in the optimal direction. But it is not enough if we only use this mechanism. We need location information for more precise decision.

The value of W_i is a weighted score for choosing next hop. The node with largest weighted score among packet carrier itself and its neighbors will be chosen as next hop. If the packet carrier has the largest score, it will carry the packet and forward it later. The packet carrier gets the knowledge of its neighbors' information by beacon messages. Algorithm 1 shows this procedure.

By adjusting the value of α and β, we can make tradeoff between position and direction when forwarding. Extremely, this protocol becomes greedy forwarding if we set $\alpha = 1$ and $\beta = 0$. On the contrary, it becomes Direction First Forwarding if we set $\alpha = 0$ and $\beta = 1$ (of course, one more step to select the node closest to destination along this direction is needed). In practice, we always let $\alpha > \beta$. It is because the value of $(1 - D_i/D_c)$ is very small when the vehicle is faraway from the destination. But the cosine value will

Algorithm 1 Pseudo code for DGR

Notations:

```
$currentnode$: the current packet carrier
$loc_c$: the location for $currentnode$
$\overrightarrow{v_c}$: the speed vector for $currentnode$
$dest$: destination for the packet
$loc_d$: the location for $dest$
$nextHop$: the node selected as next hop
$neigh_i$: the $i$th neighbor
$loc_i$: the location of the $i$th neighbor
$\overrightarrow{v_i}$: the speed vector of the $i$th neighbor

$loc_c \leftarrow getLocation(currentnode)$
$\overrightarrow{v_c} \leftarrow getSpeed(currentnode)$
$loc_d \leftarrow getLocation(dest)$
$D_c = distance(loc_{current}, loc_{dest})$
$\overrightarrow{p_{c,d}} = loc_d - loc_c$
$W = \beta \times \cos(\overrightarrow{v_c}, \overrightarrow{p_{c,d}})$
$nextHop = currentnode$
for all neighbors of $currentnode$ do
    $loc_i \leftarrow getLocation(neigh_i)$
    $\overrightarrow{v_i} \leftarrow getSpeed(neigh_i)$
    $D_i = distance(loc_i, loc_d)$
    $\overrightarrow{p_{i,d}} = loc_d - loc_i$
    $W_i = \alpha \times (1 - D_i/D_c) + \beta \times \cos(\overrightarrow{v_i}, \overrightarrow{p_{i,d}})$
    if $W_i > W$ then
        $W = W_i$
        $nextHop = neigh_i$
    end if
end for
if $nextHop \neq currentnode$ then
    forward the packet to $nextHop$
else carry the packet with $currentnode$
end if
```

not be affected too much in this case. To get a tradeoff here, we set $\alpha > \beta$.

3.2. Predictive Directional Greedy Routing (PDGR)

In DGR, when calculating weighted score for choosing next hop, we only take into account the packet carrier's current neighbors. In fact, a further prediction by considering the packet carrier's possible future neighbors can make routing more efficient.

Consider the example in Figure 3. Node A and B are moving towards destination while B is overtaking A. When A wants to forward a packet to the destination at time t_1, it will not choose B as next hop according to DGR. Instead, A will carry it in this case. After a short interval, B is closer to the destination at time t_2. So at t_2, A will forward the packet to B. But if A forwarded the packet to B at t_1, B can begin next forwarding at t_2, which will decrease the end-to-end delay.

We extend DGR with prediction and propose a predictive approach, Predictive Direction Greedy Routing (PDGR). In PDGR, we calculate the weighted score

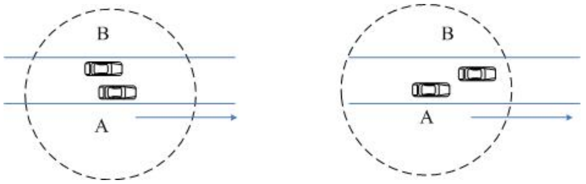


Figure 3: A scenario of overtaking.

not only for the packet carrier and its current neighbors but also for its possible future neighbors in very near future. To get the knowledge of possible future neighbors, the packet carrier requires the information about its 2-hop neighbors, which can also be achieved by beacon messages. According to all these weighted scores, next hop is then decided.

The algorithm for PDGR has two parts. One is to calculate weighted score for current neighbors, which is the same as Algorithm 1. The other is used for future neighbors in a short interval. It is similar to Algorithm 1, but including the steps to get future position of current neighbors and possible future neighbors.

4. Simulation Results and Analysis

In this section, we evaluate the performance of DGR and PDGR in an open environment. A-STAR relies on bus routing information, it's not applicable here. Because GPCR is designed for city environment, MDDV focused mostly on reliable forwarding, and VADD is design for sparse network, they are not comparable with our schemes. So among the routing protocols we aforementioned, we choose GPSR and GSR for comparison.

We use a $2400m \times 2400m$ square street area for simulation. This street layout from simulation area is derived and normalized from a real street map of Wayne County, MI, in Topologically Integrated Geographic Encoding and Referencing (TIGER) database from U.S. Census Bureau. Using the software from [9], we generate realistic traffic model from TIGER format for open environment. The data format of this model can be supported by ns2. Figure 4 shows the simulation setup area.

Traffic density is represented by the number of vehicles. Different number of vehicles can be deployed to the map. Initially, each vehicle will choose one intersection as its destination randomly and move along the street to the destination. The average speed is based on the parameters in TIGER map, usually 15-60 miles per hour. The traffic density in the area is not even. We can vary the traffic density by setting the number of vehicles.

Among all the vehicles, 30 of them are picked up randomly to send data packets to others during the move. The data transmission density can be adjusted by setting different CBR rates. In PDGR, we predict the fu-

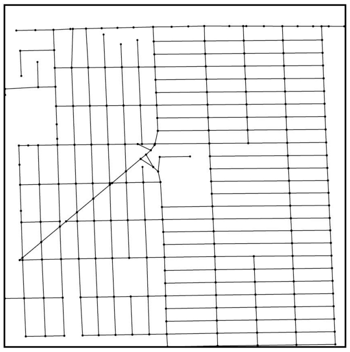


Figure 4: Simulation setup area from a map of Wayne County, Michigan, U.S.A.

Table 1: Simulation parameters.

Parameter	Value
Simulation area	$2400m \times 2400m$
# of vehicles	50, 80, 120, 160, 200, 240
# of packet senders	30
Communication range	250m
Vehicle velocity	15-60 mph
CBR rate (packets/sec)	0.1, 0.2, 0.5, 0.8, 1, 2, 3, 4
Packet size	512 bytes
Vehicle beacon interval	0.5 second
MAC protocol	802.11 DCF
Weighting factor (α, β)	(0.9, 0.1)

ture case after a beacon interval. All simulation parameters are shown in Table 1.

The performance metrics used to evaluate simulation results are:

- Packet delivery ratio: the ratio of the packets that successfully reach destination to the original sent ones;
- End-to-end delay: the average time for a packet from its source to its destination;
- Routing overhead: since all protocols employs beacon messages, the overhead here is the average number of hops for delivering a packet from source to destination.

4.1. Packet Delivery Ratio

In this part, we compare the performance of GPSR, GSR, DGR, and PDGR in terms of packet delivery ratio. We choose packet delivery ratio because wireless systems are usually designed with tolerance to a small packet loss [13]. We will show how packet delivery is affected by the data transmission density and traffic density.

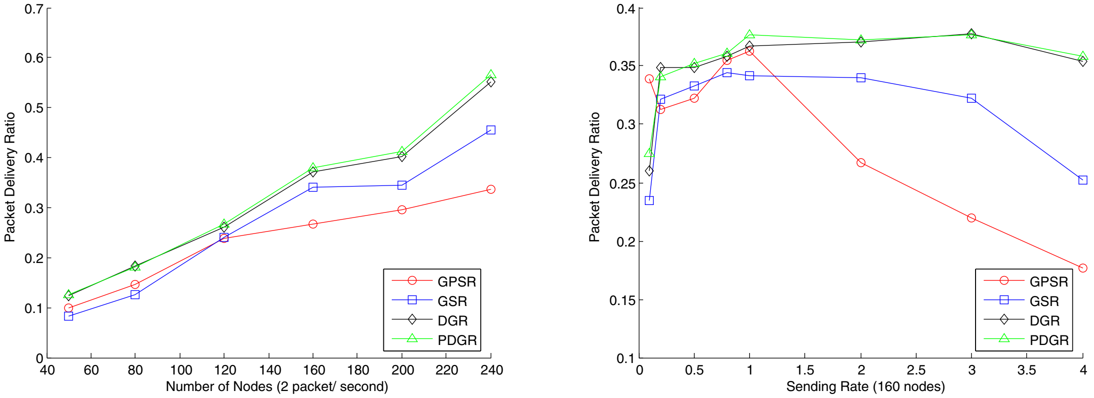
Figure 5 shows the packet delivery ratio as a function of CBR rate and compare the performance under different vehicle traffic densities. As shown in Figure 5(a), with the fixed CBR rate, GSR and GPSR have smaller packet delivery ratio. But the underlying reasons are different. For GPSR, it is due to the property that when it meets the node where greedy forwarding is impossible, it will do the perimeter phase so that it will be less possible to have the chance to forward the packet to the node coming to it in the near future. For GSR, it is because the routing in GSR is along the pre-selected path. Especially, when the vehicle density is low, fewer vehicles will be available for next hop along a specific path. When the vehicle density becomes higher, where the connectivity is much better, all routing strategies achieves better delivery ratio, since more nodes can be met to forward packets. GSR outperforms GPSR in terms of packet delivery ratio when the vehicle density is high. Since DGR and PDGR have no restriction on the routing path and employ direction probing in routing, it is more likely for them to take a node which is closer to the destination and moving toward the destination. So the packet delivery ratios of these two protocols are better than that of GSR. With prediction, PDGR can slightly outperform DGR in packet delivery ratio.

In Figure 5(b), we compare packet delivery ratio of these four protocols in the case of 160 nodes with varying CBR rates. The network will not be able to handle all the packets if the CBR rate increases to a certain value. That is why the packet delivery ratio will drop finally with the increase of CBR rate. Because GPSR has the perimeter phase, it will lead to more packets transmission. So the packet delivery ratio for GPSR will drop more significantly due to high packets sending rate. Because some nodes along the pre-selected path may suffer more packet transmission, this leads to the decrease of packet delivery ratio for GSR with high packer sending rate. For DGR and PDGR, high packet sending rate will not impair their performance too much in terms of packet delivery ratio.

4.2. End-to-end Delay

In this part, we compare the end-to-end delay from the source node to the destination. Note that a low packet delivery ratio may reduce the end-to-end delay because most undelivered packets may experience long delay. In Figure 6(a), the CBR rate is fixed. At first, the end-to-end delay is small for all routing schemes. But this is mostly due to the low packet delivery ratio in the case of high disconnection, especially for 50 nodes. When there are more nodes in the network, the packet delivery ratio increases and some extreme long-delay packets may be involved. So the end-to-end delay is becoming larger. The end-to-end delay for GPSR increases much faster than others. GSR, DGR, and

27th International Conference on Distributed Computing Systems Workshops (ICDCSW'07)
0-7695-2838-4/07 $20.00 © 2007 IEEE



(a) Sending rate = 2 packet/second

(b) 160 nodes

Figure 5: Packet delivery ratio due to different routing algorithms.

PDGR have comparative small end-to-end delay when nodes become more. It is because when there is no node for greedy forwarding any more at some moment, these three schemes will use carry and forward instead of perimeter phase in GPSR. More nodes in network will provide more opportunities to find some suitable node for forwarding intermittently. But for GPSR, the perimeter phase may make it miss this chance. The end-to-end delay for GPSR will drop when the vehicle density is high enough (n=240). It is because high vehicle density will provide more possibility for nodes to do greedy forwarding directly. Avoiding perimeter phase will decrease end-to-end delay. In addition, the restriction for GSR when choosing next lop leads to a larger end-to-end delay compared with DGR and PDGR.

In Figure 6(b), the end-to-end delay increases for all protocols in the case of 160 nodes with varying CBR rates because more packets in the network will lead to more congestion. The end-to-end delay for GPSR increases significantly when the CBR rate becomes large because the perimeter phase will lead to much more packets in the network when more packets are transmitted. But the end-to-end delays for GSR, DGR and PDGR are increasing slowly because they avoid this. More precisely, from this figure, we can observe DGR and PDGR outperform GSR in terms of delay slightly.

4.3. Routing Overhead

We measure the routing overhead by the average number of hops for one packet to reach destination. In fact, this overhead may include the beacon messages. But for all these protocols, beacon messages are used by the same method. So we only compare average number of hops here.

In Figure 7(a), the number of nodes varies with the same CBR rate. When the number of nodes is small, the average number of hops is not big because high disconnection may lead that the successful delivery only happens in a small scope. It is noticeable that GPSR has larger average number of hops than others due to its perimeter phase.

Figure 7(b) shows the average number of hops for GSR, DGR, PDGR will not change a lot with the variant of CBR rate. And DGR and PDGR have a smaller average number of hops than GSR.

5. Conclusion

In this paper we have investigated routing aspects of VANETs. We have identified the properties of VANETs and previous studies on routing in VANETs. We have commented on their contributions, defects and limitations. By using the uniqueness of VANETs, we have proposed a new Directional Greedy Routing (DGR) scheme. Furthermore, we have enhanced the DGR algorithm by using predicable mobility in VANETs.

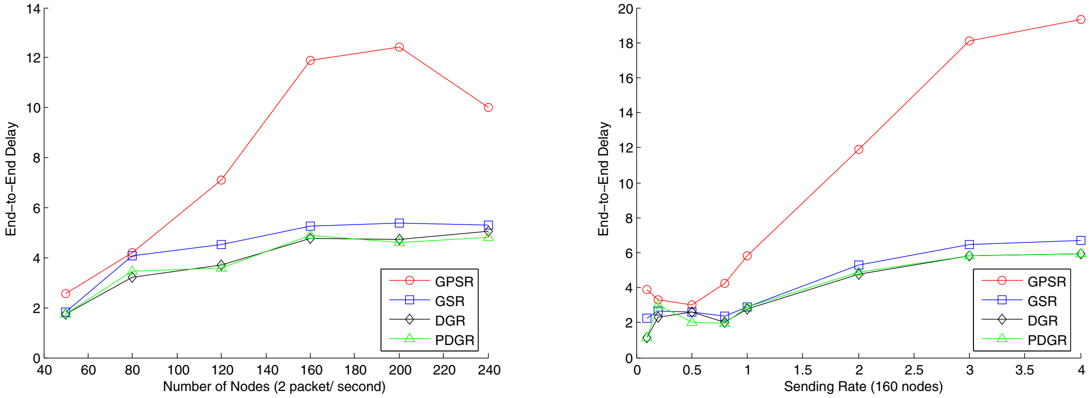
Our simulation results have shown DGR and PDGR outperform GPSR significantly in the terms of packet delivery ratio, end-to-end delay and routing overhead. Also these two strategies outperform GSR. PDGR outperforms DGR slightly because of the use of prediction. In the future, since the routing strategy we proposed is now only simulated for open environment, city scenario requires modifications on our proposed routing strategy by taking into account the city environment characteristics, such as radio obstacles.

Acknowledgment

This research was supported in part by U.S. NSF grants ACI-0203592, CCF-0611750, and MCS-0624849, and NASA grant 03-OBPR-01-0049.

References

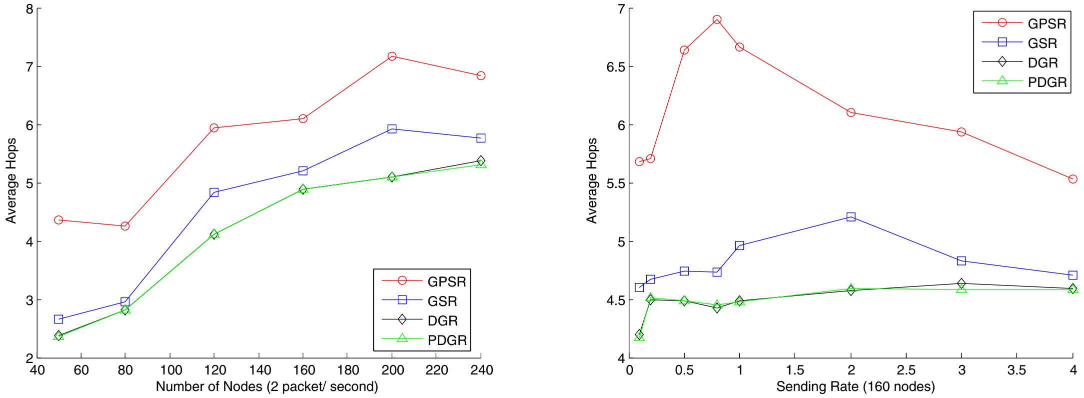
[1] T. H. Clausen and P. Jacquet. "Optimized Link State Routing (OLSR)", RFC 3626, 2003.



(a) Sending rate = 2 packet/second

(b) 160 nodes

Figure 6: End-to-end delay due to different routing algorithms.



(a) Sending rate = 2 packet/second

(b) 160 nodes

Figure 7: Average number of hops due to different routing algorithms.

[2] H. Fubler, M. Mauve, H. Hartenstein, D. Vollmer, and M. Kasemann. “A comparison of routing strategies in vehicular ad-hoc networks”, Technical Report TR-02-003, Department of Computer Science, University of Mannheim, 2002.

[3] H. Fubler, M. Mauve, H. Hartenstein, D. Vollmer, and M. Kasemann. “Location-Based Routing for Vehicular Ad-Hoc Networks” (poster), ACM MobiCom 2002.

[4] D.B. Johnson and D. A. Maltz. “Dynamic source routing in ad hoc wireless networks”, in *Mobile Computing*, Tomasz Imielinske and Hank Korth, Eds, vol. 353. Kluwer Academic Publishers, 1996.

[5] B. Karp and H. T. Kung. “GPSR: Greedy perimeter stateless routing for wireless networks”, ACM MobilCom 2002.

[6] C. Lochert, H. Hartenstein, J. Tian, D. Herrmann, H. Fubler, M. Mauve: “A Routing Strategy for Vehicular Ad Hoc Networks in City Environments”, IEEE Intelligent Vehicles Symposium (IV2003).

[7] C. Lochert, M. Mauve, H. Fler, H. Hartenstein. “Geographic Routing in City Scenarios” (poster), MobiCom 2004, ACM SIGMOBILE Mobile Computing and Communications Review (MC2R) 9 (1), pp. 69–72, 2005.

[8] C. E. Perkins, E. M. Royer, and S. R. Das. “Ad hoc on-demand distance vector routing”, 2nd IEEE Workshop on Mobile Computing Systems and Applications (WMCSA 1999).

[9] A. K. Saha and D. B. Johnson. “Modeling Mobility for Vehicular Ad hoc Networks” (poster), ACM VANET 2004.

[10] B.-C. Seet, G. Liu, B.-S. Lee, C. H. Foh, K. J. Wong, K.-K. Lee. “A-STAR: A Mobile Ad Hoc Routing Strategy for Metropolis Vehicular Communications”, NETWORKING 2004.

[11] H. Wu, R. Fujimoto, R. Guensler and M. Hunter. “MDDV: A Mobility-Centric Data Dissemination Algorithm for Vehicular Networks”, ACM VANET 2004.

[12] J. Zhao and G. Cao. “VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks”, InfoCom 2006.

[13] X. Zhong and C.-Z. Xu. “Delay-Constrained Energy-Efficient Wireless Packet Scheduling with QoS Guarantee”, Globecom 2005.

27th International Conference on Distributed Computing Systems Workshops (ICDCSW'07)
0-7695-2838-4/07 $20.00