Prediction-Based Geographic Routing over VANETs

Deling Huang\textsuperscript{1,2}
\textsuperscript{1}College of Information Science and Technology, Southwest Jiaotong University, Chengdu 610031, Sichuan, China
\textsuperscript{2}College of Software Engineering, Chongqing University of Posts and Telecommunication, Chongqing 400065, China

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

Chang Su
College of Software Engineering, Chongqing University of Posts and Telecommunication, Chongqing 400065, China

Guangxia Xu\textsuperscript{1,2}
\textsuperscript{1}The Information and Communication Engineering Postdoctoral Research Station, Chongqing University, Chongqing 400044, China.
\textsuperscript{2}College of Software Engineering, Chongqing University of Posts and Telecommunication, Chongqing 400065, China

Abstract
The topology formed by vehicles changes quickly, which makes routing become unstable. Geographic routing such as GPSR, compared with traditional routing, is more scalable and feasible. However, the commonly used greedy forwarding in geographic routing often fails, due to the urban topology in VANET is particular. Some improvements have been proposed, such as GROOV, which calculates the feasibility of every neighbor node to make the forwarding decision. And it depends on intersection coordinator to choose the next road segment that packets should relay to. However, it increases the packet travel hops when improves the packets delivery ratio. And it also forces packets to stop at an intersection, thus causes more communication traffic at intersections, which may bring congestion and possibly more relay hops. This paper proposes an enhanced geographic routing protocol called PGR, which enhances GROOV’s forwarding strategy and, through kinds of predictions, reduces the packet travel hops while maintains a relatively high packets delivery ratio. Simulation results show that PGR achieves a high level of routing performance in terms of hop counts, network latency and packet delivery ratio both in dense or sparse vehicular ad-hoc networks.

Key words: VANET, Geographic Routing Protocol, Greedy Forwarding, Local Optimal

1. INTRODUCTION
Geographic routing has become one of the popular routing protocols in Vehicular Ad hoc Networks (VANETs) recent years (Li and Wang, 2007). Unlike conventional on-demand and table-driven routing protocols, geographic routing obviates the costly need of route discovery and maintenance procedures (Hussey et al., 2013). It is hence more suitable in dynamic topology networks (Katsaros et al., 2011; Li, 2009). However, vehicles move fast. Data transmission is less reliable under the highly dynamic topology formed by them. This requires VANET routing protocols must operate reliably in scenarios embracing high speed nodes. Designing efficient geographic routing in VANET is challenging and attracts many scholars.

Numerous geographic routing protocols for VANET have been proposed. They try to adapt to the frequently changed topology.

Intermediate node in GPSR (Karp and Kang, 2000) forwards a packet to a radio neighbor that is closest to the destination. This approach is called greedy forwarding. It is assumed that each node can determine its own position using a GPS. Nodes exchange their locations with neighbors, and obtain the position of the destination by a location service. In some cases, there might be no next-hop when greedy forwarding. GPSR introduced a strategy, called perimeter routing, to overcome this local maximum. Perimeter routing requires the graph to be a planer graph. Many researches (Kim et al., 2005; Frey and Stojmenovic, 2006; Kim et al, 2006) have done much in building a planar vehicular graph, which can become an inaccurate, cumbersome, and continuous process as nodes are highly mobile in VANETs.

GPCR (Lochert et al., 2005) eliminates graph planarization in VANETs. It uses the underlying road represents the planar graph, and omits the entire node planarization process. Packets are relayed as far as they
can at each hop in road segment and are stopped at junctions to determine which next road segment should be taken. This strategy saves the packets’ travel hops and has shown higher packet delivery rate. Hereafter algorithm using repairing mode in VANET have no need to build a planar graph.

GPSR+AGF (Naumov et al., 2006) presents a problem in GPSR when under vehicular environment. The moment a node selects a next hop, as the fast mobile nature of vehicles in VANETs, the using position information in neighbor table may be outdated. The authors proposed AGF that incorporates velocity information in the beacon message. Thus, each source or relay node can filter out outdated nodes from its neighbor table. Also total travel time is added into the beacon message to help each node determine the movement of the destination. Results have shown a considerable improving in PDR to GPSR. However, a significant overhead is involved.

Several routing protocols (Barba et al., 2013; Rabayah and Malaney, 2013; Chang et al., 2012; LIU et al., 2010), employing greedy forwarding, have proved that end-to-end delay could be greatly improved by sharing some extra information. However, these strategies require a relatively high demand on every node and contribute little to the reliability. Finding out greedy forwarding may result in low packet delivery rate in highly volatile networks, GROOV (Sanjay et al., 2012) finds the most reliable relay node using mathematical calculations, rather than making a greedy forwarding. It calculates transmission feasibility for each vehicle’s neighbor, using link quality, range weight and direction when select the next relay node and achieves a high level of routing performance. In the next section, we will make an in-depth discussion about GROOV.

2. BACKGROUND

Each vehicle in GROOV broadcasts beacon message periodically to inform others about its presence. The beacon message contains id, position coordinates, speed, direction and state of the node. The node state indicates whether the node is at an intersection or not. Every node maintains a neighbor table using the information from beacons received.

Packets are forwarded in different manners while they are under different environments. When a node is in the straight road, it calculates the feasibility (P) of each node in its neighbor table. In order to work out P, the radios range (r) of the node is divided into 5 zones with boundaries r/5, 2r/5, 3r/5, 4r/5 and r. Neighbors are weighted based on the region in which they lie. The range weight (w) is 1, 2, 3, 5 and 4 for each region respectively, as is shown in Figure 1(a), where different zones are shown by concentric circles. Also, the link quality is calculated for each neighbor. The one with high-speed variations is assigned with low link quality, as predicting the behavior of such a node is difficult. The last thing to confirm is whether the node is in the direction of the destination. GROOV makes the decision as is shown in Figure 1(b). It believes that node in the shadow region is in the direction of the destination. Finally, a node integrates all the above-mentioned factors, and picks up the neighbor with the highest feasibility to pass on the packets.

![Figure 1. Range weight and direction assignment](image)

When a node is in the intersection scenario, it will calculate the current coordinates of its neighbors, using the location and the speed of them from the neighbor table. Then evaluate the new feasibility for them, and choose the one with the highest feasibility to relay the packets. However, vehicle at intersection always has sudden changes in speed, and it is not reliable to predict the location on the basis of speed. Meanwhile, it is not necessary that all packets are demanded to stop at an intersection. In the next section, we proposed an algorithm, which improves the selecting criteria when a node is on the road segment, and enhanced the prediction when a node is near an intersection.
3. PROPOSED ALGORITHM: PGR

PGR is proposed based on GROOV to optimize the efficiency, it redesigns the forwarding and repair algorithm of GROOV. The purpose of PGR is to reduce the network delay while keeping relatively high packet delivery ratio.

PGR keeps the advantages of GROOV, not using pure greedy forwarding. Furthermore, it enhances prediction before an intersection, and thus saves one hop in some cases. Simulation results show that PGR achieves a high level of routing performance in terms of hop counts, network latency and packet delivery ratio both in dense or sparse vehicular ad-hoc networks.

3.1. Assumptions
1. Each vehicle has GPS equipment on board to obtain its coordinate information.
2. Each vehicle broadcasts beacon messages periodically to inform its directed neighbors about its real-time information.

3.2. Beacon message
The beacon message includes node id, geographic coordinate and a flag, which indicates if the vehicle generated the beacon message is at an intersection or not.

A node can determine itself whether or not at an intersection by checking its one-hop neighbor’s position and comparing them with the road width, as proposed in (Wang et al., 2010). It broadcasts a beacon message, whose flag is set to value 1, to inform the neighboring vehicles immediately when a node ascertain itself at an intersection.

Each vehicle calculates its one-hop neighbors’ speeds and directions using the following two equations (Kumar and Rao, 2008), where \((x_{i2}, y_{i2})\) is the present location in the beacon message from directed neighbor \(i\), \((x_{i1}, y_{i1})\) is the location of node \(i\) in one’s the neighborhood table.

\[
\theta_i = \tan^{-1}\left(\frac{y_{i2}-y_{i1}}{x_{i2}-x_{i1}}\right) \tag{1}
\]

\[
\text{speed}_i = \frac{\sqrt{(x_{i2}-x_{i1})^2+(y_{i2}-y_{i1})^2}}{\text{timestamp}_{i2}-\text{timestamp}_{i1}} \tag{2}
\]

The records of the speed and the direction of a node is removed from beacon message, as there are two obvious advantages. First, this reduces the excessive overhead of beacon messages. Second, this also increases security as no node can tell mendacious information to get the priority to relay packets. Meanwhile, each node doesn’t need to store its position one beacon interval ago for calculating its velocity information.

3.3. Neighbor Table
Each node maintains a neighbor table, using beacon messages received from its neighbors, where node id, node coordinates and intersection flag can be read. Also, every node calculates speed, directions, link qualities and timestamps for all its directed neighbors, and stores them in its neighbor table.

Each time a node receives a beacon message, it refreshes its neighbor table. All the mentioned calculations are done at this moment. Thus it will not contribute to the delay when relay a packet.

3.4. City Straight Road
PGR is a position-based routing protocol which does not employ greedy forwarding. As reference (Sanjay et al., 2012) indicates greedy forwarding may not be the best way based on the location of the destination. If the furthest node is chosen, that is the node on the edge of the radio range of the relay node, thus it is more probably that the node moves out of the relay node’s range during the interval between two beacon messages. PGR, therefore, uses three main factors, link quality (Q), range weight (W) and direction factor (m) to calculate the transmission feasibility (P), which helps predicting the link reliability when the node is chosen as the next hop.

PGR calculates link quality, Q, for every vehicle in the neighbor table. Q is a factor that incorporates the vehicle’s stability as a node in a link. Q varies from 1 to 5, which was defined on the basis of the mathematical concept of average acceleration. Nodes with high-speed variation are arranged with a small Q, while those with small speed variation are arranged with bigger Q. This is because the faster the speed varies, the harder to predict its location, thus the harder to guarantee the link quality. When receiving a beacon message, node refreshes Q for every entry in its neighbor table, using equation 3(Sanjay et al., 2012).
Where \( Q_{i-1} \) represents the value of \( Q \) stored in the neighbor table one beacon interval ago.

To reduce the chances of packet dropping, GROOV gives a smaller weight for the boundary region nodes. This will always result in missing some stable nodes in that region, which will make more progress if are chosen as forwarding nodes. On the other hand, each region covers 50 meters long road length (assuming the radio range is 250m), it is obviously too rough to assign same weight to all the vehicles in the same region. As for these reasons, the average number of hops will be larger and accordingly end-to-end delay will be longer. PGR redefines the calculation of range weight, \( W \), as defined in equation 4.

\[
W = \frac{\sqrt{(x_i-x)^2+(y_i-y)^2}}{\text{radius}_{\text{range}}} \times 5
\]

Where \((x_i, y_i)\) is the position of neighbor under computing, \((x, y)\) is the position of source or relay node who is doing the computing. In this way, packets will make the most progress as they could, while the chances of vehicle’s moving out of the range can be still controlled by factor \( Q \).

The last factor \( m \) decides the sign of \( P \). If the node lies in the direction of the destination, \( m \) equals 1, otherwise \( m \) equals -1. Recall that the proposed protocol is running in Vanet, which makes the direction information simpler, as vehicle’s moving direction is subjected to the street. Therefore, under the city straight road scenario, the direction towards the next intersection is in fact the direction of the destination. Figure 2 illustrates the direction geographically towards the destination does not work well in a city straight road scenario. Under that situation, \( N_1 \) will be chosen by mistake, as its \( m \) factor equals 1. In fact \( N_2 \) is actually towards node \( D \) rather than \( N_1 \). Thus PGR assigns a positive sign to \( m \) if the node is in the road segment closer to the destination (shaded area in Figure 3), otherwise assigns a negative sign to \( m \), as is shown in Figure 3.

Before a data packet is forwarded, the relay node calculates \( P \) for all its neighbors and chooses the one with the highest value of \( P \) as the best relay node. Equation 5 defines the calculation of \( P \).

\[
P = m (Q + W)
\]
3.5. City Intersection Scenario

When a vehicle catches a beacon message from an intersection coordinator, it activates another prediction. It predicts whether the packet will make a turn. PGR calculates the angle between two vectors, \( \text{Vec}_{\text{des}} \) and \( \text{Vec}_{\text{pos}} \). \( \text{Vec}_{\text{des}} \) denotes vector from intersection coordinator to destination, \( \text{Vec}_{\text{pos}} \) denotes vector from source/relay node to intersection coordinator. The intersection mode works as Figure 4.

![Flow Chart of Predictive Mode](image)

**Figure 4.** The flow chart of predictive mode

With calculating the vector angle in advance, PGR can save one hop if there is no need to make a turn at the intersection. In this case, PGR directly forwarding packets to such a next hop that lies on the extension of the present road segment.

3.6. Overcome VOID

Existing position based routing protocols always suffer local optimum. With PGR’s strategy, we enlarge the area of a void. Traditional void is the hatched area and our void enlarged to the shaded area. That means PGR reduces the possibility of encountering a void. When greedy forwarding, \( x \) will not choose the path \( x \rightarrow y \rightarrow z \rightarrow D \). However, node \( y \) is calculated to be the best next hop with our algorithm, and the path will be chosen. Second, if a void presents, PGR can solve it with its own strategy, there is no need to design another mode to overcome the void. Figure 5 depicts the scenario when encounter a void, PGR will always find a way \( x \rightarrow B_1 \rightarrow B_2 \rightarrow B_3 \rightarrow D \) if there is any.

![Scenario of Void](image)

**Figure 5.** PGR overcomes a Void
4. SIMULATION RESULT

In this section, we will compare the performance of PGR with GROOV. The experiments were conducted on NS-2 simulator. Medium access control (MAC) is IEEE 802.11g, with a radio range of 250m. The mobility traces were generated in an area of 1500m ×1500m using VANetMobiSim. The micro-mobility is controlled by IDM_IM (Mohamed et al., 2012). Table 1 illustrates the simulation parameters.

The 20% sender-receiver pairs were randomly selected for each simulation run. We measured the packet delivery rate (PDR) versus the numbers of vehicles participate in (Figure 6). Each point in the graphs comes out from 10 independent runs. PGR gets a PDR comparable to that of GROOV, very rarely lower than that of GROOV. This is because both of the two protocols select the most stable node, using the mobility information, to relay the packet to. Thus, any packet dropping caused by mobility is seldom happen. However, the further the distance is, the higher the attenuation is, and the greater the possibility of packet loss is. PGR, makes every hop as far as possible, while GROOV always gives less priority to the farthest nodes. Thus, PGR may sometimes bear packet dropping caused by the signal attenuation. But in order to overcome this kind of packet loss, giving way to tens of meters in packet’s progress every hop is not advisable. Figure 7 demonstrate the link stability is achieved at the sacrifice of the average hop count.

Table 1. Simulation Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network simulator</td>
<td>NS-2</td>
</tr>
<tr>
<td>Mobility simulator</td>
<td>VANetMobiSim</td>
</tr>
<tr>
<td>Dimension</td>
<td>1500m×1500m</td>
</tr>
<tr>
<td>Simulation time</td>
<td>600s</td>
</tr>
<tr>
<td>Routing protocol</td>
<td>PGR, GROOV</td>
</tr>
<tr>
<td>Average vehicle speed</td>
<td>50 km/hr</td>
</tr>
<tr>
<td>Transmission power</td>
<td>0.005w</td>
</tr>
<tr>
<td>802.11g rate</td>
<td>11Mbps</td>
</tr>
<tr>
<td>Source-destination pairs</td>
<td>Random</td>
</tr>
<tr>
<td>Packet rate</td>
<td>18Kbps</td>
</tr>
<tr>
<td>Packet size</td>
<td>512Byte</td>
</tr>
</tbody>
</table>

Figure 6. PDR vs node density

Figure 7. Hop count vs node density

Figure 7 shows the average number of hops for a packet during routing to its ultimate destination compare with GROOV. The reduction in hop count for PGR is due to two key aspects of improving. First, PGR makes more progress every hop; Second, PGR may hopefully save one hop count when crossing an intersection thanks for the prediction.

Figure 8 depicts the end-to-end delay. PGR shows good results for our approach compared to GROOV. It’s obvious that PGR makes more reasonable choice of the intermediate nodes under comparable PDR with GROOV. Recall that the choice is based on the same computing time complexity as GROOV. Meanwhile we arrange the calculating at the time receiving a Hello message rather than the time relaying the packet. Thus, the computation cost rarely contributes to the latency.
5. CONCLUSION

This work aims at improving the route stability in vehicular ad-hoc networks. The simulation results have proved the proposed algorithm adapts itself to VANET in varying node densities. However, sometimes the routing at an intersection doesn’t give the optimal path. In future, we plan to make the forwarding decision at intersection integrated with city map. That may reduce the hop count, bring consequent reduction on end-to-end delay, and make our algorithm perform better in various scenarios.

Acknowledgments

This work was supported by 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, KJ1400409).

REFERENCES


