A New Greedy Routing Algorithm for Vehicular Ad Hoc Networks

Reihane Baharvand, Seyed Abdolmajid Mousavi

Department of Computer Engineering, Dezful Branch, Islamic Azad University, Dezful Iran
reihane.baharvand@yahoo.com
Faculty member, Lorestan University, Khoramabad, Iran
mousavi.m@lu.ac.ir

Abstract
Vehicular ad hoc networks (VANET\(^1\)) are a new generation of wireless mobile networks playing an important role in communication and safety of vehicles. In VANETs, exchange of traffic and road accidents data improves road safety. Packet routing is a critical issue in VANETs. Due to constant changes of network topology and nodes high moving speed in VANETs, mobile ad hoc network (MANET\(^2\)) routing is not effective. The purpose of this study is to investigate routing algorithms based on greedy approach and propose a weight-based greedy algorithm. The results revealed that the proposed algorithm required lower transmission time and fewer steps compared with existing algorithms.

Keywords: vehicular ad hoc network, vehicular ad hoc network routing, greedy algorithms, greedy routing

I. Introduction

VANETs are wireless sensor networks with several applications in communication between moving vehicles. In VANETs, a group of vehicles provide a dynamic network to communicate and transmit packets. Due to vehicles speed and constant changes of network topology, these networks are different from wireless sensor networks. Existing routing protocols are not very effective for wireless sensor networks since the nodes’ moving speed, velocity, and direction and the changes of these parameters by traffic. In VANETs, routing protocols are divided into 2 major groups: with road side equipment and without road side equipment. Routing protocol without road side equipment is divided into 3 groups: Bus, location-based, and clustering. Routing protocol with road side equipment relies on vehicles and road side equipment. Increased cost is a disadvantage of this method. The proposed algorithm is a location-based

---

1 Vehicular ad hoc network
2 Mobile ad hoc network
greedy method. Thereby, we present some location-based greedy routing methods. Greedy routing algorithms can be divided into 2 groups: single direction and dual direction. In single direction greedy algorithms, packets can only be transmitted to vehicles moving in the direction of source vehicle while in dual direction greedy algorithms packets can be transmitted to all vehicles. Although several studies have been carried out on packet routing in VANETs, there is no study to investigate vehicle moving direction, vehicle density, and the shortest path to next vehicle and destination vehicle simultaneously. Appropriate prioritization of these parameters results in selecting the most effective greedy routing. In this study, a weight-based greedy algorithm is proposed which assigns a specific weight to each of the aforementioned parameters. According to these weights the priority of transmission of packets to different directions is considered and the best greedy path is chosen. Finally, the proposed algorithm is compared with greedy single direction algorithm and dual direction algorithm in terms of number of steps and transmission delay.

II. Related works

In this section previous studies are reviewed. There is a wide variety of routing algorithm in VANETs. We address the algorithms based on greedy approach. Routing algorithms in VANETs can be based on Broadcasting. UMB\textsuperscript{3} routing method based on Broadcasting (Kumari et al., 2013), (Khan et al. 2013), (Paul et al., 2012). And DP-IB routing method which is more reliable for pervasive transmission and slightly solves the problem of flooding transmissions (Gozalvez et al., 2012), (Son et al., 2012). are routing methods with inter-road equipment. In (Shinkawa et al., 2006). The method based on vehicle density has been discussed. The function of this method is based on bus. IVC is a routing method without inter-road equipment based on inter-vehicles interactions (Gang et al., 2006). Similar to GSR\textsuperscript{4}, routing methods based on guide nodes such as A-Star, rely on anchor points. This algorithm improves the rate of pocket delivery using urban traffic data (Naumov et al., 2007), (Seet et al., 2004). Using a map and computing the shortest Dijkstra’s path in which graph vertexes are connection nodes and edges of road are connection of the vertexes, GSR algorithm provides a sequence of path connections towards destination. Next, packets are transmitted greedily. CAR greedy algorithm designed for urban areas and highways applies AODV\textsuperscript{5}, PGB and guard concept to discover path, broadcast data, and maintain path, respectively (Lochert et al., 2005).

\textsuperscript{3} Urban Multihop Broadcast Protocol

\textsuperscript{4} Geographic Source Routing

\textsuperscript{5} Ad Hoc on Demand Distance Vector
The main difference between GSR and CAR is that CAR does not use map and leads packets through finding active path using anchor points (Zhao et al., 2006). GPSR\(^6\) algorithm chooses the closest node to final destination using GPS and leads packets through greedy algorithm. In GPSR, a node transmits a packet to the geographically closest neighbor to destination which is a greedy mode for transmission of packets (Y.-S.C et al., 2010). GyTAR\(^7\) algorithm is an improved version of GPSR algorithm. GyTAR algorithm applies a greedy approach to choose next node. GyTAR assumes that the number of vehicles in every path is determined by road side units and assigns a point to every neighboring connection based on their distance to destination and congestion. This weight can be adjusted by parameters. The result revealed that rate of delivery of GSR is superior to GYTARs (Taleb et al., 2007). GPSRJ+ algorithm applies the right hand rule to choose the best direction. If the furthest node is in routing direction, it will be the best alternative to lead the packet; otherwise the best choice will be intersection node. GPSRJ+ improved delivery rate of packet compared with GPCR\(^8\), and also if there was a failure, it reduced the number of steps by 20% compared with GPSR. GPCR algorithm is a position-based routing protocol working based on a predetermined path using greedy algorithms. This algorithm has been designed to deal with urban challenges. Not only GPCR solves the problem of inaccurate designs, but also improves the function of packet routing in shorter paths (Kevin et al., 2008).

III. Methods

The proposed algorithm combines advantages of single direction and dual direction greedy algorithms and uses path density. In this algorithm, a density is assigned to every node which equals the number of node’s neighbors within its radio range. Increased number of vehicles around a vehicle leads to increased density. Density is calculated by vehicles. In fact, every vehicle transmits a pervasive signal to its radio range and the vehicles within this range receive and return it to source vehicle. The signals are counted by source vehicle. Thus, every vehicle identifies the number of vehicles within its radio range. The proposed algorithm is based on this reason that where density is higher and the direction of vehicles is toward destination the paths are superior (this behavior model works in average mode and to obtain results, more tests are required).

The proposed algorithm relies on assigning scores. The most important parameters that can be used to assign scores are the following:

Density: Node’s density is defined as the number of its neighbors within its radio range. Increased density leads to higher scores (i.e. scores show the priority of a node over others to

\(^6\) Greedy Perimeter Stateless Routing  
\(^7\) Greedy Traffic Aware Routing protocol  
\(^8\) Greedy Perimeter Coordinator Routing
receive packets). The path with higher density is more likely to transmit packets with a slight delay.

Shortest distance to source: Nodes closer to the node containing packets have higher scores and priority.

Moving direction: When the direction of nodes is the same as the node containing packets, they can be assigned higher score and priority; while if their direction is different from the node containing packets, they are assigned lower scores and even minus scores. On doing this, packets are led to correct destination.

Shortest distance to destination: The proposed algorithm assigns higher points and priority to nodes closer to destination. Every node can calculate it’s destination to destination using GPS.

Overall calculation of a node’s weight in the proposed algorithm: Where the effects of parameters priority are indicated by \( W_i \), the priority of node \( N \) is:

\[
P(N) = \sum_{i=1}^{n} W_i
\]

Where \( n \) denotes the number of consider parameters and \( n \) equals 4. Moving direction, density of vehicles, the shortest path to next vehicle and destination vehicle were investigated. \( W_i \) denotes the weight of every parameter. In fact, the weight of every parameter is as follows:

- \( W_1 = \) weight assignment based on density
- \( W_2 = \) weight assignment based on shorter distance to source node
- \( W_3 = \) weight assignment based on shorter distance to destination node
- \( W_4 = \) weight assignment based on direction

To simplify, it can be assumed that the weights are singles. For instance, if moving direction is towards up and right (we assume that the direction of packet destination is the same as this direction), the weight of directions is +1. When moving direction is towards down and left (we assume that the direction of packet destination is different from this direction), the weight of directions is -1. Weights can be assumed single when all weights of distance are considered single (i.e. \( W_2 \) and \( W_3 \) are only 0 and 1; \( W_4 \) is +1 or -1). To calculate \( W_2 \) and \( W_3 \), 1 is assigned to closer one and 0 to the other one. According to the significance of parameters, their weight can steadily be changed on routing. For instance, the value of weight of direction can be first high and then be reduced. \( W_1, W_2, W_3, W_4 \) of nodes are calculated, and the nodes with lower total in equation (1) will be chosen for transmission of packet. Fig.1 illustrates the diagram of the proposed algorithm. As shown in fig.1, the weights of direction are assigned ±1 but can be increased (e.g. ±2). The weights of direction are assigned by the designer of network, but it is suggested to change them dynamically.
IV. Results and Analysis

In this section, implementation and evaluation of 3 greedy algorithms will be discussed. An urban map with some intersections is applied for simulation. To visualize vehicles’ movements graphically, Visual Studio and C# were used. To compare 3 greedy algorithms (one of them is the proposed algorithm), a computer program was produced which received vehicles’ density, vehicles’ velocity, and… as input. Table 1 presents some simulated parameters. These parameters have been changed in different tests to determine the effects of these changes on the number of steps and transmission delay (2 output parameters). The first location and movement of nodes change by Poisson Distribution to approximate the results more accurately.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Simulation area</td>
<td>7km × 9km</td>
</tr>
<tr>
<td>Number of vehicles</td>
<td>10-100</td>
</tr>
<tr>
<td>Number of sent packets</td>
<td>25-30</td>
</tr>
<tr>
<td>Transmission rate</td>
<td>2 packets per ms</td>
</tr>
<tr>
<td>Radio range</td>
<td>500-1000m</td>
</tr>
<tr>
<td>Data packet size</td>
<td>512B</td>
</tr>
<tr>
<td>Transmission delay</td>
<td>1ms</td>
</tr>
<tr>
<td>Physical layer protocol</td>
<td>MAC 802.11 DCF</td>
</tr>
<tr>
<td>Simulation time</td>
<td>5-10 min</td>
</tr>
</tbody>
</table>
The program consists of 2 main parts. One part includes the map of city, roads, and vehicles’ movement and the other part presents speed, velocity, density, etc. Fig.2 illustrates the simulation layout.

![Simulation layout](image)

**Figure. 2.** Simulation layout

One characteristic of the simulator is producing different scenarios for different number of vehicles with different velocity and speed. Yellow node denotes source node. This node is chosen randomly and transmits data packets to destination. In this scenario, destination is chosen randomly. On some streets, the vehicles move towards one direction, and on some streets they move towards 2 directions. This study tests several simulations and investigates the effects of changes of parameters on the number of steps and transmission delay. Fig.3 shows the average time it takes for a packet to traverse the network from its source to destination (end-to-end delay) using every 3 algorithms. Decreased density results in higher slope while increased density leads to a gentler slope, and its value moves towards a constant value. According to the diagram, the function of the proposed algorithm was superior to single direction and dual direction greedy algorithms under lower and medium density. Increased density of nodes leads to lower end-to-end delay and under a density ≥40 it moves towards a constant value.
Fig. 3 shows that under lower density, the average transmission time of the proposed algorithm is lower compared with other algorithms. In fact, increased speed leads to lower average transmission time from source to destination. The proposed algorithm includes fewer steps compared with the 2 other algorithms.

Fig. 4 shows that under high density the proposed algorithm includes fewer steps. Thereby, routing overhead (average step numbers required to transmit a packet from its source to
destination) is lower compared with the 2 other algorithms. High density results in increased number of steps of single and dual greedy algorithms, but for the proposed algorithm the procedure is descending.

![Graph showing the relationship between number of steps and density of 3 algorithms]

Figure 5. The relationship between number of steps and density of 3 algorithms

V. Conclusions

In this study, we investigated routing in VANETs and discussed different routing algorithms. We introduced a greedy algorithm which works based on weight assigning, prioritizing the parameters such as vehicle moving direction, the shortest path to next vehicle and destination vehicle. Several simulations showed that the proposed algorithm has a superior function in terms of end-to-end delay (average time it takes for a packet to traverse the network from its source to destination) and routing overhead (average step numbers required to transmit a packet from its source to destination) under different densities, especially under medium and low density. Increased speed and density lead to lower average transmission time between source and destination. In future studies, we are going to improve the function of the proposed algorithm using behaviors of particles swarm intelligence.

References


xi.


47