Directed Information Dissemination in Vehicular Ad-hoc Networks[1]

A.K.M. Mahtab Hossain[*2] · **Preechai Mekbungwan**[3] ·

Kanchana Kanchanasut[4]

Abstract In this article, we utilize the idea of multipoint relays (MPRs) found in literature [2] to propagate accident information in a restricted way (e.g., only backwards). We devise an algorithm to identify MPRs that are geographically situated behind a particular node using only its neighbor table, and speed information of the neighboring vehicles. With the identification of the backward MPRs, it is possible to restrict the information dissemination to vehicles behind a particular Vehicular Ad-hoc Network (VANET) node only. This might benefit the approaching vehicles so that the driver could take preventive measures in real-time since he/she will have an indication of the severity of road conditions ahead. We assume that there exists an inter vehicular network using Optimized Link State Routing (OLSR) where accident information can be propagated to all nodes using on-going OLSR control packets. We envision our application will run on top of existing routing protocols (e.g., OLSR), thereby resulting in very little integration effort, and retaining OLSR's reduced network traffic advantage through the use of MPRs. We analyze our back MPR identification algorithm in a detailed manner. We also show that by using our approach the location of the accident alert instigator node could be pinpointed if a subset of the nodes in the same VANET know their geographical positions. We use VANET mobility models generated by SUMO into NS-3 for our simulations, and also perform preliminary experiments to verify the algorithm's effectiveness. Our analysis and experiments show favorable results.

[*]Corresponding Author – this work was done while he was working at intERLab, Asian Institute of Technology, Thailand.
[1]This article is an extended version of [1].
[2]Department of Computer Science, University College Cork, Ireland, E-mail: m.hossain@cs.ucc.ie
[3]intERLab, Asian Institute of Technology, Thailand, E-mail: preechaim@ait.ac.th
[4]intERLab, Asian Institute of Technology, Thailand, E-mail: kanchana@ait.ac.th

Keywords

Vehicular Ad-hoc Network (VANET), Multi-Point Relay (MPR), Traffic Information Propagation, Optimized Link State Routing (OLSR), Driver Assistance.

1 Introduction

The goal of Vehicular Ad-hoc Network (VANET) research is to develop a vehicular communication system to enable real-time and cost-effective distribution of data for the benefit of passengers' safety or comfort [3]. VANETs are generally characterized by highly dynamic topology, frequently disconnected networks, varying communication environments (e.g., highways and city center roads), and hard delay constraints [4]. The adaptability of a proactive routing protocol such as Optimized Link State Routing (OLSR) to the challenging VANET environment has been supported by certain research works [5, 6]. Our research is focused on disseminating traffic information such as safety warning while taking into consideration the inherent characteristics of VANETs on top of OLSR. More specifically, we concentrate on propagating traffic information in a restricted way (e.g., only backwards) to alleviate severe road conditions. For example, if an accident warning can be sent only to approaching vehicles, it would allow the drivers to take preventive measures which in turn would reduce further accidental situations and traffic jams.

In [1], we focus on directed traffic information propagation in order to avoid network congestion and delay. We utilize the idea of multipoint relays (MPRs) found in literature [2] to propagate accident information in a restricted way. In the OLSR protocol, the MPR set of a particular node is calculated such that it contains a subset of its one hop neighbors which covers all of its two hop neighbors. In other words, each node in the network selects a set of nodes in the neighborhood independently, which retransmits its packets. This set of selected neighbor nodes is called the multipoint relays (MPRs) of that node. Therefore, an OLSR route consists of a sequence of hops through the MPRs from source to destination. MPR's utility can be realized through reduced the control message overhead needed to generate routes for OLSR. An ad-hoc node's control message is only broadcasted by its selected MPRs. All other nodes do not retransmit the message, thereby reducing network traffic [2]. Our directed traffic information propagation solution retains the lower network traffic advantage through the use of MPRs. Moreover, the scheme utilizes a subset of a node's MPRs that are geographically behind a particular node. Note that, we do not assume any knowledge of location information (e.g., provided by GPS) for the VANET nodes which makes it particularly applicable to urban environment where GPS may be unable to provide finer position estimates. Our algorithm only utilizes the network layer

information and relative speed of a VANET node to identify the MPRs that are backwards in the following manner:

– Firstly, each node detects the hop distance changing events for each of its neighbors, i.e, a neighbor becoming its 2-hop neighbor or vice versa (i.e., a 2-hop neighbor becoming its immediate neighbor). Both these events could be realized by just inspecting each individual node's own OLSR neighbor tables.
– Secondly, the speed and direction information of each vehicle could be obtained from its on-board diagnostics system. This information can then easily be incorporated inside the HELLO messages of OLSR to be exchanged among the neighbors and is utilized to infer the relative distance travelled between the hop distance changing events, mentioned in the first bullet.

For example, suppose the detected event is a neighbor changing to a 2-hop neighbor, and during that interval, the node gains distance on both the particular neighbor and the MPR through which the 2-hop neighbor is reached, then the MPR is most probably behind the node.

We envision our application to run on top of existing routing protocols (e.g., OLSR), thereby resulting in very little integration effort. A node would select only its back MPRs to propagate the accident information or any other emergency traffic information. As a result, i) only the approaching vehicles will be notified so that the driver could take preventive measures in real-time or ii) inform a control center about the severity of road conditions through the use of communication infrastructure like 3G, should any of the vehicles inside the ad-hoc network contain that additional functionality.

Furthermore, we also show that if a number of VANET nodes know their locations, the traffic location of the incident (e.g., accident location) could be pinpointed. This assumption of a few nodes knowing their locations in an inexpensive way without the need of GPS is quite reasonable since MPRs have been utilized to estimate location of a node in MANET [7]. In [8], the authors propose OLSR-L which enables simultaneous routing and localization in wireless multi-hop networks. This traffic location could then be notified to an infrastructure by a VANET node by some other communication means (e.g., 3G) which would enable this traffic information to be propagated throughout the whole city's network. We chose vehicular ad-hoc networks (VANETs) for our algorithm's evaluation purpose where directed information dissemination has obvious applications. In general, our algorithm could work with other ad-hoc networks as well (not only VANETs) using OLSR (or any other routing protocol that keeps neighbor table information) where directed information dissemination might be required by the applications.

In this article, we devise an algorithm to identify MPRs that are geographically situated behind a particular node using only OLSR neighbor table and speed information of the vehicles. We analyze our back MPR identification algorithm in a detailed way in this "Wireless Networks" journal submission compared to our previous paper. We prove its correctness using graph theory, and also provide both simulation and experimental results to verify the algorithm's effectiveness. In [1], only simulation results were presented. We assume the communication environment of the vehicles could be modeled with the help of unit disk graphs [9]. In the field of wireless networking, the multi-hop radio network is often modeled with the help of unit disk graphs, especially for analysis purpose. In this model, the nodes are assumed to be scattered over a Euclidean plane having identical (unit) communication range. The vertices $u, v \in V$ of the graph, $G = (V, E)$ represent the wireless nodes and an edge $(u, v) \in E$ between the two vertices exists if the Euclidean distance between them is at most 1. In other words, the nodes can communicate only if they are within mutual communication range.

Due to the frequently disconnected networks characteristics of VANET [4], we envision two different cases for propagating information: i) VANET may be severely disconnected such that it may only consist of 1-hop neighbors, and ii) A multi-hop network may indeed exist as shown in [5] running a MANET routing protocol (i.e., OLSR). Note that we address the second category above, where we deal with the issue of directed traffic information propagation without incurring additional overhead. The reduction of network traffic (i.e., overhead) is achieved by the choice of MPRs to disseminate traffic information rather than approaches like flooding or gossiping [2]. When the network only consists of 1-hop neighbors, we assume the nodes which are located behind or ahead of a particular vehicle could be differentiated using some other techniques. For example, each node can record a landmark's ID (emitted by a roadside landmark itself) and its passing timestamp throughout its journey. By comparing the passing timestamps of a particular landmark, the front or rear vehicles could be identified. However, this information may not be totally correct at a particular instance, as in case of varying speeds of the vehicles, and the absence of landmarks in a vehicle's path for longer duration. We term this information as *a priori* knowledge, and assume it to be used only for disconnected networks where the efficiency of our algorithm is reduced.

The rest of the paper is organized as follows. We provide a brief description of related works in Section 2. In Section 3, we discuss the methodology of the back MPRs identification idea in detail, which is the cornerstone of our directed traffic information propagation algorithm. We analyze our algorithm's effectiveness using graph theory in Section 4. We point out some practical considerations for implementing the algorithm in

Section 5. We present the simulation results, and preliminary experimental findings supporting our claims in Section 6, and 7, respectively. Finally, we depict in Section 8 the conclusions drawn, and our future work.

2 Related Work

For a conventional centralized traffic management system, traffic is detected and measured automatically by using sensors (e.g., cameras) deployed along the roads. The collected information is transmitted to the control center where it is processed and analyzed. The resulting data is then forwarded to an appropriate radio broadcast station, which eventually distributes the traffic information to vehicles. Because of the inherent delay of such centralized system (e.g., STEID [10]) to spread the traffic information to the vehicles, this type of system may not be attractive for our application. Moreover, the system depends on fixed infrastructure where a large number of sensors are needed to be deployed. The system requires vast investment to actually build the infrastructure and cover its maintenance which might be a problem. Our system aims to disseminate traffic information in VANET where there is no infrastructure or sophisticated positioning service like GPS is available.

An alternative decentralized approach based on Vehicle-to-Vehicle (V2V) communications is attracting increasing interest, with some protocols already proposed by which a traffic information system is constructed [11]. We also focus on such decentralized approaches. Ziliaskopoulos and Zhang propose a traffic sensing system based on opposing vehicle communications [12]. Vehicles exchange their travel times in road segments (in and out times) with neighbors moving in the opposite direction. The information can be utilized to help drivers judge whether the road they are approaching is congested. Wischhof *et al.* propose SOTIS [13] within the FleetNet [14] project. In SOTIS, vehicles monitor and analyze data individually using the periodically exchanged traffic data by themselves. The results are then transferred to all surrounding neighbors. Goel *et al.* [15] examine the characteristics of V2V communications for traffic information system, and their simulations show that Wi-Fi based V2V network can support effective traffic information dissemination. Zeng *et al.* [16] present a directional routing and scheduling scheme (DRSS) in vehicular delay tolerant network (DTN) to optimize the energy efficiency using Nash Q-learning approach by taking into account congestion, buffer capacity, and delay constraints. Zhang *et al.* [17] study the information propagation process in a 1-D mobile ad hoc network formed by vehicles, and derive analytical formulas for the fundamental properties of the information propagation process and the information propagation speed. Xiaomin *et al.* [18] propose and justify a distributive cross-layer scheme for the design of the control channel in DSRC with three levels of broadcast services that are critical to most potential vehicle safety related applications. Jerbi *et al.* [19] pro-

pose an infrastructure-free mechanism for road density estimation. Ghafoor *et al.* [20] propose a street-aware Intelligent Beaconless (IB) geographical forwarding protocol to determine the relay node by using modified 802.11 MAC layer's Request To Send (RTS) / Clear To Send (CTS) frames and geographical street map information. Lo and Tsai [21] propose an event-based reputation system (ERS) of traffic events for a traffic information system to prevent the spread of false alarms. To the best of our knowledge, no work has delved into using the network layer information to propagate road conditions. We propose the traffic information to be propagated in a restricted way using multipoint relays (MPRs). Finding the optimal MPR set has been proved to be an NP-complete problem [22]. It is shown in [2] that the MPR set need not be optimal, however, it should be small enough in order to realize the effectiveness of the routing protocol in terms of control message retransmission overhead. Some of the MPR selection algorithms are discussed in [22, 23].

3 Identification of MPRs that are behind a Node

3.1 Background

The idea of multipoint relays (MPRs) was introduced in OLSR [2] to minimize the flooding of broadcast packets in the network. This is achieved by reducing the duplicate retransmissions in the same region. Only a node's selected MPRs retransmit its packets. Each node in the network selects a set of nodes in the neighborhood independently, which retransmits its packets. This set of selected neighbor nodes is called the multipoint relays (MPRs) of that node. Any other neighbor that is not selected as an MPR by the node, reads and processes the packet but does not retransmit it. Consequently, the network traffic is reduced. The MPR set is calculated in a manner to contain a subset of one hop neighbors which covers all the two hop neighbors, i.e., the union of neighbor sets of all MPRs contains the entire two hop neighbor set (see Fig 1). Our goal is to select the MPRs that are geographically behind the node in order to propagate the traffic information only in a backwards direction.

Fig. 1: Selection of MPRs of Node k. The thicker lines represent the retransmission paths from Node k.

3.2 Problem Statement

Suppose the MPR set of a node k at time instance t is denoted by, $\mathcal{M}_k^t = \{1_k^t, 2_k^t, \ldots, m_k^t\}$, where $|\mathcal{M}_k^t| = m_k^t$. Our goal is to identify $\mathcal{B}_k^t \subset \mathcal{M}_k^t$ where $\mathcal{B}_k^t$ denotes the MPR set that is geographically behind node k at time t.

3.3 Description

First, we list the simplified assumptions on which our algorithm is based upon – (i) all the vehicles are moving in the same direction, i.e., we only consider vehicles in a VANET that have the same direction (see Fig. 2), and (ii) the vehicles could be modeled with the help of unit disk graphs [9].

In the field of wireless networking, the multi-hop radio network is often modeled with the help of unit disk graphs. In this model, the nodes are assumed to be scattered over a Euclidean plane having identical (unit) communication range. The vertices $u, v \in V$ of the graph, $G = (V, E)$ represent the wireless nodes, and an edge $(u, v) \in E$ between the two vertices exists if the Euclidean distance between them is at most 1. Clearly, the unit disk graph model neatly captures this behavior, and it has become a de-facto standard when studying ad-hoc and sensor networks, especially for analysis purpose. We also utilize the unit disk graph model for the VANET nodes of our analysis. In other words, we assume the communication range of each vehicle to be equal (e.g., 802.11b).

Fig. 2: Our algorithm is only targeted at the vehicles moving in the same direction. (a) single lane road. and (b) double lane road.

Assume $\mathcal{F}_k^t$ denotes the MPR set that is geographically ahead of node k, and $\mathcal{U}_k^t$ specifies the MPR set that could not be identified as either behind or ahead of node k at time t. Consequently, we have

- $\mathcal{B}_t^k \cup \mathcal{F}_t^t \cup \mathcal{U}_k^t = \mathcal{M}_k^t$

- $\mathcal{B}_t^k \cap \mathcal{F}_t^t = \emptyset$, $\mathcal{B}_t^k \cap \mathcal{U}_t^t = \emptyset$, and $\mathcal{F}_t^k \cap \mathcal{U}_t^t = \emptyset$.

Let us inspect if the node, $i \in \mathcal{M}_k^t$ is behind node k. Suppose the neighbor set of k at time t is denoted by, $\mathcal{N}_k^t = \{1_k^t, 2_k^t, \ldots, n_k^t\}$, where $|\mathcal{N}_k^t| = n_k^t$, and $m_k^t \leq n_k^t$. Similarly, the neighbor set of node i can be denoted as, $\mathcal{N}_i^t = \{1_i^t, 2_i^t, \ldots, n_i^t\}$, where $|\mathcal{N}_i^t| = n_i^t$. Two events trigger our algorithm of deciding whether $i \in \mathcal{M}_k^t$ is behind the node k:

- Any node, $j \in \mathcal{N}_k^{t-\delta}$ which is a neighbor of node k at time $(t-\delta)$ becomes a 2-hop neighbor of k through MPR i at time t, $j \in (\mathcal{N}_i^t \setminus \mathcal{N}_k^t)$. Let us denote this incident as, $j \in N_k^{t-1} \to j \in (\mathcal{N}_i^t \setminus \mathcal{N}_k^t)$. Here, δ is the time elapsed between j first becoming a neighbor of node k, and then changing to 2-hop neighbor through MPR i.

- Any node, $j \in (\mathcal{N}_i^{t-\delta} \setminus \mathcal{N}_k^{t-\delta})$ which is a 2-hop neighbor of node k at time $(t - \delta)$ becomes a neighbor of k through MPR i at time t, $j \in \mathcal{N}_k^t$. Let us denote this incident as, $j \in (\mathcal{N}_i^{t-\delta} \setminus \mathcal{N}_k^{t-\delta}) \to j \in \mathcal{N}_k^t$.

Here, δ is the time elapsed between j first becoming a 2-hop neighbor of node k through MPR i, and then changing to 1-hop neighbor.

When the two events occur, our algorithm makes use of the relative distance between two VANET nodes in order to infer whether an MPR is behind the node. Suppose r_{ki} and r_{kj} represents the relative distance of the node k from MPR i and neighbor j, respectively during time period δ. $r_{ki} > 0$ means node k is gaining distance on node i, and $r_{ki} < 0$ means the opposite. The relative distance calculation between two VANET nodes is shown in detail in Section 3.3.1.

Now let us proceed to discuss the back MPR decision making criteria. Let us consider the first case (Fig. 3). If $r_{ki}, r_{kj} > 0$, we can safely deduce that MPR i is behind node k. $r_{ki}, r_{kj} > 0$ means node k gained distance on both the MPR i and neighbor j during the time period δ. If MPR i were ahead of node k then j could not be a 2-hop neighbor of node k through i at time t since $r_{kj}, r_{ki} > 0$.

Fig. 3: 1-hop neighbor j of node k becomes 2-hop neighbor through MPR i.

Fig. 4: 2-hop neighbor j through MPR i of node k becomes 1-hop neighbor.

Now consider the 2$^{\text{nd}}$ case (Fig. 4). If $r_{ki} > 0$, and $r_{kj} < 0$, we can safely deduce that, MPR i is behind node k. If MPR i were ahead of node k then 2-hop neighbor at time $t - \delta$ could not become 1-hop at time t since $r_{ki} > 0$, and $r_{kj} < 0$.

We summarize the back MPR identification in algorithm 1. Notice that, our algorithm is generally invoked when the topology around the node goes through changes. In case of accidents, traffic jam, etc., the VANET topology is likely to change. Therefore, our algorithm is expected to successfully identify the back MPRs, thereby, meeting our goal of directed propagation of the information. However, if the VANET topology is *relatively* static, then the algorithm might not be invoked at all. As a result, some MPRs' direction (i.e., forward or backward) might be "undetected". We investigate these scenarios in our experiments in Section 6. Apart from the "undetected" case, our algorithm may cause two types of errors in terms of identifying the MPR's relative direction w.r.t. node: i) *False negative:* identifying a node's MPR to be ahead of it while in the actual scenario, it is behind the node, and ii) *False positive:* the MPR is identified as backwards while it is actually ahead of the node.

False positives may compel the traffic information to propagate in the wrong direction. Therefore, this type of error is crucial for the performance of our algorithm. On the other hand, false negatives may not

---

Algorithm 1 Identification of back MPR

We record the time instance when j becomes a 1-hop neighbor or a 2-hop neighbor through MPR i for node k for the first time. Suppose δ time period has been elapsed when the same node j becomes a 2-hop neighbor (before δ time period, it was a 1-hop neighbor) or it becomes a 1-hop neighbor of node k (previously 2-hop neighbor through MPR i). Suppose r_{ki} and r_{kj} represents the relative distance of the node k from MPR i and neighbor j, respectively during time period δ.

1: **if** 1-hop neighbor j becomes 2-hop neighbor after δ time period **then**
2:    **if** $r_{ki}, r_{kj} > 0$ **then**
3:       MPR i is behind node k
4:    **else if** $r_{ki}, r_{kj} < 0$ **then**
5:       MPR i is ahead of node k
6:    **end if**
7: **else if** 2-hop neighbor j becomes 1-hop neighbor after δ time period **then**
8:    **if** $r_{ki} > 0$ and $r_{kj} < 0$ **then**
9:       MPR i is behind node k
10:    **else if** $r_{ki} < 0$ and $r_{kj} > 0$ **then**
11:       MPR i is ahead of node k
12:    **end if**
13: **end if**

---

be so detrimental for our cause since they will not propagate the information in the wrong direction. Both types of errors are investigated in the experiments of Section 6.

3.3.1 Relative Distance Calculation between two VANET nodes

As mentioned previously, we also require to measure the relative distance between node k and MPR i for our back MPR identification algorithm. The relative distance between node k and neighbor j is also used. For brevity, we only discuss the first scenario. The other relative distance calculation between node k and neighbor j follows exactly the same procedure.

Assume v_k^t and v_i^t denote the speeds of node k and MPR i, respectively, at time t. Our goal is to find the relative distance between k and i after some time instance, δ. If we could measure the speeds of both nodes at every second, the relative distance between nodes k and i every one-second interval could be calculated as, $r_{ki}^{t+1} = \frac{1}{2}\{(v_k^{t+1} + v_k^t) - (v_i^{t+1} + v_i^t)\}$. Consequently, the relative distance between k and i after δ time instance becomes,

$$r_{ki} = r_{ki}^{t+1} + r_{ki}^{t+2} + \ldots + r_{ki}^{t+\delta} \tag{1}$$

Based on the relative distance, three different conclusions could be made on the relative positions of k and i: I) $r_{ki} < 0$, MPR i gains distance on node k, or II) $r_{ki} > 0$, node k gains distance on MPR i, or III) $r_{ki} = 0$, the relative distance remains the same. A pictorial description of the three conditions could be found in Fig. 5.

For our algorithm to work, we first record any time instance, t when (i) a node j becomes a neighbor of k, or (ii) a node j becomes a 2-hop neighbor of k through MPR i. Suppose at time instance $t+\delta$, the same node j of (i) becomes a 2-hop neighbor of k through MPR i, or the same node j of (ii) becomes a neighbor of k.

Fig. 5: Three different relative distance scenarios calculated from vehicles' speeds.

Subsequently, the relative distance, r_{ki} between node k and MPR i during time interval δ is calculated using (1). Note that, we assume the vehicles exchange their speed information every second to compute the relative distance. This could be done by incorporating it inside the HELLO message which is exchanged among the nodes to discover symmetric neighbors in OLSR protocol. However, this assumption may not scale well where the HELLO message interval is generally set to two seconds. One way to get around this problem could be to reduce the HELLO message interval or to store speed information at each second within the node, and send all information in a condensed format inside the HELLO message. Loss of every second's speed information may have adverse effect on the computation of Eq. (1). For example, suppose the speeds of node k and MPR i during a three-second interval are given as,

Node	Speed (m/s)		
i	2	3	4
k	2	6	4

If we lose the speed information of the 2nd second, then we are left to believe, $r_{ki} = 0$, i.e., the vehicles' relative positions remained the same. However, $r_{ki} = 9m - 6m = 3m > 0$ which indicates node k actually gained distance on node i during this three second interval.

4 Analysis using Graph Theory

Assume the vertices $k, i \in V$ of the graph, $G = (V, E)$ represent the vehicular wireless nodes creating a 1-D topology, and an edge $(k, i) \in E$ between the two vertices exists if they are within mutual communication range. Let $G^t = (V^t, E^t)$, and $G^{t-\delta} = (V^{t-\delta}, E^{t-\delta})$ denote the graphs comprising of the VANET nodes at time instance t and $t - \delta$, and $\mathcal{M}_k^t \subset V^t$ and $\mathcal{M}_k^{t-\delta} \subset V^{t-\delta}$ denote the MPR set of node k at time t and $t - \delta$, respectively. The two events which trigger the algorithm 1 to decide whether $i \in \mathcal{M}_k^t$ is behind node k can be represented as:

- For any node $j \in V_k^{t-\delta}, V_k^t$, if $(k, j) \in E^{t-\delta}$, and $(k, j) \notin E^t$, $(i, j) \in E^t$, then the event corresponds to the incident where a neighbor of node k at time $(t - \delta)$ becomes a 2-hop neighbor via MPR i at time t (see Fig. 6a).

- For any node $j \in V_k^{t-\delta}, V_k^t$, if $(k, j) \notin E^{t-\delta}$, $(i, j) \in E^{t-\delta}$, and $(k, j) \in E^t$, then the event corresponds to the incident where a 2-hop neighbor of node k via MPR i at time $(t - \delta)$ becomes a neighbor at time t (see Fig. 6b).

Here, $V_k^{t-\delta}, V_k^t$ denote the vertex set of the connected graph rooted at node k at time $t-\delta$ and t respectively.

Fig. 6: Simple graph representation of VANET nodes to represent the neighbor change events. (a) 1-hop neighbor j of node k becomes 2-hop neighbor through MPR i. (b) 2-hop neighbor j through MPR i of node k becomes 1-hop neighbor.

4.1 Analytical Motivation

In this section, we provide the formal proofs that have motivated our back MPR identification algorithm.

4.1.1 The incident, for any node $j \in V_k^{t-\delta}, V_k^t$, $(k,j) \in E^{t-\delta}$, and $(k,j) \notin E^t$, $(i,j) \in E^t$

Theorem 1 *If $i \in \mathcal{M}_k^{t-\delta}, \mathcal{M}_k^t$, and $(k,j) \in E^{t-\delta}$, $(k,j) \notin E^t$, $(i,j) \in E^t$, and $r_{ki}, r_{kj} > 0$ then MPR i is behind node k.*

Proof Suppose MPR i is not behind k. Consequently, node j is not behind k either since $(k,j) \notin E^t$, and $(i,j) \in E^t$. This inference could be made since the communication range of all the nodes (including k and MPR i) is the same. At time $(t-\delta)$, MPR i and neighbor j would have been an additional r_{ki} and r_{kj} distance ahead of k, respectively where δ denotes the time elapsed between the incidents, $(k,j) \in E^{t-\delta}$, and $(k,j) \notin E^t$, $(i,j) \in E^t$.

The node j that has become a 2-hop neighbor of k through MPR i could not have been a neighbor of k at time $(t-\delta)$ since $r_{kj} > 0$, and thus j is ahead of k. In other words, if $j \in E^{t-\delta}$, then it would be $(k,j) \in E^t$ as well, because node k will be relatively closer to j at time t than at time $(t-\delta)$. However, $(k,j) \notin E^t$. Therefore, j is behind node k which subsequently yields $i \in \mathcal{B}_k^t$.

Theorem 2 *If $i \in \mathcal{M}_k^{t-\delta}, \mathcal{M}_k^t$, and $(k,j) \in E^{t-\delta}$, $(k,j) \notin E^t$, $(i,j) \in E^t$, and $r_{ki}, r_{kj} < 0$ then MPR i is not behind node k.*

Proof Suppose MPR i is behind k. Consequently, node j is also behind k since $(k,j) \notin E^t$, $(i,j) \in E^t$, and the communication range of all the nodes (including k and MPR i) is the same. At time $(t-\delta)$, MPR i and node j would have been an additional r_{ki} and r_{kj} distance behind k, respectively.

The node j that has become a 2-hop neighbor of k through MPR i could not have been a neighbor of k at time $(t-\delta)$ since $r_{kj} < 0$, and thus j is behind k. In other words, if $(k,j) \in E^{t-\delta}$, then it would be $(k,j) \in E^t$ as well, because node k will be relatively closer to j at time t than at time $(t-\delta)$. However, $(k,j) \notin E^t$. Therefore, j is not behind node k which subsequently yields $i \notin \mathcal{B}_k^t$.

4.1.2 The incident, for any node $j \in V_k^{t-\delta}, V_k^t$, $(k,j) \notin E^{t-\delta}$, $(i,j) \in E^{t-\delta}$ *and* $(k,j) \in E^t$

Theorem 3 *If* $i \in \mathcal{M}_k^{t-\delta}, \mathcal{M}_k^t$, *and* $(k,j) \notin E^{t-\delta}$, $(i,j) \in E^{t-\delta}$, $(k,j) \in E^t$, *and* $r_{ki} > 0 > r_{kj}$, *then MPR* i *is behind node* k.

Proof Suppose MPR i is not behind k. At time $(t-\delta)$, MPR i would have been an additional r_{ki} distance ahead of k. Consequently, neighbor j must have been ahead of MPR i, thereby ahead of k too at time $(t-\delta)$ since $(k,j) \notin E^{t-\delta}$ and $(i,j) \in E^{t-\delta}$.

If $(k,j) \notin E^{t-\delta}$ then it would be $(k,j) \notin E^t$ as well since j will be further away from node k at time t than at time $(t-\delta)$. This is because node j was ahead of k at time $(t-\delta)$, and $r_{kj} < 0$. However, $(k,j) \in E^t$. Therefore, $i \in \mathcal{B}_k^t$.

Theorem 4 *If* $i \in \mathcal{M}_k^{t-\delta}, \mathcal{M}_k^t$, *and* $(k,j) \notin E^{t-\delta}$, $(i,j) \in E^{t-\delta}$, $(k,j) \in E^t$, *and* $r_{ki} < 0 < r_{kj}$ *then MPR* i *is not behind node* k.

Proof Suppose MPR i is behind k. At time $(t-\delta)$, MPR i would have been an additional r_{ki} distance behind k. Consequently, neighbor j must have been behind MPR i, thereby behind k too at time $(t-\delta)$ since $(k,j) \notin E^{t-\delta}$ and $(i,j) \in E^{t-\delta}$.

If $(k,j) \notin E^{t-\delta}$ then it would be $(k,j) \notin E^t$ as well since j will be further away from node k at time t than at time $(t-\delta)$. This is because node j was behind k at time $(t-\delta)$, and $r_{kj} > 0$. However, $(k,j) \in E^t$. Therefore, $i \notin \mathcal{B}_k^t$.

4.2 Algorithm

Algorithm 1 lists the pseudo-code for identifying back MPR of a node k after the two events (2-hop neighbor becoming 1-hop or vice versa) have already been detected. In this section, we provide the algorithm for detecting these two events using the graph representation of VANET. We assume adjacency list representation of graphs [24] where the list of a vertex incorporates all other vertices to which it has an edge. It can be thought as an array indexed by the vertex number that points to a singly linked list of neighbors of the vertex. We also assume a *boolean* data-type within the list to keep track of whether a particular neighbor is an MPR or not.

All the pseudo-codes listed in algorithm 1, 2 and 3 are parts of the same algorithm which identify the back MPR of a particular node. Algorithms 2 and 3 that detect the two events of neighbor table changes are

---

Algorithm 2 Detection of 2-hop neighbor becoming 1-hop event

Suppose E_k^t denotes edge-set of a node k at time t where $E_k^t \subset E^t$.

1: **for** each edge $(k, i) \in E_k^t$ **do**
2:   **if** i is an MPR of k **then**
3:     compute $W = \{j\}$ where $\forall_j[(i, j) \in \{E_i^{t-\delta} \setminus (E_i^{t-\delta} \cap E_k^{t-\delta})\}]$
4:     **for** each $j \in W$ **do**
5:       **if** $(k, j) \in E_k^t$ **then**
6:         2-hop neighbor j at time $(t - \delta)$ becomes a neighbor at time t
7:       **end if**
8:     **end for**
9:   **end if**
10: **end for**

Algorithm 3 Detection of 1-hop neighbor becoming 2-hop event

Suppose E_k^t denotes edge-set of a node k at time t where $E_k^t \subset E^t$.

1: **for** each edge $(k, i) \in E_k^t$ **do**
2:   **if** i is an MPR of k **then**
3:     compute $W = \{j\}$ where $\forall_j[(i, j) \in \{E_i^t \setminus (E_i^t \cap E_k^t)\}]$
4:     **for** each $j \in W$ **do**
5:       **if** $(k, j) \in E_k^{t-\delta}$ **then**
6:         1-hop neighbor j at time $(t - \delta)$ becomes 2-hop neighbor via MPR i at time t
7:       **end if**
8:     **end for**
9:   **end if**
10: **end for**

---

actually referred inside algorithm 1 (at line 1 and 7, respectively) which decides whether an MPR is back or not. They have been listed here separately just for clear illustration purpose.

4.2.1 Complexity Analysis

The following analysis provides an upper bound on the computational costs for running the complete back MPR identification algorithm. The complexity primarily depends on lines 2 and 4 of algorithm 2 or 3. Searching all the edges of node k (line 2) costs $\mathcal{O}(|E_k^t|)$, while line 4 also costs $\mathcal{O}(|E_k^t|)$. After detection of any of the two events by the algorithms, the comparison to identify back MPRs (see algorithm 1) only costs $\mathcal{O}(1)$. Therefore, the complexity of our algorithm can be denoted as, $\mathcal{O}(|E_k^t|^2)$. This running time corresponds to a single node's identification of its back MPRs using only its 1-hop and 2-hop neighbor information. When a node does not have an MPR which is observed to be quite a common scenario for highly mobile VANET, the complexity reduces to $\mathcal{O}(|E_k^t|)$. For a VANET node disconnected from the rest of the network, the identification algorithm's complexity reduces to $\mathcal{O}(1)$.

If the knowledge of the complete graph (i.e., the whole VANET) is available then line 2 of algorithm 2 and 3 could be modified to identify the back MPRs for all the nodes. The complexity of this algorithm can then be denoted as, $\mathcal{O}(|E|^2)$. It can be easily seen that this algorithm will be particularly useful for sparse

graphs (i.e., graphs with relatively fewer edges) which is generally the case for frequently disconnected VANETs.

5 Practical Considerations

The network layer parameters, e.g., 1-hop or 2-hop neighbors that are used in our algorithm are easily available from OLSR API. The traffic/accident information can be propagated by embedding it inside OLSR's TC (topology control) message [2] so that only back MPRs retransmit the information (not all of its MPRs).

In this article, we address only the one-way traffic for simplicity (without loss of generality) taking into account the application requirement that may use our algorithm. For example, in case of traffic jam information propagation, only the approaching vehicles that might potentially be blocked will be interested in this information. Hence, forwarding the traffic information only backwards to the same direction vehicles would suffice. However, our algorithm can be adapted to include opposite direction vehicles given the additional direction information per node. It can easily be shown that only the 1-to-2 hop event (not the 2-to-1 hop event) could be utilized to identify the backward MPR for the vehicles moving in opposite direction (see Fig. 7a). Correspondingly, 2-to-1 hop incident may only be indicative of a front MPR as shown in Fig. 7b. Similar analysis as Section 4.1 can be carried out to prove the correctness of this detection procedure. Unlike the same direction scenarios where there are four cases (discussed in Theorem 1, 2, 3 and 4), only two cases arise for opposite direction vehicles as depicted in Fig. 7a and 7b, respectively. Note that unlike the same direction case, the vehicles' speed information are not necessary here.

Fig. 7: MPR identification scenarios using opposite direction vehicle. (a) 1-to-2 hop scenario: MPR's position can be detected as back. (b) 2-to-1 hop scenario: MPR's position may be detected as front.

In Section 1, we also discussed that a few of the VANET nodes may obtain location information inexpensively [7, 8]. If that is the case, then our algorithm could also help pinpointing the accident/traffic jam location. Let us assume all the vehicles came to halt during those situations, and lined up in a queue one after another. The traffic information is propagating backwards using our algorithm. If this information reaches one of the vehicles with known location (x) after h hops, the originating location of accident/traffic congestion may be approximated as, $(x + h \times d)$ where d denotes the separation between two halted vehicles during accident/traffic jam. So rather than notifying the vehicle's own location as the accident/traffic jam location, the refined location of the originator could be reported to the infrastructure, and thereby, spread over the whole city quickly.

6 Simulation Results

For simulation, we used SUMO [25] to generate a simple urban layout of roads which consist of single or double lanes (Fig. 2). SUMO takes care of the routing of traffic, and generates vehicle traces for our experiments. We then convert the SUMO trace output to the format of NS-3 simulator's [26] mobility model. The OLSR routing daemon implementation of NS-3 is used for the network layer of vehicles. The HELLO and TC message interval are kept as 2 and 5 sec., respectively according to OLSR's standard setting. 802.11b module of NS-3 has been selected as the underlying wireless technology.

6.1 Single lane experiments

In this experiment, we consider a single lane consisting of 30 vehicles moving in the same direction (e.g., Fig. 2a). Fig 8 is constructed by taking snapshots of back MPRs at various time intervals for a randomly chosen vehicle where the maximum speed has been set to 80 km/hr. All other vehicle traces with different speeds also show similar trend. Note the absence of MPRs at many time instances in Fig 8. This implies the intermittent connectivity of the network which is one of the main challenge faced by VANET. In such cases, our algorithm can not be applied as outlined previously since there is no MPRs. The traffic information dissemination problem is thereby reduced to distributing only to the 1-hop neighbors that are backwards. We discussed the landmark approach in Section 1 for such scenarios, and also discussed how our relative distance idea could enhance its accuracy in Section 5. Additionally, the two incidents discussed in Section 4.1.1 and 4.1.2 may not occur at all, thereby, leaving an MPR to be "undetected". This situation occurs when the network is relatively static as observed in our simulation scenarios (some time instances are evident in Fig 8 also). The following section studies the effectiveness of the back MPR identification algorithm in depth using the same experimental setup. In Section 6.2, we further investigate our algorithm's performance in accidental or traffic jam scenarios which might be of practical importance.

Fig. 8: Snapshots of identified back MPRs of a node for various time intervals.

6.1.1 Analysis of Our Algorithm's Performance

To analyze the back MPR identification algorithm's effectiveness, for an individual trial, we run our algorithm for every vehicle k at each second of our simulation time of 200 sec. We record the number of instances of the two incidents of Section 4.1.1 and 4.1.2, i.e., 1-hop becoming 2-hop via MPR i for node k or vice versa. Our

goal is to identify whether MPR i is behind node k. The performance metric which we term as "Percentage of Correct Prediction (PCP)" is calculated by the following formula = (number of times MPR i is correctly identified as back for node k) / (number of times MPR i is actually behind k during the two instances of Section 4.1.1 and 4.1.2). We repeat for 21 trials to obtain the PCPs with 95% confidence interval. The confidence interval is calculated using the results of 21 trials as a vindication that our results are not biased towards any particular vehicle configuration. In each run, the flows of the vehicles have been randomly generated. The results are listed in Table 1.

Table 1: PCPs (in percentage with confidence interval) with varying vehicle speeds (km/hr)

Speed	20	50	80	100
PCP	99.5 ± 0.5	99.5 ± 0.5	95 ± 2.1	89 ± 6

As can be seen, our algorithm detects almost all back MPR cases during the two incidents of Section 4.1.1 and 4.1.2 for lower speeds. However, our algorithm's performance slightly suffers because of speeding vehicles, such that the number of "undetected" cases increase, as can be seen from Table 1 – 5% for 80 km/hr case compared to only 0.5% for 20km/hr. Since speeding vehicles would most likely create stale states in the OLSR neighbor tables, thereby, our algorithm fails to correlate network layer information with relative distances correctly for the identification. Incorrect identification of back MPRs (i.e., false positive) is seen to be non-existent for lower speeds, however, it rises slightly for higher speeds but is still negligible ($< 0.2\%$). The identification results of MPRs that are not back, show a similar trend as the back MPR identification results. These results are not presented for brevity. Just like false positives, false negatives are also seen to be quite negligible.

Fig. 9: Performance of our algorithm in case of accidents/traffic jam. (a) Back MPR Identification of the Leading vehicle which is stopped at 360 sec. (b) Front MPR Identification of the Leading vehicle which is stopped at 360 sec. (c) Back MPR Identification of the Middle vehicle which is stopped at 450 sec. (d) Front MPR Identification of the Middle vehicle which is stopped at 450 sec.

6.2 Double lane experiments with simulated accident/traffic jam

In this experiment, the vehicles are moving in the same direction along two lanes (Fig. 2b). Each lane is 10km long and our simulation time is 600 sec. There are 15 vehicles in one lane with average speed of 20km/hr while another 15 vehicles are moving in the other lane with average speed of 60km/hr. Accident is simulated by two different ways –

- The leading vehicle of each lane is stopped at distance 2km around 360 sec.

- The middle vehicle of each lane is stopped at distance 2km around 450 sec.

Fig. 9a and 9b show our algorithm's performance for the first case, while Fig. 9c and 9d depict the performance for the second case. We consider one of the two stopped vehicles from the two lanes in each case. In both cases, the back MPR for the stopped vehicle is detected within 1 min as can be verified from Fig. 9a and Fig. 9c. The "undetected" front MPR in Fig. 9d would move away farther with time and thereby, will not be an MPR of the stopped vehicle anymore. For the leading vehicle stop scenario, there should not be any front MPR which is evident in Fig 9b. In Fig. 10, we list the ratios of identified back/front MPRs for all the vehicles that were approaching the stopped leading vehicle. After roughly 540 sec., all of the approaching vehicles' MPRs are seen to be correctly identified which is almost 3 minutes later than the vehicle stopping occurrence. For the front MPR identification, it took a little additional time. In other words, the traffic jam or accident information is expected to be spread in directed fashion over the whole network within $3 \sim 4$ minutes. Note that the empty ratios in Fig. 10 do not always mean our algorithm could not detect the back/front MPRs. Most of the times, there was no MPRs for the vehicles as mentioned previously.

Fig. 10: Performance of our algorithm in case of accidents/traffic jam. (a) Ratios of identified Back MPRs for all the vehicles behind the leading vehicle which is stopped at 360 sec. (b) Ratios of identified front MPRs for all the vehicles behind the leading vehicle which is stopped at 360 sec.

We consider a junction to be an important road landmark for our algorithm since it is a location where the vehicles might stop or change speeds. As seen in the experimental results, such incidents facilitate the backward vehicles' identification algorithm's performance for directed dissemination of traffic information. We contend that our scheme is applicable for a junction as in Fig. 11, but only for the vehicles moving in the same direction along the lanes. Such scenarios would simply be an extension of a single or double lane scenarios discussed here. The algorithm would be applied to the independent streams of vehicles moving in the same direction (i.e., not considering the vehicles oppositely directed). We repeated our experiments for such junctions but due to page limitations these results are not provided but they show similar trends to the results (e.g., Fig. 8, 9, 10) discussed in this section. Note that, we also assume the directions of the vehicles could easily be obtained through simple inexpensive compass or more sophisticated magnetometer type of sensors. To detect backward vehicles in the opposite direction, the algorithm requires substantial changes, which is not within the scope of our current work. We foresee this research as an extension of the current proposal. The resultant algorithm would be able to identify all the backward vehicles regardless of their directions.

Fig. 11: An example of a junction scenario. Our algorithm could be applied to the independent streams of vehicles moving in the same direction.

7 Experimental Results

We conducted real experiments to verify our back MPR identification algorithm's effectiveness with ad-hoc nodes moving at walking speeds. In this field-test, four persons carrying laptops integrated with OLSR [27], and a HOLUX M-1000 GPS device create an ad-hoc network. We emulated the two events by movement (i.e., 2-hop neighbor becoming 1-hop and vice versa) which trigger our algorithm. We developed a plugin for the OLSR daemon which logs the OLSR neighbor table information, and the speed information collected from the GPS device at every second. Subsequently, the back MPR identification algorithm is run offline utilizing the logs in order to generate the results presented here. Note that, the algorithm is executed offline only for convenience so as to generate different types of results. It could easily have been integrated within the OLSR plugin, currently run by each individual node to produce various log files to be used by the algorithm offline.

Fig. 12: The scenario when the front vehicle is met with an accident – all the other vehicles also came to halt. (a) 1-to-2 hop scenario: the accident met vehicle #6 is at 2-hop distance from stopped vehicle #9. Initially all the vehicles were neighbors to each other. (b) 2-to-1 hop scenario: the accident met vehicle #6 was at 2-hop distance from vehicle #9. After the accident, vehicle #5, #6, #8 and #9 are all neighbors.

Fig. 13: The scenario when the middle vehicle is met with an accident – all the vehicles behind it also came to halt. (a) 1-to-2 hop scenario: the accident met vehicle #5 is at 2-hop distance from stopped vehicle #9. Initially all the vehicles were neighbors to each other. (b) 2-to-1 hop scenario: the accident met vehicle #5 was at 2-hop distance from vehicle #9. After the accident, vehicle #5, #8 and #9 are all neighbors while vehicle #6 moves away.

We emulate the two accident scenarios of Section 6.2 where the leading (Fig. 12) or middle vehicle (Fig. 13), i.e., the person carrying laptop is stopped after a certain time interval. On each occasion, we try to generate the two neighbor table change scenarios which trigger our algorithm. At first, each vehicle is stationary for 1 minute at the starting positions (see the positions of the vehicles on the left-hand side of the arrow in Fig. 8 and 9). Then, each person starts walking along a straight lane for 3 minutes after which it is assumed that the accident takes place. Then, we continue taking readings for another minute which makes the total duration of each scenario of Fig. 12a, 12b, 13a, 13b five minutes long.

7.1 Leading vehicle stop scenario

As mentioned previously, the vehicles, i.e., persons carrying the laptops are moving in the same direction at walking speed in this experiment. The leading vehicle is stopped after four minutes to emulate the accident scenario. The states as MPR (front or back) w.r.t. leading vehicle #6 of the other nodes of Fig. 12's scenarios are shown in Fig. 14 at 30 sec. time interval.

Fig. 14e and 14f reveal that vehicle #5 who is behind the leading vehicle #6 does not generally become an MPR of #6. When it is before the accident, it was correctly identified (see Fig. 14e) following the accident. Fig. 14c and 14d show vehicle #8 which is in the middle of #5 and #9 in Fig. 12 is correctly identified as back MPR after the accident, i.e., after 240 sec. Fig. 14b depicts vehicle #9 which is the furthest back in Fig. 12b does not become an MPR of #6. However, Fig. 14a shows that vehicle #9 does become an MPR at some time instances even if it was the furthest back. This is due to an OLSR's neighbor table's inconsistent entry, where a node is seen to reach a node geographically closer to it (#5) via a node that is actually farther from it (#9). The table entries were also observed to change quite frequently. Based on our results, it can be stated that all of the approaching vehicles' MPRs are seen to be correctly identified simultaneously with the accident occurrence. We observe a similar phenomenon of MPR states w.r.t. the other vehicles' (#5, #8, and #9) results that are not shown here for brevity. Note that, in the resultant figures, all the vehicles' numbers appear as the last octet of its IP, e.g., 192.168.1.vehicle#.

Fig. 14: Vehicle #6 of Fig. 12 is met with an accident at 240 sec. The figures 14a, 14c, and 14e correspond to the scenario of Fig. 12a, while 14b, 14d, and 14f correspond to the scenario of Fig. 12b. (a) Vehicle (#9) at the furthest back. (b) Vehicle (#9) at the furthest back. (c) Vehicle (#8) at the middle. (d) Vehicle (#8) at the middle. (e) Vehicle (#5) at the immediate back. (f) Vehicle (#5) at the immediate back.

Fig. 15: Vehicle #5 of Fig. 13 is met with an accident at 240 sec. The figures 15a, 15c, and 15e correspond to the scenario of Fig. 13a, while 15b, 15d, and 15f correspond to the scenario of Fig. 13b. (a) Vehicle (#9) at the furthest back. (b) Vehicle (#9) at the furthest back. (c) Vehicle (#8) at the immediate back. (d) Vehicle (#8) at the immediate back. (e) Vehicle (#6) at the front. (f) Vehicle (#6) at the front.

7.2 Middle vehicle stop scenario

In this experiment, the middle vehicle (#5) of Fig. 13 stops after four minutes to emulate the accident scenario. Similar to the "leading car stop scenario" of the previous section, all the other nodes' states as MPR (front or back) w.r.t. middle vehicle #5 are shown in Fig. 15 at 30 sec. time interval. Fig. 15e and 15f show the cases where the front vehicle #6 is selected as MPR by #5 to reach a node (#8 or #9) that is geographically actually behind it. However, our algorithm still correctly identifies this MPR as front. Identifying other vehicles' (#8 or #9) MPR states follow similar trend as in the previous section, i.e., they are generally correctly identified after the accident occurrence. We also observe similar phenomenon of MPR states w.r.t. the other vehicles (#6, #8, and #9) for the middle car stop scenario, which are not shown here for brevity. Fig. 16 shows vehicle #8's MPR states w.r.t. the stopped vehicle #5 at various time intervals. This result depicts the importance of how fast the MPR state could be identified, and

thereby assist in timely directed propagation of accident information. For the case of Fig. 13a, we observe that this information is detected immediately, while for 13b, it takes nearly one minute to be detected. For other cases, the detection time is within 1 minute, which is a improvement from the simulated evaluation, where it takes 3 to 4 minutes to disseminate the accident information over the whole network.

Fig. 16: Vehicle #5 of Fig. 13 is met with an accident at 240 sec. Snapshot of the identified back MPR (vehicle #8) for different time intervals: (a), (c) & (e) for Fig. 13a, and (b), (d) & (f) for Fig. 13b. (a) Back MPR identification for 5-sec time intervals. (b) Back MPR identification for 5-sec time intervals. (c) Back MPR identification for 10-sec time intervals. (d) Back MPR identification for 10-sec time intervals. (e) Back MPR identification for 20-sec time intervals. (f) Back MPR identification for 20-sec time intervals.

8 Conclusion and Future Work

In this paper, we discuss the idea of directed traffic information propagation using MPRs of OLSR, which incurs very little overhead to VANET. The traffic/accident information can be embedded inside OLSR's TC (topology control) message [2] for propagation in a manner that only back MPRs retransmit the information (not all of its MPRs). We also utilize the OLSR's neighbor table to detect hop distance changing events, and HELLO messages to exchange speed information among neighboring vehicles. Hence, we do not incur an extra burden on the OLSR protocol, thereby achieving our goal of reduced network traffic. The back MPR identification algorithm performed well in general, especially during certain drastic traffic conditions such as traffic jam or accident. There were no false positives (i.e., MPR that is ahead is identified as back) in those scenarios. Similar to any VANET application, our algorithm is also seen to suffer from the discontinuity of network connectivity (i.e., no MPRs) for vehicles moving at higher speeds. We carried out both simulated and real experimentation to verify the effectiveness of our algorithm.

We foresee a number of future work directions. Although our back MPR identification algorithm is analyzed in detail, more experiments could be planned with vehicles under normal road conditions (e.g., different densities, speeds, radio ranges, etc.). Our current algorithm is tailored to work for identifying backward vehicles that are moving in the same direction. In the future, we plan to extend the scope of our algorithm by including vehicles moving in all directions. During real experiments, we also observe inconsistent neighbor table entries in the OLSR protocol which affected our algorithm's performance. Experiments with newer releases of OLSR may overcome this inconsistency.
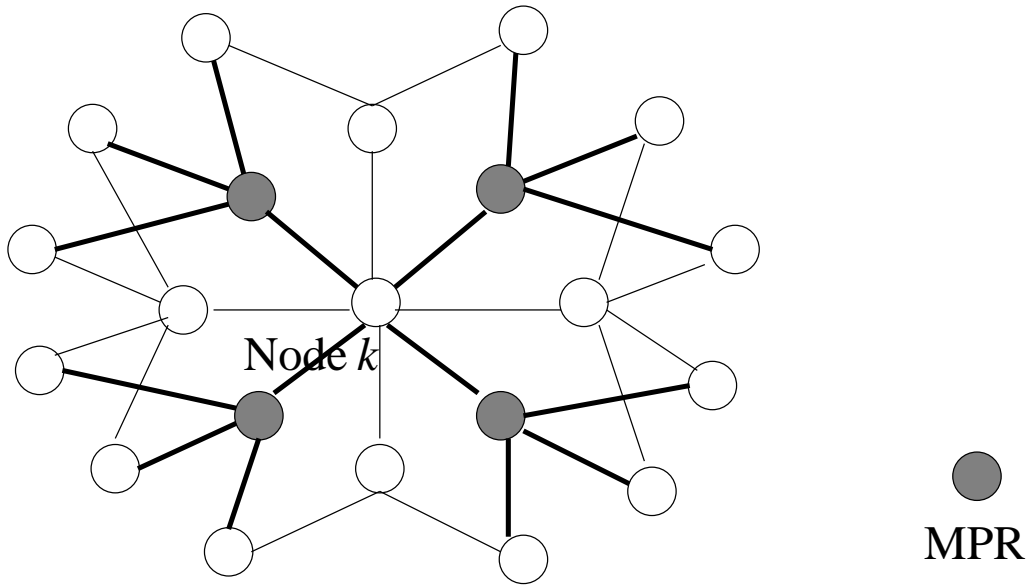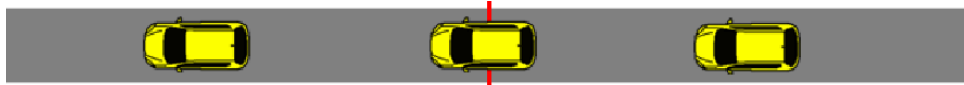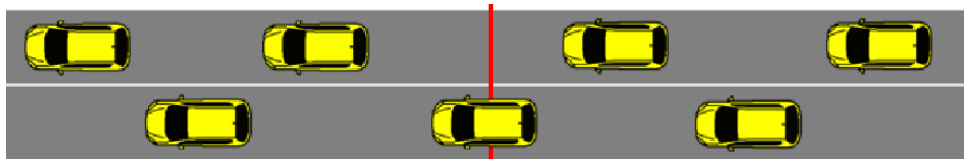
Acknowledgements

References

1. Hossain AKMM, Mekbungwan P, Kanchanasut K: **Directed Information Dissemination in VANET**. In *AINTEC'11*, Bangkok, Thailand 2011.
2. Jacquet P, Mühlethaler P, Clausen T, Laouiti A, Qayyum A, Viennot L: **Optimized Link State Routing Protocol for Ad Hoc Networks**. In *Proc. of IEEE INMIC* 2001.
3. Mohammad S, Rasheed A, Qayyum A: **VANET Architectures and Protocol Stacks: A Survey**. In *Communication Technologies for Vehicles*, *Volume 6596 of* Lecture Notes in Computer Science, Springer Berlin Heidelberg 2011:95–105.
4. Li F, Wang Y: **Routing in vehicular ad hoc networks: A survey**. *IEEE Vehicular Technology Magazine* 2008, **2**(2):12–22.
5. Luís M, Oliveira R, Bernardo L, Pinto P: **Improving routing performance in high mobility and high density ad hoc vehicular networks**. In *Proceedings of the 33rd IEEE conference on Sarnoff*, NJ, USA 2010:121–126.
6. Toutouh J, Garcia-Nieto J, Alba E: **Optimal Configuration of OLSR Routing Protocol for VANETs by Means of Differential Evolution**. In *Proceedings of the International Conference on Metaheuristics and Nature Inspired Computing (META'10)*, Tunisia 2010.
7. Ermel E, Mühlethaler P: **Using OLSR Multipoint Relays (MPRs) to estimate node positions in a Wireless Mesh Network**. In *Proceedings of the European Wireless* 2007.
8. Minenoa H, Sogab K, Takenakad T, Terashimac Y, Mizunod T: **Integrated protocol for optimized link state routing and localization: OLSR-L**. *Simulation Modelling Practice and Theory* 2011, **19**(8):1711–1722.
9. Clark BN, Colbourn CJ, Johnson DS: **Unit disk graphs**. *Discrete Math.* 1991, **86**(1-3):165–177.
10. Nzouonta J, Borcea C: **STEID: A protocol for emergency information dissemination in vehicular networks**. Tech. rep., Department of Computer Science, NJIT 2006.
11. J Blum AE, Hoffmman L: **Challenges of intervehicle ad hoc networks**. *IEEE Trans. Intelligent Transportation Systems* 2004, **5**(4):347–351.
12. Ziliaskopoulos A, Zhang J: **A Zero Public Infrastructure Vehicle Based Traffic Information System**. In *Proc. of TRB* 2003.
13. Wischoff L, Ebner A, Rohling H, Lott M, Halfmann R: **SOTIS-a self-organizing traffic information system**. In *Proc. of IEEE VTC-Spring* 2003.
14. Hartenstein H, Bochow B, Ebner A, Lott M, Radimirsch M, Vollmer D: **Position-aware ad hoc wireless networks for inter-vehicle communications: the Fleetnet project**. In *Proc. of ACM MobiHoc* 2001.
15. Goel S, Senouci S, Ozbay K: **Ascertaining Viability of Wi-Fi based Vehicle-to-Vehicle Network for Traffic Information Dissemination**. In *Proc. of IEEE ITS* 2004.
16. Zeng Y, Xiang K, Li D, Vasilakos AV: **Directional routing and scheduling for green vehicular delay tolerant networks**. *Wireless Networks* 2012, [http://dx.doi.org/10.1007/s11276-012-0457-9].
17. Zhang Z, Mao G, Anderson B: **On the Information Propagation Process in Mobile Vehicular Ad Hoc Networks**. *Vehicular Technology, IEEE Transactions on* 2011, **60**(5):2314–2325.
18. Ma X, Zhang J, Yin X, Trivedi K: **Design and Analysis of a Robust Broadcast Scheme for VANET Safety-Related Services**. *Vehicular Technology, IEEE Transactions on* 2012, **61**:46–61.
19. Jerbi M, Jannotti J, Rasheed T, Ghamri-Doudane Y: **An Infrastructure-Free Traffic Information System for Vehicular Networks**. In *Proc. of IEEE VTC-Fall* 2007.
20. Ghafoor KZ, Abu Bakar K, Lloret J, Khokhar RH, Lee KC: **Intelligent beaconless geographical forwarding for urban vehicular environments**. *Wireless Networks* 2012, [http://dx.doi.org/10.1007/s11276-012-0470-z].
21. Lo NW, Tsai HC: **A Reputation System for Traffic Safety Event on Vehicular Ad Hoc Networks**. *EURASIP Journal on Wireless Communications and Networking* 2009, **2009**:9:1–9:2.
22. Viennot L: **Complexity Results on Election of Multipoint Relays in Wireless Networks**. Tech. rep., INRIA 1998.
23. Qayyum A, Viennot L, Laouiti A: **Multipoint Relaying: An Efficient Technique for flooding in Mobile Wireless Networks**. Tech. Rep. RR-3898, INRIA 2000.
24. Cormen TH, Leiserson CE, Rivest RL, Stein C: *Introduction to Algorithms*. The MIT Press, 3rd edition 2009.
25. Simulation of Urban MObility (SUMO): **Open source, highly portable, microscopic road traffic simulation package**[http://sumo.sourceforge.net].
26. The ns-3 network simulator: **A discrete-event network simulator**[http://www.nsnam.org].
27. olsrd: **An adhoc wireless mesh routing daemon**[http://www.olsr.org].

Node k

MPR

(a)

(b)

2

(a)

(b)

Leading Car Stop Scenario

(a)

Leading Car Stop Scenario

(b)

Middle Car Stop Scenario

(c)

Middle Car Stop Scenario

(d)

Leading Car Stop Scenario

(a)

Leading Car Stop Scenario

(b)

10

Moving direction

Accident

9 **(a)** 8 5 6 9 8 5 6

Moving direction

Accident

9 **(b)** 8 5 6 9 8 5 6

(a)

(b)