A Review of Information Dissemination Protocols for Vehicular Ad Hoc Networks

Sooksan Panichpapiboon, *Member, IEEE*, and Wasan Pattara-atikom, *Member, IEEE*

Abstract—**With the fast development in ad hoc wireless communications and vehicular technology, it is foreseeable that, in the near future, traffic information will be collected and disseminated in real-time by mobile sensors instead of fixed sensors used in the current infrastructure-based traffic information systems. A distributed network of vehicles such as a vehicular ad hoc network (VANET) can easily turn into an infrastructure-less self-organizing traffic information system, where any vehicle can participate in collecting and reporting useful traffic information such as section travel time, flow rate, and density. Disseminating traffic information relies on broadcasting protocols. Recently, there have been a significant number of broadcasting protocols for VANETs reported in the literature. In this paper, we classify and provide an in-depth review of these protocols.**

Index Terms—**Broadcasting protocols, vehicular ad hoc networks, traffic information systems.**

I. Introduction

WITH the fast development in ad hoc wireless communications and vehicular technology, it is foreseeable that, in the near future, there will be a paradigm shift in traffic information systems. In particular, real-time traffic data will be collected and disseminated by distributed mobile probes, instead of fixed sensors used in the current infrastructure-based systems. A distributed network of vehicles such as a vehicular ad hoc network (VANET) can easily turn into an infrastructure-less self-organizing traffic information system, where any vehicle can become a mobile sensor, participating in collecting and disseminating useful traffic information such as section travel time, flow rate, and density.

Disseminating traffic information in a VANET is a unique problem. In contrast to the unicast data typically transmitted in a network such as the Internet, the traffic information generally has a broadcast-oriented nature. In other words, the traffic information is of public interest, and it usually benefits a group of users rather than a specific individual. Consequently, it is more appropriate to use a broadcasting scheme rather than a unicast routing scheme in disseminating the traffic information. The main advantage of a broadcasting scheme is that a vehicle does not need to know a destination address and a route to a specific destination. This eliminates the complexity of route discovery, address resolution, and topology management, which are difficulties in dynamic networks such as VANETs. In this paper, we will mainly focus our attention on broadcasting protocols for VANETs. For information on unicast routing protocols, readers are referred to [1], [2]. In addition, we will only concentrate on source broadcasting (i.e., distributing packets in a one-to-all type of scenarios). Other types of information dissemination methods such as geocasting [3], [4], [5], multicasting [6], [7], [8], peer-to-peer content distributing [9], [10], [11], and streaming [12], [13], [14] will not be discussed in this paper.

Over the past few years, there have been a number of broadcasting protocols for VANETs reported in the literature. However, they can generally be divided into two main categories:

- Multi-hop Broadcasting
- Single-hop Broadcasting

A major contrast between these two types of protocols is in the way that the information packets are spread in the network.

In multi-hop broadcasting, a packet propagates through the network by way of flooding. In general, when a source vehicle broadcasts an information packet, some of the vehicles within the vicinity of the source will become the next relay vehicles (nodes) and perform a relaying task by rebroadcasting the packet further. Similarly, after a relay node rebroadcasts the packet, some of the vehicles in its vicinity will become the next relay nodes and forward the packet further. As a result, the information packet will be able to propagate from the source to the other distant vehicles.

On the contrary, in single-hop broadcasting, vehicles do not flood the information packets. Instead, when a vehicle receives a packet, it keeps the information in its on-board database. Periodically, each vehicle selects some of the records in its database to broadcast. Thus, with single-hop broadcasting, each vehicle will carry the traffic information with itself as it travels, and this information will be transferred to other vehicles in its one-hop neighborhood in the next broadcast cycles. Ultimately, a single-hop broadcasting protocol relies heavily on vehicle mobility in spreading the information. As a first glance, the broadcasting protocols discussed in this paper are classified as shown in Fig. 1.

The rest of the paper is organized as follows. Multi-hop broadcasting protocols and single-hop broadcasting protocols are reviewed in Section II and in Section III, respectively. In Section IV, we discuss the performance evaluation of these protocols. Summary and discussion on open issues related to broadcasting protocols in VANETs are provided in Section V. A list of acronyms is given in the Appendix.

Manuscript received 26 September 2010; revised 26 March 2011.

S. Panichpapiboon is with the Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Bangkok 10520, Thailand (e-mail: sooksan@it.kmitl.ac.th).

W. Pattara-atikom is with the Network Technology Laboratory, National Electronics and Computer Technology Center (NECTEC), Pathumthani 12120, Thailand (e-mail: wasan@nectec.or.th).

Digital Object Identifier 10.1109/SURV.2011.070711.00131

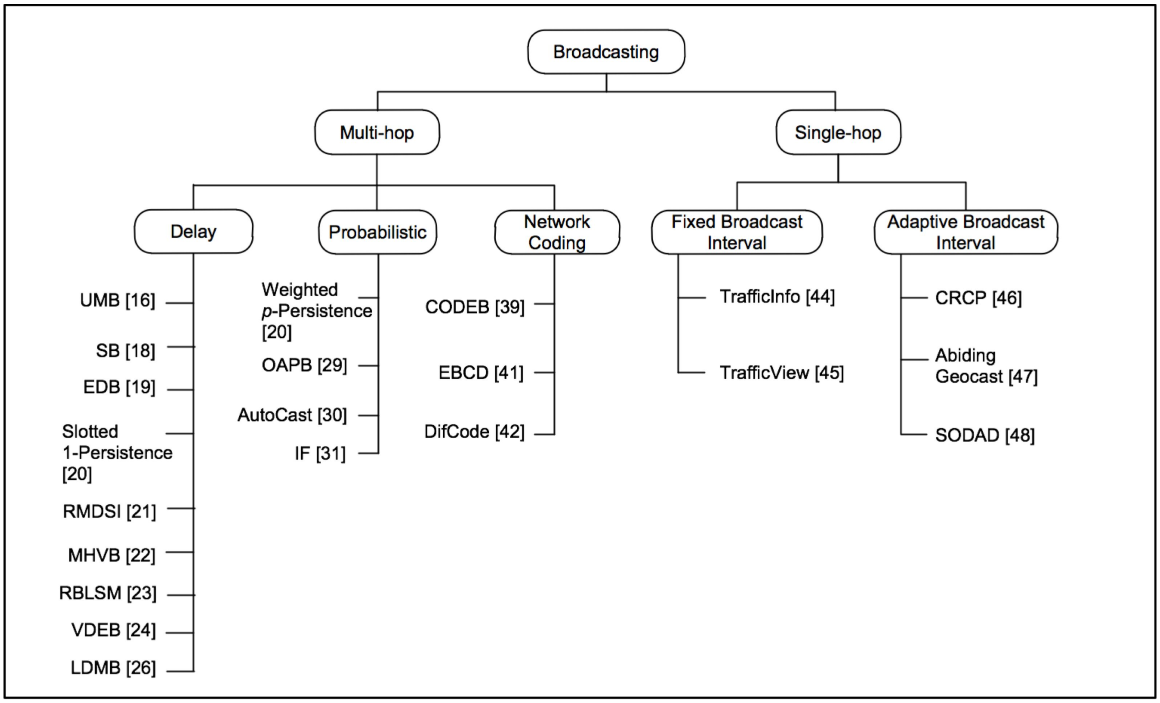

Fig. 1. A classification of broadcasting protocols for VANETs.

II. Multi-hop Broadcasting

As mentioned earlier, in multi-hop broadcasting, an information packet propagates through the network by way of flooding. However, a *pure* flooding scheme, where every single vehicle rebroadcasts the packet, is inefficient because of two main reasons: (i) scalability and (ii) packet collision. As the network becomes denser, the same information packet will be rebroadcasted more redundantly. This wastes the limited radio channel bandwidth; thus, it makes pure flooding not scaled with the network density. In addition, in a dense network, packet collision becomes a severe problem since a large number of vehicles in the same vicinity may rebroadcast the packet at the same time. This is usually referred to as a broadcast storm problem [15]. A good multi-hop broadcasting protocol must be able to resolve these issues.

A common solution employed by most researchers to solve the scalability and collision problems is reducing the number of redundant rebroadcast packets. This is typically done by selecting only some of the vehicles to relay the packet as opposed to letting every single vehicle rebroadcast it. In the succeeding sections, we discuss existing approaches used in reducing the number of packet transmissions in details.

A. Delay-Based Multi-hop Broadcasting

In a delay-based approach, a different waiting delay before rebroadcasting the packet is assigned to each receiving vehicle [16], [17], [18], [19], [20], [21], [22], [23], [24], [25], [26]. Basically, the vehicle with the shortest waiting delay gets the highest priority in rebroadcasting the packet. In addition, in order to avoid redundancy, the other vehicles abort their waiting process once they know that the packet has already been rebroadcasted. Typically, the delay assigned to each vehicle is a function of the distance between the vehicle and the transmitter. Generally, the farthest vehicle is given the shortest delay and is implicitly selected as the next rebroadcast node, since it maximizes the packet forward progress. In the following, we review a variety of delay-based multi-hop broadcasting protocols.

1) Urban Multi-hop Broadcast (UMB): The UMB protocol [16] is designed to solve the broadcast storm, the hidden node, and the reliability problems in multi-hop broadcasting. Basically, UMB divides a road inside the transmission range of a transmitter into small segments, and it gives the rebroadcast priority to the vehicles that belong in the farthest segment. In UMB, two types of packet forwarding are defined: (i) *directional broadcast* and (ii) *intersection broadcast*. The directional broadcast works as follows. When a vehicle has a packet to send, it first transmits a control packet called *Request-to-Broadcast (RTB)*, which includes its own position and the direction of packet propagation. Once the vehicles in the transmission range of the transmitter receives the RTB packet, each of them starts transmitting a jamming signal called *black-burst* for a specified period of time. The duration of the black-burst is a function of the distance between the vehicle that receives the RTB packet and the transmitter. More specifically, each vehicle computes its own black-burst duration according to the following function

$$L = \left\lfloor \frac{d}{R} \times N_{\max} \right\rfloor \times S \qquad (1)$$

where L is the black-burst duration, d is the distance between the vehicle and the transmitter, R is the transmission range, $N_{\max}$ is the number of road segments inside the transmission range, and S is the time slot duration. With this assignment function, a vehicle that is farther away from the transmitter will have a longer black-burst duration.

After a vehicle has transmitted the black-burst according to the duration calculated in (1), it listens to the channel again. If it detects that the channel is busy, then this means that the other vehicles are still transmitting the black-burst. In this case, the vehicle does nothing and relegates the rebroadcast duty to those vehicles that are still transmitting the black-burst. On the other hand, if the vehicle finds the channel idle after its black-burst transmission, it transmits a control packet called *Clear-to-Broadcast (CTB)* back to the vehicle that initiates the RTB packet. The vehicle that successfully transmits the CTB packet will be designated as the next rebroadcast node.

Once the vehicle that initiates the RTB packet receives the CTB packet, it can start transmitting a DATA packet. When the designated rebroadcast node receives the DATA packet, it acknowledges by transmitting an ACK packet back to the data transmitter. If the ACK packet is not received within a certain amount of time, the whole process (RTB-CTB-DATA-ACK) starts over again. However, UMB is not a collision-free protocol. It is possible that there are more than one vehicle on the same road segment, and these vehicles may transmit the CTB packet at the same time, which will result in a collision. It is proposed in [16] that the collision be resolved among these vehicles by further splitting the segment into $N_{\max}$ subsegments and let the vehicles in conflict repeat the black-burst transmission process again until the transmission of the CTB packet is successful. The vehicle that wins the CTB contention will be designated as the next rebroadcast vehicle.

The second type of forwarding function defined in UMB is intersection broadcast. This forwarding function is used for broadcasting a packet at a road intersection. It is suggested that a repeater be installed at an intersection in order to forward a packet to other road directions. The UMB protocol is later extended in [17] so that fixed repeaters at intersections can be eliminated.

2) Smart Broadcast (SB): SB [18] is proposed to improve the limitation of UMB. It is pointed out in [18] that UMB is inefficient in a sense that the next rebroadcast vehicle has to wait the longest before being able to transmit the CTB packet. This is because the longest black-burst duration is assigned to the next rebroadcast vehicle. SB solves this problem by assigning the next rebroadcast vehicle the shortest waiting delay.

A packet forwarding process in SB is done as follows. First, when a source vehicle has a packet to send, it transmits a RTB packet which contains its location and other information such as packet propagation direction and contention window size. Second, the vehicles in the range of the source that receive the RTB packet determine the "sector" in which they belong by comparing their locations with that of the source vehicle. Next, each vehicle that receives the RTB packet chooses a contention delay based on the sector that it resides. According to [18], given that there are N_s sectors, the waiting delay W_r for vehicles in sector r is randomly obtained from the following set

$$W_r \quad = \quad \{(r-1)cw, (r-1)cw+1, \ldots, rcw-1\} \quad (2)$$

where $r = 1, 2, \ldots, N_s$ is the sector number ($r = 1$ refers to the outermost sector), and cw is the duration of contention window. With this delay function, the vehicles in the outermost sector (i.e., farthest away from the source) will be given the shortest waiting time. In addition, since vehicles in the same sector randomly pick the waiting times, this further reduces packet collision.

When its waiting time expires, a vehicle transmits a CTB packet back to the source. Once the source receives the CTB packet successfully, it then transmits the data packet. The data packet also specifies the identification (ID) of the vehicle that is chosen as the next rebroadcast vehicle. This protocol is better than UMB in terms of latency. As mentioned earlier, a designated rebroadcast vehicle in the UMB protocol will have the highest waiting delay before rebroadcasting, but in the SB protocol a designated rebroadcast vehicle will have the shortest delay.

In [18], the performance of SB is compared with that of two other protocols, which are the UMB protocol and the Geographic Random Forwarding (GeRaF) protocol [27]. GeRaF is a position-based routing protocol, where a packet will be routed toward the geographical location of a destination. Instead of using a destination address, a source specifies the geographical area of the destination node in a packet. A packet will be cooperatively forwarded by intermediate nodes that receive it. Basically, when a node receives a packet, it decides whether it should act as a relay based on its relative position to the destination. It is shown that SB outperforms UMB and GeRaF in message propagation speed. The main reason is that UMB and GeRaF have a higher number of collisions when the vehicle density increases; therefore, they waste more time in resolving the collisions. It is also shown that the message propagation speed in SB is constant even when the vehicle density increases. Thus, SB is also robust to density changes. In addition, it is also later reported in [28] that a timer-based protocol such as SB performs better than a black-burst-based protocol such as UMB.

3) Efficient Directional Broadcast (EDB): EDB [19] is a delay-based multi-hop broadcasting protocol that works quite similar to UMB and SB protocols. However, the RTB and CTB control packets are not used in this protocol. In addition, EDB also exploits the use of directional antennas. In particular, it is proposed that each vehicle be equipped with two directional antennas, each with 30-degree beamwidth.

Similar to UMB, there are two types of packet forwarding in EDB, namely directional broadcast on the road segment and directional broadcast at the intersection. In directional broadcast on the road segment, a source vehicle broadcasts a packet and the downstream vehicles will rebroadcast it further. To reduce the number of redundant rebroadcast packets, EDB assigns a different waiting time before rebroadcasting to each vehicle within the range of the transmitter. The waiting time is a function of the distance between the vehicle and the transmitter. In fact, when a vehicle receives a packet, it computes its own waiting time according to the following function

$$W \quad = \quad \left(1 - \frac{d}{R}\right) maxWT \quad (3)$$

where R is the transmission range, d is the distance between the vehicle and the transmitter, and $maxWT$ is the maximum waiting time. With this waiting time assignment function, a vehicle that is farther away from the transmitter will be given a higher rebroadcast priority. After its waiting time expires, the vehicle immediately transmits an ACK packet. This is done to inform the other neighboring vehicles that they need not perform the rebroadcasting task. After transmitting the ACK packet, the vehicle can start rebroadcasting the data packet. In addition, in order to increase reliability, the vehicle will periodically keep rebroadcasting the packet if no other vehicle forwards the packet within the interval $maxWT$. Similar to the UMB protocol, it is proposed in [19] that a repeater be installed at an intersection to broadcast the packets further to other road directions.

The performance of EDB is compared, by simulation, with two other variations called Random Directional Broadcast (RDB) and Simple Distance-based Directional Broadcast (SDDB). These variations differ by how a waiting delay is assigned. In RDB, each receiver simply waits for a random time before rebroadcasting the packet, and in SDDB each receiver waits for the duration computed in (3) but no ACK packet is used. It is shown that the EDB protocol performs better than the other two variations in terms of packet delivery ratio and average forward nodes ratio (a ratio between the number of vehicles that rebroadcast the packet and the total number of vehicles in the network). The main reason why EDB has a higher packet delivery ratio is that it can reduce packet collision through the use of ACK packets. In addition, since the neighbors of a rebroadcast node are informed through a quick transmission of an ACK packet, they can refrain from rebroadcasting the duplicates. This improves the average forward nodes ratio.

4) Slotted 1-Persistence Broadcasting: Packet forwarding in the Slotted 1-Persistence Broadcasting protocol [20] is similar to those of the other delay-based multi-hop broadcasting protocols, where the rebroadcast priority is given to the vehicles that are farther away from the transmitter. In this protocol, when a vehicle receives a packet, it rebroadcasts the packet according to an assigned time slot, where the time slot is a function of the distance between the vehicle and the transmitter. In particular, each vehicle computes the time slot in which it will rebroadcast the packet based on the following function

$$T_{S_{ij}} = S_{ij} \times \tau \tag{4}$$

where τ is the estimated one-hop propagation and medium access delay, and S_{ij} is the assigned slot number. The assigned slot number is computed from

$$S_{ij} = N_s \left(1 - \left\lceil \frac{\min(D_{ij}, R)}{R} \right\rceil \right) \tag{5}$$

where D_{ij} is the distance between transmitter i and vehicle j, R is the transmission range, and N_s is the pre-determined number of slots. It is stated in [20] that N_s should be chosen carefully based on the traffic density. In general, the value of N_s should increase as the traffic becomes denser. Next, a vehicle rebroadcasts the packet in the time slot computed in (4) if it hears the packet for the first time and no one has transmitted the packet before its waiting time expires. It is possible that more than one vehicle will transmit in the same time slot, resulting in packet collision. Thus, the number of slots will have an impact on the protocol performance.

A similar slotted-based broadcasting protocol called Vehicle-density-based Emergency Broadcasting (VDEB), where a waiting time slot is assigned based on the distance between a transmitter and a receiver, is presented in [24]. As an improvement over the Slotted 1-persistence protocol, VDEB explicitly takes the vehicle density into consideration when determining an appropriate number of slots to use. The density is estimated from the number of neighbors around the transmitter. Another variation of the Slotted 1-persistence protocol can also be found in [25].

5) Reliable Method for Disseminating Safety Information (RMDSI): The RMDSI protocol [21] aims at solving the reliability problem when the network becomes disconnected. This protocol also uses delay to differentiate the rebroadcast priority of each vehicle. Similar to EDB, when a vehicle receives a packet, it computes a waiting time before rebroadcasting the packet according to the function given in (3). After the waiting time expires, the vehicle rebroadcasts the packet. Vehicles that hear the duplicate rebroadcast before their waiting times expire cancel their retransmissions.

An additional feature of this protocol is a mechanism for solving the network fragmentation problem. Basically, a relay vehicle will keep a copy of the packet that it has just rebroadcasted until it hears a duplicate transmission by the next relay node or until the packet lifetime expires. Not hearing a duplicate rebroadcast by the next relay vehicle is an indication that the network may be fragmented. In this case, the relay vehicle keeps rebroadcasting a small control packet until it finds the next relay vehicle to which it can forward the packet. In the case that the packet lifetime expires before the vehicle is able to find the next relay vehicle, it stops rebroadcasting that packet.

The performance of this protocol, in terms of the percentage of vehicles that receive the broadcast packets, is compared with that of the UMB protocol. It is shown, by simulation, that when the network is heavily fragmented, RMDSI performs better than UMB which does not take the network fragmentation problem into consideration.

6) Multi-hop Vehicular Broadcast (MHVB): MHVB [22] is also a delay-based multi-hop broadcasting protocol. Like other protocols in this category, when a vehicle receives a packet, it computes the waiting time before rebroadcasting the packet based on the distance between itself and the transmitter. A shorter waiting time will be assigned to a vehicle that is farther away from the transmitter. However, the waiting time assignment function is not explicitly given in [22]. After the waiting time expires, the vehicle rebroadcasts the packet. Vehicles that hear a duplicate rebroadcast from a vehicle that is farther away from themselves cancel their packet rebroadcast processes.

In addition, MHVB also has a traffic congestion detection feature. Intuitively, when the traffic is congested, the network will be dense. As a result, the interval in which each vehicle broadcasts its own information should be extended. The traffic congestion detection mechanism in MHVB is done as follows. Basically, each vehicle uses the number of its neighbors and its speed as an indication of congestion. For example, if vehicle A detects that the number of its neighbors is more than a threshold X and its own speed is less than a threshold $V_{\max}$, then this might indicate that the traffic is congested. In this case, a source vehicle increases its periodic broadcast interval. The performance of this protocol is evaluated via simulation. However, its performance is not compared with other existing protocols.

7) Reliable Broadcasting of Life Safety Messages (RBLSM): In RBLSM [23], after receiving a packet from the source, each node in the transmission range of the source determines its waiting time before rebroadcasting the packet. In contrast to the conventional strategy where the rebroadcast priority is given to the farthest vehicle, in this protocol the priority is given to the vehicle nearest to the transmitter. The main reason is that the nearer vehicle is considered more reliable than the vehicles that are farther away from the transmitter. For example, a nearer vehicle supposedly has a better received signal strength. This protocol also employs the use of the RTB and CTB control packets. The performance of the protocol is evaluated via simulation; however, only a single hop latency is provided.

A similar protocol which assigns the waiting delay based on a link quality, called Link-based Distributed Multi-hop Broadcast (LDMB), is also proposed in [26]. In computing the waiting delay, LDMB not only takes the distance between a transmitter and a receiver into consideration but it also takes other factors such as traffic density, transmission range, and packet transmission rate into account as well. However, LDMB does not perform significantly better, in terms of packet delivery ratio, than a simple distance-based protocol where the broadcast priority is given to the farther vehicle.

B. Probabilistic-based Multi-hop Broadcasting

While a different delay is assigned to each vehicle in a delay-based broadcasting protocol, a different rebroadcast probability is assigned to each vehicle in a probabilistic-based protocol. In probabilistic-based broadcasting, each vehicle rebroadcasts a packet according to its assigned rebroadcast probability [29], [30], [31], [32]. Since not all the vehicles will rebroadcast the packet, the number of redundant packets as well as the number of collisions are reduced. One of the main challenges in this type of protocols is in determining an optimal probability assignment function. There are many ways to assign the rebroadcast probability (also referred to as forwarding probability). While the simplest protocol uses a pre-defined fixed value for the forwarding probability, more sophisticated protocols let each vehicle adjust its forwarding probability dynamically based on factors such as vehicle location and network density. Some of the probabilistic-based multi-hop broadcasting protocols proposed in the literature are discussed in the following.

1) Weighted p-Persistence: In the Weighted p-Persistence protocol [20], a vehicle that receives a packet for the first time computes its own rebroadcast probability based on the distance between itself and the transmitter. The distance can be obtained by comparing its current position with the position of the transmitter specified in the packet. In particular, the rebroadcast probability is computed from the following function [20]

$$p_{ij} \;=\; \frac{D_{ij}}{R} \tag{6}$$

where D_{ij} is the distance between transmitter i and vehicle j, and R is the transmission range. Based on this function, the vehicles that are farther away from the transmitter will be given higher rebroadcast probabilities. However, this probability assignment function does not take the vehicle density into account; therefore, the number of rebroadcast packets can still be large if the network is dense. A similar probabilistic-based approach, where the rebroadcast probability is proportional to the distance between the transmitter and the receiver, is also considered in [32].

2) Optimized Adaptive Probabilistic Broadcast (OAPB): In OAPB [29], the forwarding probability is computed from the local vehicle density (in terms of the number of neighbors). To select an appropriate forwarding probability, each vehicle uses the local density information, which is obtained from exchanging periodic HELLO packets. In particular, when a vehicle receives a packet, it computes its own forwarding probability based on the following function

$$\overline{\phi} \;=\; \frac{P_0 + P_1 + P_2}{3} \tag{7}$$

where P_0, P_1, and P_2 are functions of the number of one-hop neighbors, the number of two-hop neighbors, and a set of two-hop neighbors that can only be reached through a particular one-hop neighbor [29].

In addition, in order to further reduce the number of rebroadcast vehicles, each rebroadcast vehicle that has the same forwarding probability $\overline{\phi}$ will be assigned a different delay, which is computed from

$$\Delta(t) \;=\; \Delta(t)_{\max} \times (1 - \overline{\phi}) + \delta \tag{8}$$

where $\Delta(t)_{\max}$ is a maximum delay time and δ is a random variable which takes values on the order of milli-seconds. The performance of OAPB is compared with that of the Deterministic Broadcast (DB) protocol, where each vehicle rebroadcasts with a fixed probability. It is shown that OAPB outperforms DB in terms of broadcast overheads and broadcast delivery ratio. This is clearly due to the fact that nodes in OAPB are allowed to adjust the forwarding probability based on the network characteristics.

3) AutoCast: In AutoCast [30], the rebroadcast probability is calculated from the number of neighbors around the vehicle. In particular, the probability is obtained from the following function

$$p \;=\; \frac{2}{N_h \times 0.4} \tag{9}$$

where N_h is the number of one-hop neighbors. With this probability assignment function, the rebroadcast probability decreases as the number of neighbors increases. Obviously, this function only works when the number of neighbors, N_h, is greater than or equal to 5. However, it is not clearly specified in [30] how the probability is assigned in the cases where $N_h < 5$.

Due to the nature of probabilistic flooding, a packet may not always be able to reach the distant vehicles because some vehicles may decide not to forward it. In order to increase coverage and reachability, in this protocol, a packet is also rebroadcasted periodically. The rebroadcast interval is also adjusted dynamically based on the following function

$$t = N_h/\alpha \tag{10}$$

where α is a constant specifying the desired number of broadcast packets per second. The performance of AutoCast is also compared with pure flooding and the other protocols called MILE and MILE-on-demand [33]. MILE is simply a periodic broadcast protocol where a node broadcasts the received data periodically. On the other hand, MILE-on-demand is an improved version of MILE where each node periodically broadcasts small metadata units instead of a complete data unit in each broadcast cycle. In addition, a node can also request any missing data on demand. This reduces the amount of data that need to be transmitted and updated in each cycle at the cost of additional delay. It is shown that AutoCast outperforms these protocols in terms of delivery ratio and dissemination speed. The reason is that AutoCast also takes density (i.e., in terms of number of neighbors) into consideration when assigning the rebroadcast probability. This helps in reducing packet collision and thus increases the delivery ratio.

4) Irresponsible Forwarding (IF): The IF protocol [31], [34] assigns the forwarding probability based on the distance between the vehicle and the transmitter as well as the vehicle density. In IF, when a vehicle receives a packet from a transmitter, it computes its own forwarding probability according to the following function

$$p = e^{-\frac{\rho_{\mathrm{s}}(z-d)}{c}} \tag{11}$$

where ρ_{s} is the vehicle density, z is the transmission range, d is the distance between the vehicle and the transmitter, and $c \geq 1$ is a coefficient which can be selected to shape the rebroadcast probability. Basically, the higher the value of c is, the higher will be the rebroadcast probability. Note that the forwarding probability given in (11) is different from the conventional forwarding probability assignment function, where the probability is normally a linear function of the distance. Based on this probability assignment function, the rebroadcast probability increases as the distance between the vehicle and the transmitter increases. In addition, the rebroadcast probability decreases as the network becomes denser, which is a desirable property.

It is shown in [31] that the number of rebroadcast packets can be controlled by adjusting the shaping parameter c. In addition, the IF protocol is able to keep the expected number of rebroadcast packets at a constant level even when the vehicle density increases. Thus, it scales with the network density.

C. Network Coding-Based Multi-hop Broadcasting

Recently, network coding has caught attentions of many researchers in the field of ad hoc wireless communications. Network coding is a new way of information dissemination which is expected to yield a much higher throughput than the traditional way of transmission [35]. A good overview of network coding concept can be found in [36]. The idea of network coding and its distinction from the traditional transmission approach can best be described through the following classical example [37]. Consider a simple wireless network shown in Fig. 2.a, where node C is an intermediate node between node A and node B. In addition, in this considered scenario, node A and node B are not directly connected. Suppose that A has a packet destined to B, and B also has a packet destined to A. In a traditional packet transmission approach, A will need to send its packet to C and then let C forward the packet further to B. Similarly, B will need to send its packet to A via C. Note that the total number of transmissions required to complete the packet exchanges in this case is equal to four, as illustrated in Fig. 2.a. In contrast, using network coding, the packet exchanges between node A and node B can be accomplished in a fewer number of transmissions as illustrated in Fig. 2.b. First, A transmits its packet to C. Second, B transmits its packet to C. Next, node C creates an encoded packet by XORing the packets it received from A and B together, and then it broadcasts the XORed packet to both A and B. Finally, node A and node B each can decode the packet sent to them by XORing the received packet with their own packet. Note that this process requires only three transmissions, which is fewer than that of the traditional approach.

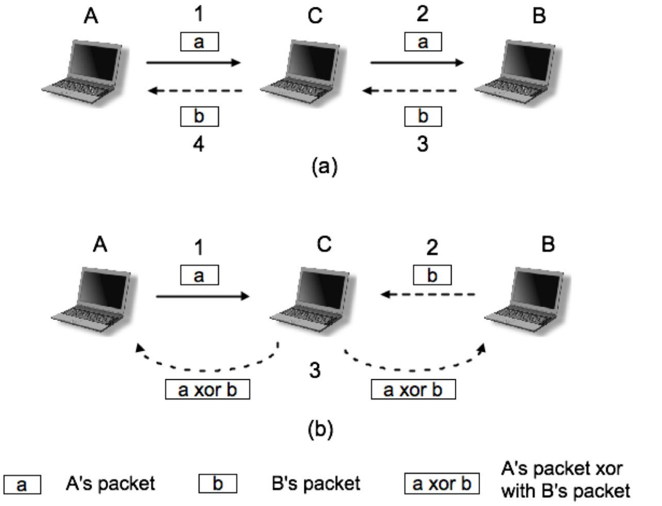

Fig. 2. (a) Example of traditional transmission (b) Example of network coding-based transmission.

As illustrated in the above example, disseminating information with network coding potentially requires a fewer number of packet transmissions. This helps utilize the bandwidth more efficiently. Although there are a number of work which investigate how to apply the concept of network coding to broadcasting in mobile ad hoc wireless networks, there are not many network coding-based broadcasting protocols designed specifically for VANETs. However, since the concept may be adapted for VANETs, we briefly mention a few of network coding-based protocols here.

1) COPE: A network coding-based protocol, called COPE, is introduced in [37]. Although COPE is a unicast routing protocol, it is a foundation which many protocols build upon, and thus it is worth mentioning here. The objective of COPE is to extend and realize the benefit of network coding beyond the simple duplex flows example discussed in Fig. 2. The operation of COPE is based on three key techniques: (i) opportunistic listening, (ii) opportunistic coding, and (iii) neighbor state learning. Opportunistic listening simply lets nodes take advantages of the wireless broadcast medium by snooping all the packets that they overhear. Each node will store the overheard packets in its buffer for a limited period. These packets will later be used for coding when the opportunity presents. The second technique, opportunistic coding, defines some ground rules for a node to encode and transmit a packet. Basically, a node should ensure that its next hop neighbor has enough information to decode the transmitted encoded packet. Generally, a node will be able to correctly decode a packet p_i from an encoded packet created from packets $p_1, p_2, \ldots, p_n$ if it has $n-1$ of these packets. Thus, learning what packets its neighbors have is crucial, and this is achieved through a periodic broadcast of reception reports. Basically, each node periodically announces to its neighbors which packets are stored in its reception buffer.

The concepts of network coding is extended to apply more specifically for VANETs in [38]. Particularly, a new protocol called Local-directed Network Coding (LDNC) is introduced. The main idea is similar to that of COPE; however, LDNC exploits the directions of packet propagation in VANETs. In LDNC, all packets incoming to each node are classified and placed into two separate queues based on their propagation directions: (i) forward propagation and (ii) backward propagation. From each node's perspective, packets in the forward propagation direction are those to be relayed to the vehicles in front whereas packets in the backward propagation direction are those to be relayed to the vehicles behind it. In addition, each node looks for an opportunity to encode the packets by XORing the packets from these two queues. This creates a setting which is similar to the example given in Fig. 2. However, LDNC is still a unicast routing protocol.

2) CODEB: CODEB is a network coding-based broadcasting protocol introduced in [39]. It extends the concepts and techniques used in COPE to cover broadcasting scenarios in ad hoc wireless networks. Similar to COPE, it relies on opportunistic listening, where each node has to snoop all packets that it overhears. Moreover, each node also periodically broadcasts a list of its one-hop neighbors. This allows a node to build a graph of its two-hop neighbors, which will further be used to construct a broadcasting backbone. In addition, CODEB also relies on opportunistic coding, which determines if a node can exploit coding opportunities to transmit coded packets to its neighbors. It is pointed out that opportunistic coding for broadcast is quite different from coding for unicast. In unicast routing, only the intended next-hop node needs to receive a given packet whereas in broadcasting all the neighbors of the node must receive the packet. This adds another level of complexity as a node must ensure that all of its neighbors are able to decode the packet.

Unlike a probabilistic-based broadcasting protocol where the next neighbors to forward the packet are selected randomly, CODEB selects a subset of neighbors to forward the packet deterministically. Particularly, each node creates a forwarder list using a Partial Dominant Pruning (PDP) algorithm [40]. A forwarder list of a node contains the minimum number of broadcast nodes such that all nodes in its two-hop neighborhood are covered. Only nodes in the forwarder list are allowed to forward the packet. However, a forwarder may decide not to transmit the packet if it determines that all of its neighbors have already received the given packet. The performance of CODEB is evaluated by simulation. It is shown that CODEB outperforms a scheme that only uses PDP without network coding in terms of packet delivery ratio and the number of transmissions required to deliver a packet to all nodes in the network. This is due to the fact that network coding can save the number of transmissions required to broadcast a packet at the expense of coding and decoding operations.

3) Efficient Broadcasting Using Network Coding and Directional Antennas (EBCD): EBCD is a network coding-based broadcasting protocol that combines the advantage of network coding with that of directional antennas [41]. Similar to CODEB [39], EBCD determines a subset of neighboring nodes which will be performing a forwarding task deterministically. However, EBCD uses a different algorithm called Dynamic Directional Connected Dominating Set (DDCDS). Ultimately, the DDCDS algorithm constructs a directional virtual network backbone where each node determines both its forwarding status as well as the outgoing edges (antenna sectors) in which the packets will be transmitted. Another main difference between EBCD and CODEB is that, in EBCD, network coding is applied in each sector of the directional antennas around the node instead of omnidirectional. It is shown in [41] that using both directional antennas and network coding provides a significant improvement, in terms of number of transmissions, over a scheme that uses only network coding and a scheme that uses none of these two techniques.

4) DifCode: The goal of DifCode is to reduce the number of transmissions required to flood packets in an ad hoc wireless network [42]. Similar to CODEB, DifCode selects the next forwarding nodes deterministically. However, the selection algorithm used in DifCode is based on multi-point relay (MPR) [43]. An MPR of a node is defined as a set of its one-hop neighbor that covers its two-hop neighborhood. In DifCode, each node in the network encodes and broadcasts only the packets that are received from nodes that select it as their MPR.

A main distinction between DifCode and CODEB is in the opportunistic coding techniques. In CODEB, all the neighbors of a transmitter are required to be able to decode the received packets immediately. This limits coding opportunities. In DifCode, this constraint is relaxed by allowing nodes to buffer packets that are not immediately decodable. Particularly, each node will maintain buffers for keeping three types of packets: (i) successfully decoded packets, (ii) not immediately decodable packets, and (iii) packets that need to be encoded and broadcasted further. Classifying packets into these three types allows DifCode to exploit *background decoding*, which is a process where a node XORs an incoming encoded packet with those that are stored in the type-ii buffer in order to decode it. If decoding results in a native (uncoded) packet, then it is moved to the type-i buffer. This improves an opportunity of being able to decode the packets. It is shown in [42] that DifCode is able to achieve a lower redundancy rate than a probabilistic network coding protocol, where the next forwarding nodes are chosen randomly.

III. Single-hop Broadcasting

As opposed to multi-hop broadcasting, in single-hop broadcasting a packet broadcasted by a vehicle will not be flooded in the network. When a vehicle receives a packet, it does not rebroadcast the information immediately. Instead, a vehicle updates its information database according to what it learns from the received packet, provided that the information in the packet is considered newer than that in its database. Periodically, a vehicle disseminates some of the information in its database to the other vehicles in the network. The design choices that need to be considered in this type of protocols are: (i) broadcast interval and (ii) information that needs to be broadcasted. To reduce redundancy and keep the most up-to-date information, the broadcast interval should be set appropriately (i.e., not too long and not too short). In addition, only the important and relevant information should be selected to broadcast. We divide the single-hop broadcasting protocols into two categories, which are the fixed broadcast interval protocols and the adaptive broadcast interval protocols. While the main focus of the fixed broadcast interval protocols is only on the selection and aggregation of information, an adjustment of broadcast intervals is also taken into consideration in the adaptive broadcast interval protocols.

A. Fixed Broadcast Interval

1) TrafficInfo: In TrafficInfo [44], each vehicle in the network periodically broadcasts the traffic information stored in its database. A particular type of traffic information reported is the travel times on the road segments. The broadcasting process is done as follows. It is assumed that each vehicle has a digital map of the road network, and each road segment has a unique ID number. In addition, each vehicle is equipped with a global positioning system (GPS) receiver so it knows its own position on the digital map at any time. When a vehicle travels through a road segment, its travel time is recorded and kept in its on-board database. In addition to the travel time on its current road segment, a vehicle also learns about the travel time on the other road segments from other vehicles when they report the travel times on their road segments.

As in any single-hop broadcasting scheme, it is inefficient to broadcast all the records in the vehicle's database. TrafficInfo has a way to select the most relevant information to broadcast. In order to use the bandwidth efficiently, only the top k most relevant information in the database will be broadcasted by the vehicle. The relevance of the information is determined by a ranking algorithm, which is based on the current location of the vehicle and the current time. For example, if the vehicle is on segment A, the information about the road segment B which is adjacent to segment A is more relevant than the information about the road segment C where C is ten blocks away from segment A. Basically, the relevance of information decreases with distance and time. A formal way of ranking by quantifying the relevance of information in terms of demand and supply is also presented in [44].

2) TrafficView: TrafficView is a single-hop broadcasting scheme [45] designed for enabling an exchange of traffic information among vehicles. The types of information exchanged among the vehicles are speed and position. In this scheme, when a vehicle receives a broadcast packet, it stores the information in its database. The information is then rebroadcasted in the next broadcast cycle. However, instead of broadcasting every record in its database, the vehicle aggregates the speed and positions of many vehicles into a single record and then broadcast this aggregated information. Two aggregation algorithms, namely the ratio-based algorithm and the cost-based algorithm, are presented. In the ratio-based algorithm, a road will be divided into small regions, and an aggregation ratio will be assigned to each region. An aggregation ratio in each region is assigned based on the importance of the region and the level of information accuracy required for that region. A region which is assigned a small value of aggregation ratio will have information with less accuracy. On the other hand, the cost-based algorithm also takes the cost of aggregating the records into consideration. The aggregation cost can be regarded as the loss of accuracy incurred from combining the records. The performance of these algorithms are evaluated and compared by simulation. It is shown that although the cost-based algorithm yields better accuracy, the ratio-based algorithm gives more flexibility. The main reason why the cost-based algorithm yields better accuracy than the ratio-based algorithm is that it selects records with the lowest loss of accuracy (i.e., minimum cost) to aggregate whereas the ratio-based algorithm simply aggregates records blindly.

B. Adaptive Broadcast Interval

1) Collision Ratio Control Protocol (CRCP): In CRCP [46], each vehicle disseminates the traffic information periodically. The traffic information in this case are the location, speed, and road ID. It is assumed that these data can be measured at every second. In this protocol, a mechanism for dynamically changing a broadcast interval based on the number of packet collisions is proposed. Basically, the protocol aims at keeping the collision ratio at a targeted level regardless of the vehicle density. Intuitively, the number of packet collisions increases as the network density increases. Thus, in order to keep the number of packet collisions at the desired level, vehicles need to adaptively adjust their broadcast intervals. In particular, in this protocol, the broadcast interval will be doubled if the estimated collision ratio observed by a vehicle and the estimated bandwidth efficiency are greater than the pre-defined thresholds. Otherwise, the broadcast interval is shorten by one second.

In addition to the broadcast interval adjustment mechanism, three methods for selecting the data to be disseminated are also proposed: (i) Random Selection (RS), (ii) Vicinity Priority Selection (VPS), and (iii) Vicinity Priority Selection with Queries (VPSQ). In the RS scheme, a vehicle randomly chooses the information in its database to disseminate, whereas in the VPS and VPSQ schemes the information of the nearby road segments (e.g., within a distance of x km from the vehicle) will be given the priority in being selected for dissemination. In other words, a certain amount of traffic data in the nearby area will be selected for transmission first. If, however, there is still room, the data in the distant area will then be selected. The main difference between VPS and VPSQ is that, in VPSQ, a node also has an option of querying the traffic information in the area of interest rather than only passively collecting them. The performance of the CRCP protocols that use each of these three data selection schemes are evaluated via simulation. It is shown that the CRCP protocol which uses VPSQ as its data selection scheme generally performs better than the cases where it uses the other two data selection schemes. This is due to the ability of VPSQ to query information in the relevant areas.

2) Abiding Geocast: The Abiding Geocast protocol [47] is designed for disseminating safety warnings within an effective area where these warnings are still relevant and applicable. In this scheme, when an emergency situation occurs, the first vehicle that detects it starts broadcasting a warning packet. In the packet, an effective area where the warning is still relevant and should be kept alive is also specified. When another vehicle receives the warning packet, it will become an active relay node and keep broadcasting the warning packet as long as it is still traveling in the effective area. The vehicle stops broadcasting when it goes outside of the effective region.

In order to keep the number of redundant warning packets at minimum, each vehicle adjusts its rebroadcast interval dynamically. The rebroadcast interval is determined from the transmission range, speed, and the relative distance between the vehicle and the emergency site. Basically, the rebroadcast interval increases as the distance between the active relay node and the emergency site increases. In addition, the rebroadcast interval also increases as the vehicle speed decreases. The performance of the protocol is evaluated in terms of the number of overheads via simulation. However, a comparison with other protocols is not given.

3) Segment-oriented Data Abstraction and Dissemination (SODAD): In this protocol [48], roads are divided into segments of known length. Each vehicle collects the data on its current segment either by sensing the information itself or observing what the other vehicles report. In order to reduce the number of redundant rebroadcast packets, each vehicle adaptively adjusts its broadcast interval. Particularly, a vehicle adjusts its transmission interval based on the information it receives from the other vehicles. In fact, the information received will be characterized as one of these two events: (i) provocation and (ii) mollification. A provocation event is defined as an event that will reduce the time until the next broadcast, whereas a mollification event is defined as an event that will increase the time until the next broadcast. When a vehicle receives a packet, it will determine whether a provocation or a mollification event has occurred. This is done by assigning a weight to the received packet. A weight is computed from the discrepancy between the received data and those in the vehicle's knowledge database. If the newly received information is considered newer than that in the database, then the assigned weight will be high. Based on the weight of the packet, a node determines whether a provocation or mollification event has occurred by comparing it to a threshold. The time until the next rebroadcast is increased or decreased based on the weight. In this study, the performance of the interval adaptation scheme is compared with the static scheme in both simulation and in a prototype system. The results confirm that using the adaptive scheme can reduce the number of packet collisions caused by the static periodic broadcast scheme.

IV. Performance Evaluation

While it is possible to discuss the performance of the broadcasting protocols qualitatively as shown in Table I, it is quite challenging to quantitatively compare their performance based on the results reported in the literature. One of the main reasons is that these protocols are often evaluated by different performance metrics. Moreover, most of the time they are judiciously evaluated because only the metrics which are in favor of their performance are presented while the others are ignored. Thus, based on the results reported in their original papers alone, it is not possible to make a quantitative comparison among them effectively. Instead of attempting to judge each protocol based on biased information, we will discuss the metrics commonly used in the current literature and then suggest a new unified metric that is capable of giving a protocol a fair overall evaluation.

A. Existing Metrics

Essentially, what most researchers are interested to know about the broadcasting protocols in VANETs are: (i) how *frequently* the information packets are duplicated, (ii) how *far* the information packets can propagate, and (iii) how *fast* the information can be spread. The existing metrics commonly used in the literature are listed in Table II. The first column of Table II indicates the domain where each metric belongs. In our perspective, these metrics can be classified into four domains, namely *frequency*, *space*, *time*, and *mixed*. The metrics in the frequency domain are those related to frequency counting (e.g., counting of packets or the number of vehicles). The metrics in the space domain involve the measurements of distance whereas those in the time domain involve the measurements of time. The metrics in the mixed domain are those created from a combination of metrics in more than one domain. The second column indicates the ID that we assign to each metric, which will later be used for referencing purpose. The third column of Table II lists the metric names. It can be observed that some of the metrics are very similar although they are called differently. The fourth column describes how each metric is computed while the fifth column describes what each metric is designed to measure. The sixth column specifies the unit of each metric. Finally, the last column suggests the favorable value for each metric (i.e., indicating whether a low value or a high value of the considered metric is desirable).

In the frequency domain, the following three attributes of broadcasting protocols are usually of interest, and the metrics in this domain are designed to quantify them.

- **Redundancy**—Redundancy is a key performance indication of a broadcasting protocol. A good protocol should be able to disseminate information with the least amount of redundancy or overheads. As listed in Table II, the metrics used for quantifying the redundancy are *redundancy rate*, *load generated per broadcast packet*, *forward*

TABLE I
QUALITATIVE COMPARISON OF BROADCASTING PROTOCOLS FOR VANETS.

Broadcasting Type	**General Characteristics**	**Advantages**	**Disadvantages**
Multi-hop	• Packets are disseminated by ways of smart flooding • Reduce redundancy by varying broadcast probability or delay	• Packets can be disseminated quickly • Good for safety alerts and emergency warning applications • No need for large storage space to keep unbroadcast packets	• Need an algorithm to deal with the broadcast storm problem • No packet persistency
Single-hop	• Packets are disseminated by ways of periodic broadcast • No packet flooding • Reduce redundancy by varying the broadcast period of each node • Rely on node mobility to carry and spread the information	• Good for applications that need packet persistency • No broadcast storm	• Packet dissemination speed may be slow • Not suitable for delay-sensitive applications • May require large storage space to keep unbroadcast information

TABLE II
EXISTING PERFORMANCE METRICS FOR AN EVALUATION OF BROADCASTING PROTOCOLS.

Domain	**ID**	**Metric Name**	**Mathematical Definition**	**Description**	**Unit**	**Favorable Value**
Frequency	1	Redundancy rate	$\frac{\text{no. of duplicate packets}}{\text{no. of source packets}}$	Measure the number of duplicate packets per one source packet	unit-less	Low
	2	Load generated per broadcast packet	$\frac{\text{no. of bits transmitted}}{\text{no. of source packets}}$	Measure the total number of bits used in broadcasting one source packet	bit/pkt	Low
	3	Forward node ratio	$\frac{\text{no. of vehicles forwarding the packet}}{\text{no. of vehicles in the network}}$	Measure the proportion of vehicles in the network that rebroadcast the source packet	unit-less	Low
	4	Link load	$\frac{\text{no. of broadcast bits received by a vehicle}}{\text{observation period}}$	Measure amount of broadcast traffic received at each vehicle over unit time	bit/s	Low
	5	Broadcast overhead	$\frac{\text{no. of duplicate packets received by a vehicle}}{\text{no. of vehicles in a defined zone}}$	Measure the number of packets collectively duplicated in a defined area	pkt/veh	Low
	6	Delivery ratio, Success ratio	$\frac{\text{no. of vehicles successfully receiving packets}}{\text{no. of vehicles in the network}}$	Measure the proportion of vehicles that successfully receive the broadcast packets	unit-less	High
	7	Reception rate, Reachability	$\frac{\text{no. vehicles receiving the broadcast packets}}{\text{no. of vehicles reachable by pure flooding}}$	Compare the reachabiltiy of a broadcasting protocol to that of the pure flooding protocol	unit-less	High
	8	Saved rebroadcast	$\frac{\text{no. receiving host - no. transmitting hosts}}{\text{no. hosts receiving packet}}$	Measure the number of saved rebroadcast packets	unit-less	High
	9	Collision ratio, Packet loss ratio	$\frac{\text{no. of collision packets}}{\text{no. of transmitted packets}}$	Measure the rate at which the collision occurs	unit-less	Low
Space	10	Propagation distance	| Packet last position - Packet initial position |	Measure the distance between the origin of the packet and the point where it is last received	m	High
	11	Forward progress, One-hop progress	| Position of next rebroadcast vehicle - Position of current transmitter |	Measure the additional distance covered by the packet when it is rebroadcasted	m	High
	12	Number of hop propagated	Last hop the packet is received - Packet origin	Measure the number of hops that the packet can traverse	hops	High
	13	Sustainable number of hops	Last hop the packet is received with required QoS - Packet origin	Measure the number of hops that the packet can traverse with the desired quality	hops	High
Time	14	Propagation time, End to end delay	The instant the packet is received at a specific point - The instant the packet is originated	Measure the time it takes a packet to traverse from a source to a specific point in the network	s	Low
	15	Rebroadcast latency	The instant the packet is received by the next vehicle - The instant the packet is broadcasted by current vehicle	Measure the time until the packet is received successfully by the next vehicle	s	Low
Mixed	16	Propagation speed, Dissemination speed	$\frac{\text{Packet propagation distance}}{\text{Propagation delay}}$	Measure the rate at which the packet can propagate per unit time	m/s	High

TABLE III
BROADCASTING PROTOCOLS IN VANETS AND THE METRIC DOMAINS USED FOR EVALUATION.

Broadcasting Type	Protocols	Evaluation Models	Simulation Platforms	Metric Domains			
				Frequency	**Space**	**Time**	**Mixed**
Multi-hop	UMB [16]	Simulation	MATLAB [49], CSIM [50]	2, 6			16
	SB [18]	Analysis & Simulation	MATLAB		11	15	16
	EDB [19]	Simulation	Proprietary	3, 6			
	Slotted 1-Persistence [20]	Simulation	OPNET [51]	4, 7, 9	12	14	
	Weighted p-Persistence [20]	Simulation	OPNET	4, 7, 9	12	14	
	RMDSI [21]	Simulation	NS-2 [52]	6		14	
	MHVB [22]	Simulation	NS-2	6, 9			
	RBLSM [23]	Simulation	MATLAB			15	
	VDEB [24]	Simulation	NS-2	1		14	
	LDMB [26]	Simulation	Unspecified	6		14	
	OAPB [29]	Simulation	NS-2	5, 6		14	
	AutoCast [30]	Simulation	NS-2	6			16
	IF [31], [34]	Analysis & Simulation	MATLAB, NS-2	1, 7, 8		14	
	CODEB [39]	Simulation	NS-2	6			
	EBCD [41]	Simulation	NS-2	1, 6			
	DifCode [42]	Simulation	OPNET	1			
Single-hop	TrafficInfo [44]	Simulation	STRAW/SWANS [53]	6			
	TrafficView [45]	Simulation	NS-2		10		
	CRCP [46]	Simulation	NETSTREAM [54]	9			
	Abiding Geocast [47]	Simulation	OMNeT++ [55]	5			
	SODAD [48]	Simulation	NS-2	9		14	

node ratio, *link load*, and *broadcast overhead*. Generally, these metrics measure the number of duplicate packets or the number of duplicate bits used in disseminating one information packet.

- **Reachability**—An information should be disseminated in such a way that it reaches all the reachable nodes in the target area. Keeping the redundancy low while maintaining high reachability is one of the main challenges in designing a broadcasting protocol. The metrics used for quantifying the reachability are *delivery ratio* and *reception rate*. As observed from Table II, these two metrics are defined a bit differently. Basically, the delivery ratio is derived from the total number of vehicles in the network whereas the reception rate is derived from the number of reachable nodes. In other words, the reception rate measures the proportion of "connected nodes" that receives the broadcast packet.
- **Failure rate**—If the rebroadcast mechanism of a broadcasting protocol is not designed carefully, there could be a lot of packet collisions since many vehicles in the same vicinity may rebroadcast the packets at the same time. This is usually referred to as the broadcast storm problem [15]. Obviously, the failure rate should be kept at minimal. The metric commonly used in quantifying the failure rate is *collision ratio* or *packet loss ratio*.

The metrics in the space domain typically measure how far a packet can propagate. *Propagation distance* measures a distance that a packet can propagate from the point where it is originated in unit of meters whereas the *number of hops propagated* measures how far a packet can traverse in terms of the number of hops. *Sustainable number of hops*, on the other hand, measures the number of hops that a packet can traverse while maintaining a desired quality of service (QoS), for example, in terms of bit error rate [56]. In addition to the total propagation distance, a progress made at each hop is also an important quantity, and this is typically measured by *forward progress*. Basically, the forward progress measures the distance gained beyond the current transmitter if a particular vehicle was selected as a next rebroadcast node. Normally, a vehicle with the largest forward progress will be selected.

In the time domain, *propagation time* measures a time it takes a packet to traverse from a source to the other point in the network. This includes the "air time," which is the delay incurred from passing a packet from one vehicle to the other via the wireless communication channel, and the "ground time," which is the delay incurred while a vehicle is carrying the packet before rebroadcasting it to the other vehicles. *Rebroadcast latency* measures a time until the packet is received successfully by the next vehicle.

Finally, the metrics in the mixed domain are those created from a combination of the metrics in the frequency, space, and time domains. Based on our survey, the only metric in the mixed domain currently defined in the literature is *propagation speed* or *dissemination speed*. Basically, it measures the distance at which a packet can traverse the network per unit time.

B. Suggested Metric

As previously discussed, one of the main problems in comparing the performance of broadcasting protocols is that researchers often use different metrics in assessing the performance of their protocols. Moreover, most of the times only the metrics which are advantageous to the proposed protocols are selectively presented while other metrics are ignored. In order to illustrate this point more clearly, we list the broadcasting protocols and the metric domains used in evaluating their performance in Table III. In each row of the table, the metrics used in evaluating each broadcasting protocol are indicated by numbers. These numbers, appeared in the last four columns of Table III, correspond to the IDs of the metrics listed in

Table II. For instance, 1 would correspond to redundancy rate, and 16 would correspond to propagation speed. It can be observed that most of the protocols are evaluated by the metrics in only one or two domains. In other words, most protocols are not given a complete evaluation from all the angles.

In order to evaluate the performance of each protocol fairly, the metrics from the three independent domains (i.e., frequency, space, and time) should be considered. In this regard, we introduce a new metric called *Dissemination Efficiency (DE)*, which is defined as

$$\mathrm{DE} = \frac{\text{Propagation Distance} \times \text{Success Ratio}}{\text{Propagation Time} \times \text{Redundancy Rate}}. \quad (12)$$

This is a simple, yet intuitive, performance metric. Dissemination Efficiency combines the metrics in all the three independent domains; hence, it can truly reflect the overall performance of a broadcasting protocol. The propagation distance is measured in meters, the propagation time is measured in seconds, the redundancy rate and the success ratio are unitless; therefore, DE has a unit of m/s. Intuitively, it measures how far an information packet can propagate through the network per unit time and per the amount of overheads generated. In addition, it is also weighted by the success ratio, which measures the proportion of nodes that successfully receives the broadcast packet. As a result, effects of packet collision or failures are also well captured by DE. Ultimately, the DE value increases if the information can be distributed farther, faster, with high success rate, and with less redundancy.

In order to demonstrate how DE can be used to compare the performance of the broadcasting protocols for VANETs, we perform a simple simulation of a few multi-hop broadcasting protocols. Particularly, the Weighted p-persistence protocol, the IF protocol, and the Slotted 1-Persistence protocol are investigated. The Weighted p-persistence protocol and the IF protocol are selected to illustrate the effects of rebroadcast probability assignment functions on the protocol performance. In addition, the Slotted 1-persistence protocol, a delay-based broadcasting scheme, is also selected for a comparison with its counterpart protocol (i.e., the Weighted p-persistence protocol). The simulation is implemented in MATLAB by following the approach described in [31]. In each simulation trial, N vehicles are placed on a straight line of length L according to a Poisson point distribution with density $\rho_s = N/L$. We assume that $L = 10$ km in all the simulation scenarios. The transmission range of each vehicle is assumed to be 200 m. The first vehicle is designated as a source, and transmits one packet. After the source transmits a packet, each vehicle within the source's transmission range decides to rebroadcast the packet according to the protocol in consideration. In the Weighted p-persistent protocol, each vehicle rebroadcasts the packet with the probability given in (6). In IF, each vehicle rebroadcasts the packet according to the rebroadcast probability given in (11) with $c = 5$. In the Slotted 1-persistence protocol, each vehicle rebroadcasts the packet according to its designated time slot, where the slot number is calculated from (4) and (5). The number of time slots, N_s, is fixed at 10 slots, and the length of each time slot, τ, is 1 ms. In all the schemes, it is assumed that a node will not rebroadcast if it hears a duplicate broadcast from the nodes downstream. The rebroadcasting process continues until the packet dies (i.e., no vehicle rebroadcasts the packet further) or until the packet reaches the last vehicle in the network, whichever occurs first. It is assumed that the packet transmission time is 1 second. The DE value is collected at the end of each simulation trial. The simulation is repeated for 10,000 trials and the average DE value is calculated.

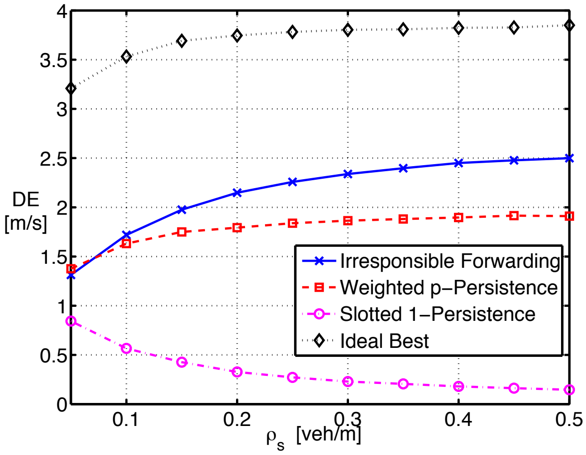

Fig. 3. Dissemination efficiency as a function of vehicle density.

In Fig. 3, the average DE value is shown as a function of the vehicle density. In addition to the three multi-hop broadcasting protocols mentioned above, we also investigate the performance of an ideal best scheme where the packet is always rebroadcasted by the farthest vehicle in the range of the transmitter in each hop and the only delay incurred from rebroadcasting in each hop is from the packet transmission time. It can be observed from Fig. 3 that the DE value of the Slotted 1-persistence protocol decreases as the vehicle density increases. This is due to the fact that when the network becomes denser, the number of vehicles that will select the same time slot for transmission will increase. As a result, the number of packet collisions will increase. This certainly reduces the DE value. In contrast, it can be observed that in the cases of IF and Weighted p-persistence protocols, the DE values generally increase as the vehicle density increases. The main reason is that these two protocols are probabilistic-based. Consequently, they are able to further reduce the number of duplicate packets as well as the number of collisions. Moreover, it can be observed that the IF protocol is able to achieve a higher DE value than the Weighted p-persistence protocol. This is expected because the rebroadcast probability assignment function in the IF protocol takes the vehicle density into consideration whereas the probability assignment function in the Weighted p-persistence protocol does not. This makes the IF protocol more efficient than the Weighted p-persistence protocol.

V. Summary and Open Issues

In this paper, we have reviewed a variety of broadcasting techniques for information dissemination in the self-organizing vehicular ad hoc networks. A classification of these protocols is illustrated in Fig. 1. The common focus in designing these protocols is in suppressing the excessive rebroadcast packets. In multi-hop broadcasting protocols, the reduction of redundant rebroadcast packets is typically done through the delay and probability assignment functions, which adjust the waiting delay and the rebroadcast probability based on the vehicle location and the physical characteristics of the network such as the vehicle density. The number of packet transmissions can also be reduced by using a network coding approach. In single-hop broadcasting protocols, where each vehicle rebroadcasts the packet periodically, the suppression of excessive rebroadcast packets is usually done by letting each vehicle adjust its rebroadcast interval dynamically.

Based on the information reported in the literature alone, it is quite difficult to quantitatively compare the performance of the existing broadcasting protocols. One of the main reasons is that these protocols are often evaluated by different metrics, and there has not been a unified metric that can capture the overall performance of each protocol. In this regard, we introduce a new metric called Dissemination Efficiency, which is a combination of the metrics in all the independent domains, and hence it is able to reflect the overall performance of a broadcasting protocol. Basically, the value of Dissemination Efficiency increases if a packet can be broadcasted farther, faster, with high success rate, and with less redundancy.

Although there are already many broadcasting protocols proposed for VANETs, there are still open issues that need to be considered:

- **Theoretical fundamental performance limits**—It is evident from Table III, that there are only a few theoretical analyses of the broadcasting protocols. Most of the protocols are evaluated only by simulation. This makes it difficult to determine the fundamental performance limits of the broadcasting protocols for VANETs. A general theoretical framework for analyzing the performance of these protocols is worth developing. The theoretical analytical framework should be able to model the two main parts that significantly affect the protocol performance. First, it should be able to model the vehicle movement which will in turn affect the topology and network connectivity. The vehicle movement has been thoroughly studied in the field of transportation science. Examples of the widely adopted mobility models are the car-following model [57] and the cellular automata model [58]. These mobility models should be parts of the theoretical analytical framework.

  Second, the dynamics of packet forwarding should be modeled in the theoretical framework. How a packet is forwarded or rebroadcasted depends on the broadcasting protocol. In a probabilistic-based multi-hop broadcasting protocol, the dynamics of packet forwarding (e.g., how far it can traverse, how long it takes, etc.) could be derived from the rebroadcast probability assignment function. Similarly, the packet forwarding behavior in a delay-based multi-hop broadcasting protocol could be determined from the delay assignment function. In a single-hop broadcasting scheme, this could be derived from the broadcast interval adjustment function.
- **Assessment of the protocols in realistic scenarios**—Based on our survey, the broadcasting protocols reported in the literature are mostly tested on a simple straight road section. However, a situation where a dissemination of traffic information will likely be the most useful is in the urban scenario in which the road structures are fairly complex. As a result, the broadcasting protocols still need a thorough test under more complex interconnected road structures, where the network characteristics (e.g., vehicle density) on each section are likely to be interdependent. A realistic evaluation could be done either by a field experiment or a thorough simulation. There are tradeoffs between these two methods.

  Although the best way to determine which protocol really works in practice is to conduct a field experiment, it is still quite a challenge to carry out an experiment on a grand scale (i.e., involving a large number of vehicles). An alternative approach is to use an integrated simulator that is capable of realistically simulating the road network environment, the vehicle mobility, and the communication among vehicles. An example of such an integrated simulator is GrooveNet [59]. In this type of simulators, a map of a real road network can be imported as an input, creating a realistic road topology that vehicles will travel. In addition, GrooveNet also has a feature which enables a communication between a real GPS-equipped vehicle and a simulated vehicle, making it possible and scalable to test a protocol in a field experiment. However, GrooveNet is still not perfect and it still needs further development.

APPENDIX

List of Acronyms

CRCP	Collision Ratio Control Protocol
CTB	Clear-to-Broadcast
DB	Deterministic Broadcast
DDCDS	Dynamic Directional Connected Dominating Set
DE	Dissemination Efficiency
EBCD	Efficient Broadcasting Using Network Coding and Directional Antennas
EDB	Efficient Directional Broadcast
GeRaF	Geographic Random Forwarding
IF	Irresponsible Forwarding
LDMB	Link-based Distributed Multi-hop Broadcast
LDNC	Local-directed Network Coding
MHVB	Multi-hop Vehicular Broadcast
MPR	Multi-point Relay
OAPB	Optimized Adaptive Probabilistic Broadcast
PDP	Partial Dominant Pruning
RBLSM	Reliable Broadcasting of Life Safety Messages
RDB	Random Directional Broadcast
RMDSI	Reliable Method for Disseminating Safety Information
RS	Random Selection
RTB	Request-to-Broadcast
SB	Smart Broadcast
SDDB	Simple Distance-based Directional Broadcast
SODAD	Segment-oriented Data Abstraction and Dissemination
UMB	Urban Multi-hop Broadcast
VANET	Vehicular Ad Hoc Network
VDEB	Vehicle-density-based Emergency Broadcasting
VPS	Vicinity Priority Selection
VPSQ	Vicinity Priority Selection with Queries

References

[1] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *IEEE Veh. Technol. Mag.*, vol. 2, no. 2, pp. 12–22, Jun. 2007.

[2] J. Bernsen and D. Manivannan, "Unicast routing protocols for vehicular ad hoc networks: A critical comparison and classification," *Pervasive and Mobile Computing*, vol. 5, no. 1, pp. 1–18, Feb. 2009.

[3] C. Maihofer, "A survey of geocast routing protocols," *IEEE Commun. Surveys Tutorials*, vol. 6, no. 2, pp. 32–42, Second Quarter 2004.

[4] A. Festag, P. Papadimitratos, and T. Tielert, "Design and performance of secure geocast for vehicular communication," *IEEE Trans. Veh. Technol.*, vol. 5, no. 59, pp. 2456–2471, Jun. 2010.

[5] R. J. Hall, "An improved geocast for mobile ad hoc networks," *IEEE Trans. Mobile Comput.*, vol. 10, no. 2, pp. 254–266, Feb. 2011.

[6] L. Junhai, Y. Danxia, X. Liu, and F. Mingyu, "A survey of multicast routing protocols for mobile ad-hoc networks," *IEEE Commun. Surveys Tutorials*, vol. 11, no. 1, pp. 78–91, First Quarter 2009.

[7] O. Badarneh and M. Kadoch, "Multicast routing protocols in mobile ad hoc networks: A comparative survey and taxonomy," *EURASIP J. Wireless Comm. and Networking*, vol. 2009, pp. 1–42, Jan. 2009.

[8] A. Sebastian, M. Tang, Y. Feng, and M. Looi, "A multicast routing scheme for efficient safety message dissemination in VANET," in *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC)*, Sydney, Australia, Apr. 2010, pp. 1–6.

[9] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of peer-to-peer overlay network schemes," *IEEE Commun. Surveys Tutorials*, vol. 7, no. 2, pp. 72–93, Second Quarter 2005.

[10] L. Zhou, Y. Zhang, K. Song, W. Jing, and A. V. Vasilakos, "Distributed media services in P2P-based vehicular networks," *IEEE Trans. Veh. Technol.*, vol. 60, no. 2, pp. 692–703, Feb. 2011.

[11] U. Shevade, Y. Chen, L. Qiu, Y. Zhang, V. Chandar, M. K. Han, H. H. Song, and Y. Seung, "Enabling high-bandwidth vehicular content distribution," in *Proc. ACM Int'l Conf. on Emerging Networking Experiments and Technologies (CoNEXT)*, Philadelphia, PA, Nov. 2010, pp. 1–12.

[12] M. Guo, M. Ammar, and E. W. Zegura, "V3: A vehicle-to-vehicle live video streaming architecture," *Pervasive and Mobile Computing*, vol. 1, no. 4, pp. 404–424, Dec. 2005.

[13] F. Soldo, C. Casetti, C. Chiasserini, and P. Chaparro, "Streaming media distribution in VANETs," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, New Orleans, LA, Nov. 2008, pp. 1–6.

[14] Y. Chu and N. Huang, "Delivering of live video streaming for vehicular communication using peer-to-peer approach," in *Proc. IEEE Workshop on Mobile Networking for Vehicular Environments*, Anchorage, AK, May 2007, pp. 1–6.

[15] S. Ni, Y. Tseng, Y. Chen, and J. Sheu, "The broadcast storm problem in a mobile ad hoc network," in *Proc. ACM Int'l Conf. on Mobile Computing and Networking (MOBICOM)*, Seattle, WA, 1999, pp. 151–162.

[16] G. Korkmaz, E. Ekici, F. Özgüner, and Ü. Özgüner, "Urban multi-hop broadcast protocol for inter-vehicle communication systems," in *Proc. ACM Int'l Workshop on Vehicular Ad Hoc Networks. (VANET)*, Philadelphia, PA, Sep. 2004, pp. 76–85.

[17] G. Korkmaz, E. Ekici, and F. Özgüner, "An efficient fully ad-hoc multi-hop broadcast protocol for inter-vehicular communication systems," in *Proc. IEEE Int'l Conf. on Comm. (ICC)*, Istanbul, Turkey, Jun. 2006, pp. 423–428.

[18] E. Fasolo, A. Zanella, and M. Zorzi, "An effective broadcast scheme for alert message propagation in vehicular ad hoc networks," in *Proc. IEEE Int'l Conf. on Comm. (ICC)*, Istanbul, Turkey, Jun. 2006, pp. 3960–3965.

[19] D. Li, H. Huang, X. Li, M. Li, and F. Tang, "A distance-based directional broadcast protocol for urban vehicular ad hoc network," in *Proc. IEEE Int'l Conf. on Wireless Comm., Networking and Mobile Computing (WiCom)*, Shanghai, China, Sep. 2007, pp. 1520–1523.

[20] N. Wisitpongphan, O. K. Tonguz, J. S. Parikh, P. Mudalige, F. Bai, and V. Sadekar, "Broadcast storm mitigation techniques in vehicular ad hoc networks," *IEEE Wireless Commun.*, vol. 14, no. 6, pp. 84–94, Dec. 2007.

[21] S. Khakbaz and M. Fathy, "A reliable method for disseminating safety information in vehicular ad hoc networks considering fragmentation problem," in *Proc. IEEE Int'l Conf. on Wireless and Mobile Communications (ICWMC)*, Athens, Greece, Jul. 2008, pp. 25–30.

[22] T. Osafune, L. Lin, and M. Lenardi, "Multi-hop vehicular broadcast (MHVB)," in *Proc. IEEE Int'l Conf. on ITS Telecomm. (ITST)*, Chengdu, China, Jun. 2006, pp. 757–760.

[23] M. Taha and Y. Hasan, "VANET-DSRC protocol for reliable broadcasting of life safety messages," in *Proc. IEEE Int'l Symp. on Signal Processing and Information Technology*, Dec. 2007, pp. 104–109.

[24] Y. Tseng, R. Jan, C. Chen, C. Wang, and H. Li, "A vehicle-density-based forwarding scheme for emergency message broadcasts in VANETs," in *Proc. IEEE Int'l Conf. on Mobile Ad Hoc and Sensor Sys. (MASS)*, San Francisco, CA, Nov. 2010, pp. 703–708.

[25] R. S. Schwartz, R. Barbosa, N. Meratnia, G. Heijenk, and H.Scholten, "A simple and robust dissemination protocol for VANETs," in *Proc. IEEE European Wireless Conf.*, Lucca, Italy, Apr. 2010, pp. 214–222.

[26] Q. Yang and L. Shen, "A multi-hop broadcast scheme for propagation of emergency messages in VANET," in *Proc. IEEE Int'l Conf. on Comm. Technology (ICCT)*, Nanjing, China, Nov. 2010, pp. 1072–1075.

[27] M. Zorzi and R. Rao, "Geographic random forwarding (GeRaF) for ad hoc and sensor networks: Energy and latency performance," *IEEE Trans. Mobile Comput.*, vol. 2, no. 4, pp. 349–365, Oct.-Dec. 2003.

[28] G. Ciccarese, M. De Blasi, P. Marra, C. Palazzo, and L. Patrono, "On the use of control packets for intelligent flooding in VANETs," in *Proc. IEEE Wireless Comm. and Networking Conf. (WCNC)*, Budapest, Hungary, Apr. 2009, pp. 1–6.

[29] H. Alshaer and E. Horlait, "An optimized adaptive broadcast scheme for inter-vehicle communication," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, Stockholm, Sweden, May 2005, pp. 2840–2844.

[30] A. Wegener, H. Hellbrück, S. Fischer, C. Schmidt, and S. Fekete, "AutoCast: An adaptive data dissemination protocol for traffic information systems," in *Proc. IEEE Vehicular Technology Conf. (VTC)*, Baltimore, MD, Sep. 2007, pp. 1947–1951.

[31] S. Panichpapiboon and G. Ferrari, "Irresponsible forwarding," in *Proc. IEEE Int'l Conf. on ITS Telecomm. (ITST)*, Phuket, Thailand, Oct. 2008, pp. 311–316.

[32] M. Slavik and I. Mahgoub, "Stochastic broadcast for VANET," in *Proc. IEEE Consumers Comm. and Networking Conf. (CCNC)*, Las Vegas, NV, Jan. 2010, pp. 1–5.

[33] H. Hellbrück and S. Fischer, "MINE and MILE: Improving connectivity in mobile ad-hoc networks," *ACM Mobile Computing and Comm. Review*, vol. 8, no. 4, pp. 19–36, Oct. 2004.

[34] S. Busanelli, G. Ferrari, and S. Panichpapiboon, "Efficient broadcasting in IEEE 802.11 networks through irresponsible forwarding," in *Proc. IEEE Global Telecomm. Conf. (GLOBECOM)*, Honolulu, HI, Nov. 2009, pp. 1–6.

[35] R. Ahlswede, N. Cai, S.-Y. R. Li, and R. W. Yeung, "Network information flow," *IEEE Trans. Inf. Theory*, vol. 46, no. 4, pp. 1204–1216, Jul. 2000.

[36] R. W. Yeung, "Network coding: A historical perspective," *Proc. IEEE*, vol. 99, no. 3, pp. 366–371, Mar. 2011.

[37] S. Katti, H. Rahul, W. Hu, D. Katabi, M. Medrad, and J. Crowcroft, "XORs in the air: Practical wireless network coding," *IEEE/ACM Trans. Netw.*, vol. 16, no. 3, pp. 497–510, Jun. 2008.

[38] L. Haojie, Z. Dongdong, and Y. Chen, "Local-directed network coding in vehicular ad-hoc networks," in *Proc. IEEE Int'l Symp. on Intelligent Signal Processing and Comm. Sys. (ISPACS)*, Chengdu, China, Dec. 2010, pp. 1–4.

[39] L. Li, R. Ramjee, M. Buddhikot, and S. Miller, "Network coding-based broadcast in mobile ad hoc networks," in *Proc. IEEE Conf. on Computer Comm. (INFOCOM)*, Anchorage, AK, May 2007, pp. 1739–1747.

[40] W. Lou and J. Wu, "On reducing broadcast redundancy in ad hoc wireless networks," *IEEE Trans. Mobile Comput.*, vol. 1, no. 2, pp. 111–122, Apr.-Jun. 2002.

[41] S. Yang and J. Wu, "Efficient broadcasting using network coding and directional antennas in MANETs," *IEEE Trans. Parallel Distributed Sys.*, vol. 21, no. 2, pp. 148–161, Feb. 2010.

[42] N. Kadi and K. Agha, "MPR-based flooding with distributed fountain network coding," in *Proc. IFIP Annual Mediterranean Ad Hoc Networking Workshop (Med-Hoc-Net)*, Juan-les-pins, France, Jun. 2010, pp. 1–5.

[43] A. Qayyum, L. Viennot, and A. Laouiti, "Multipoint relaying for flooding broadcast messages in mobile wireless networks," in *Proc. IEEE Hawaii Int'l Conf. on Sys. Sciences (HICSS)*, Big Island, HI, Jan. 2002, pp. 3866–3875.

[44] T. Zhong, B. Xu, and O. Wolfson, "Disseminating real-time traffic information in vehicular ad-hoc networks," in *Proc. IEEE Intelligent Vehicles Symp. (IV)*, Eindhoven, The Netherlands, Jun. 2008, pp. 1056–1061.

[45] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "TrafficView: A scalable traffic monitoring system," in *Proc. IEEE Int'l Conf. on Mobile Data Management (MDM)*, 2004, pp. 1–14.

[46] T. Fujiki, M. Kirimura, T. Umedu, and T. Higashino, "Efficient acquisition of local traffic information using inter-vehicle communication with queries," in *Proc. IEEE Intelligent Transportation Sys. Conf. (ITSC)*, Seattle, WA, Sep. 2007, pp. 241–246.

[47] Q. Yu and G. Heijenk, "Abiding geocast for warning message dissemination in vehicular ad hoc networks," in *Proc. IEEE Int'l Conf. on Comm. (ICC)*, Beijing, China, May 2008, pp. 400–404.

[48] L. Wischhof, A. Ebner, and H. Rohling, "Information dissemination in self-organizing intervehicle networks," *IEEE Trans. Intell. Transp. Syst.*, vol. 6, no. 1, pp. 90–101, Mar. 2005.

[49] The MathWorks, Inc., "MATLAB website," [Online], 2011, available: http://www.mathworks.com/.

[50] H. Schwetman, "CSIM user's guide rev. 1," Microelectronics and Computer Technology Corporation, Tech. Rep. ACT-126-90, 1991.

[51] OPNET Technologies, Inc., "OPNET website," [Online], 2011, available: http://www.opnet.com/.

[52] "The Network Simulator (NS-2) website," [Online], 2011, available: http://nsnam.isi.edu/nsnam/index.php/.

[53] D. R. Choffnes and F. E. Bustamante, "An integrated mobility and traffic model for vehicular wireless networks," in *Proc. ACM Int'l Workshop on Vehicular Ad Hoc Networks. (VANET)*, Cologne, Germany, Sep. 2005, pp. 69–78.

[54] E. Teramoto, M. Baba, H. Mori, H. Kitaoka, I. Tanahashi, and Y. Nishimura, "Prediction of traffic conditions for the Nagano olympic winter games using traffic simulator: NETSTREAM," in *Proc. World Congress on ITS*, 1998, pp. 1801–1806.

[55] "OMNeT++ network simulator framework," [Online], 2011, available: http://www.omnetpp.org/.

[56] S. Panichpapiboon and W. Pattara-atikom, "Connectivity requirements for self-organizing traffic information systems," *IEEE Trans. Veh. Technol.*, vol. 57, no. 6, pp. 3333–3340, Nov. 2008.

[57] M. Brackstone and M. McDonald, "Car-following: A historical review," *Transportation Research Part F*, vol. 2, no. 4, pp. 181–196, Dec. 1999.

[58] S. Maerivoet and B. De Moor, "Cellular automata models of road traffic," *Physics Reports*, vol. 419, no. 1, pp. 1–64, Nov. 2005.

[59] R. Mangharam, D. S. Weller, R. Rajkumar, P. Mudalige, and F. Bai, "GrooveNet: A hybrid simulator for vehicle-to-vehicle networks," in *Proc. IEEE Int'l Conf. on Mobile and Ubiquitous Sys.*, San Jose, CA, Jul. 2006, pp. 1–8.

Sooksan Panichpapiboon (S'05-M'07) received the B.S., M.S., and Ph.D. degrees in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, in 2000, 2002, and 2006, respectively. In April 2008, he was a Visiting Researcher with the Department of Information Engineering, University of Parma, Italy. He is currently a faculty member in the Faculty of Information Technology, King Mongkut's Institute of Technology Ladkrabang, Bangkok, Thailand. He has served as a technical program committee member for many international conferences. His current research interests include ad hoc wireless networks, intelligent transportation systems, radio frequency identification (RFID) systems, and performance modeling. Dr. Panichpapiboon was the recipient of the ASEM DUO-Thailand Fellowship in 2007 and the Doctoral Dissertation Award from the National Research Council of Thailand in 2011.

Wasan Pattara-atikom (M'04) received the B.E. degree in computer engineering from Khon Kaen University, Khon Kaen, Thailand. He received the M.S. degree in telecommunications, the M.B.A. degree in business administration, and the Ph.D. degree in information science from the University of Pittsburgh, Pittsburgh, PA.

He is currently a Senior Researcher with the Network Technology Laboratory, National Electronics and Computer Technology Center, Pathumthani, Thailand where he leads a project on vehicular traffic prediction. His current research interests are vehicular ad hoc network, intelligent transport system, and machine learning.