

# Location- and Delay-Aware Cross-Layer Communication in V2I Multihop Vehicular Networks

Boangoat Jarupan and Eylem Ekici, *The Ohio State University*

## ABSTRACT

Intelligent transportation systems are targeted to improve the traffic safety and driving experience of passengers. Vehicular ad hoc networks are wireless communication networks proposed to be used as parts of ITS. VANETs facilitate communication among vehicles, and between vehicles and roadside equipment. A key challenge in developing such systems is to design routing and MAC protocols that not only provide good end-to-end packet delay but can also quickly adapt to changes in the network topology due to vehicular mobility. In this article we outline a new framework for location- and delay-aware cross-layer communication that addresses these challenges. Our framework provides an efficient V2I data delivery system that relays packets over low-delay paths to a fixed base station or access point. Furthermore, an instance of this framework is also presented as a protocol. Our preliminary evaluations show that our design approach is promising, and provides delay predictability, fairness, and a good packet delivery ratio.

## INTRODUCTION

Intelligent transportation systems (ITS) aim to leverage modern information and communication technologies to improve current transportation systems in dealing with increasing number of accidents, transportation time, and travel cost. A vehicular ad hoc network (VANET) is an important component of ITS, which aims to improve the performance of the current communication system. VANETs allow mobile nodes to communicate with roadside infrastructure (i.e., base stations [BSs] or Internet access points [V2I]) as well as among themselves (V2V) [1].

In this article we discuss a framework that is targeted at development of efficient V2I *data delivery systems*. Unlike the majority of existing cellular communication systems, our framework supports multihop communication where nodes between a sender vehicle and the BS act as intermediate *relay nodes*. This type of multihop communication allows BSs or access points to provide service beyond their immediate single-hop service range. It reduces the implementation costs since it requires fewer BSs to serve a larger area.

Our framework can be used in many critical applications. For example, in safety-related applications, vehicles can leverage our framework to send emergency messages pertaining to serious incidents such as accidents to nearby BSs. These stations can then relay these messages to police, hospitals, fire departments, and so on. Our framework is also useful in electronic payment (e-payment) systems such as electronic toll collection (ETC). ETC systems take advantage of V2I communication to perform e-payment transactions while vehicles are passing through a toll station (or BS), without requiring them to reduce speed. Efficient V2I communication can also be used for e-commerce applications with which local merchants can broadcast their service advertisements along the road. Travelers can conveniently make electronic transactions such as shopping, reserving hotels, and ordering food while traveling. Furthermore, the BS provides Internet access to nearby vehicles, which facilitates infotainment applications like downloading music and online shopping.

Different applications mentioned above often require critical performance guarantees. Messages in safety-related applications must be in a real-time manner, and require fast and reliable wireless communication. Similar guarantees are also required in e-commerce and ETC systems while making payments. Designing a framework to support these delay-sensitive applications is one of the main research challenges in VANETs and the focus of our article.

Delay-aware communication in VANETs is a challenging task due to, among other factors, the following main reasons:

- A highly dynamic network topology
- Varying vehicle density and network connectivity
- The presence of data bottlenecks

Due to high vehicle speeds, the topology of VANETs changes at a much faster rate than traditional mobile ad hoc networks (MANETs). Therefore, existing protocols designed for MANETs cannot be directly implemented on VANETs. The majority of existing techniques developed for VANETs are location-based rather than address-based. These solutions, however, do not take channel congestion and data

traffic characteristics into account, and hence cannot always transmit packets over minimum-delay paths. A packet may experience small delays on some road segments but longer delays on others. Especially in V2I communication, high contention around BSs is likely to increase the problem of *hotspots*. It is thus essential to develop a new communication framework that can adapt to dynamic network topologies and avoid routing over congestion paths.

We propose a framework that addresses the above mentioned challenges and provides delay-aware data delivery. Our framework relies on a design where vehicles *periodically monitor and update* packet traffic information, which is then used to route packets *over low-delay paths in a distributed manner*. It is a lightweight framework with minimal overhead, and the traffic information used by vehicles is sufficient for them to identify favorable paths. The routing is performed in a distributed manner where source vehicles make high-level routing decisions that guide the intermediate relay nodes in making local decisions about next-hop nodes. Based on this framework, we propose a data delivery system called the Location-Delay-Aware Cross-Layer Communication Protocol (LD-CROP) suite that delivers packets over low-delay paths to fixed BSs.

## RELATED WORK

MANETs and VANETs share a number of similarities, which include self-organization, self-management, and low bandwidth. However, unlike MANETs, the network topology and the movement of nodes in VANETs are highly dynamic, and node movement is typically constrained by road structure. MANET ad hoc routing protocols like Ad Hoc On-Demand Distance Vector (AODV) and Dynamic Source Routing (DSR) fail to deal with dynamic topologies because of their addresses-based routing. On the other hand, position-based routing protocols such as Greedy Perimeter Stateless Routing (GPSR) do not consider the underlying vehicular road network, but try to route packets over shortest paths. Whenever these paths become invalid due to vehicular mobility, packets are likely to get routed over longer paths. Thus, instead of naively adopting the MANET protocols, we need to develop efficient methods that take specific characteristics of VANETs into account.

Existing routing protocols for VANETs can be broadly categorized into two classes: V2V and V2I. Several V2V protocols adopt position-based routing techniques since they are more lightweight and adaptable than ad hoc routing techniques [2]. For instance, Geo-Spatial Referencing (GSR) [3] makes use of global static street maps to find the path to the destination. These protocols make routing decisions dynamically at each intersection based on information about streets and neighbors. Since the success of these protocols rely on the quality of neighborhood information, vehicles must frequently update their neighborhood tables to keep up with topological changes in the network. This results in a large number of control packets and leads to packet congestion, especially around intersections. More important, since these proto-

cols lack global knowledge of the paths, the packets can potentially get routed over non-optimal paths. Although V2V is not the focus of our framework, our design also takes advantage of digital roadmaps in mapping *communication paths* with positional information into *source routes* along physical roads. Such mapping is critical in the context of vehicular mobility.

V2I protocols, on the other hand, are designed to support data delivery systems where multiple vehicles send information to common BSs. Existing V2I protocols target the problem of intermittent network connectivity in several ways. Some protocols use a store-and-forward technique where relay nodes carry packets until suitable receivers are found. While this technique increases reliability, it often leads to longer packet delays — thereby making it unsuitable for delay-sensitive applications.

VADD [4] adopts a store-and-forward technique and routes packets via high-vehicle-density road segments. This approach again may route packets over longer non-optimal paths. Furthermore, since all vehicles try to transmit packets via roads with high vehicle density, those roads may become bottlenecks with increased interference and packet contention. These problems are partially alleviated by estimating the packet delays a priori while choosing the path. Existing estimation techniques, however, are not robust as they rely on speed limit on road segments and average vehicle densities, which are likely to vary over time.

There are some other techniques that predict the optimal paths using vehicle information such as trajectory, location, direction, and speed to predict link breakage and, as a result, improve end-to-end throughput [5–7]. While these techniques are developed to deal with connectivity issues, they cannot be used for delay-sensitive applications due to the presence of channel congestion.

Unlike existing approaches, our framework makes routing decisions based on *packet traffic characteristics* captured via different real-time statistics collected and propagated by vehicles at regular intervals. Such information not only assists in acquiring knowledge about data traffic but also helps in adapting to dynamic topology changes. Since we do not rely on any neighborhood information, our framework incurs much less overhead than the state of the art.

Apart from routing protocols, a wide range of medium access control (MAC) protocols have also been proposed for VANETs. The challenge here is not only to deal with fast topological changes but also to reduce the channel access delay. Since VANETs also rely on the wireless medium, some MAC protocols are based on broadcasting. These protocols deal with the popular flooding problem by taking advantage of the road structure implementing directional broadcasting. Protocols like UMB [8] and MHVB [9] simply select the farthest node in the direction of broadcast to forward the packets. Other nodes who overhear the forwarded message omit their own retransmission, thereby reducing the overhead incurred in flooding. In our framework we leverage a modified version of UMB to achieve low-overhead communication while delivering the packets.

Unlike existing approaches, our framework makes routing decisions based on packet traffic characteristics captured via different real-time statistics collected and propagated by vehicles at regular intervals.

Our framework is based on a cross-layer design between MAC and routing functions. The MAC collects various local packet traffic statistics by observing the channel. These statistics are then used by the routing function in determining the best available path through a delay estimation model.

Recently, a new standard, IEEE 802.11p, has been proposed for VANETs. It has the basic functionality of IEEE 802.11 and the channel access priority scheme similar to IEEE 802.11e. However, the standard does not address multi-hop networking related problems such as hidden terminals. Some strategies attempt to reduce the extra channel access delay due to contention by employing dynamic channel access scheduling mechanisms such as time-division multiple access (TDMA), which allows nodes to reserve required resources [10]. The challenge here is to design these strategies in such a way that control packet overhead is minimal.

In our previous research we developed a cross-layer protocol, CVIA, for highways [11] that provides location-independent throughput and fairness to vehicles in the network. CVIA solves the hidden terminal problem and avoids packet collision by dividing a road into segments. Each segment is a single-hop cluster where the vehicles inside the cluster transmit packets to the temporary router located at the segment border. Similarly, CCBF [12] and Traffic-Gather [13] are cross-layer protocols that rely on cluster-based forwarding. In CCBF the cluster head controls the channel allocation and provides delay bounded packet transmission. Traffic-Gather implements similar cluster forming based on road segments, and uses TDMA to collect the data within the cluster. Although the cluster-based design addresses the issue of packet collisions, they suffer from synchronization among clusters. Furthermore, the cluster forming process incurs severe overhead. Such strategies, which divide the road into segments, are more suitable for highways and are not readily applicable for urban road structures.

Our framework is based on a cross-layer design between the MAC and routing functions. The MAC collects various local packet traffic statistics by observing the channel. These statistics are then used by the routing function in determining the best available path through a delay estimation model. Given such a source route by the routing function, the MAC determines individual relay nodes and forwards the packets toward the destination. Such a cross-layer design with lightweight information exchange between routing and MAC functions provides us with very good packet transmission delay and a high success percentage.

## DELAY-AWARE DATA DELIVERY FRAMEWORK

Our data delivery framework aims at transmitting messages from vehicles to BSs using multihop communication. Our framework is built on three guiding principles: collecting and propagating packet traffic characteristics, source routing over smaller delay paths, and dynamically making the forwarding decisions at relay nodes based on high-level source route information. We briefly discuss the highlights of these main features below.

*Periodic update of traffic information:* Due to high vehicular mobility, each road section can experience a change in vehicle density and packet load. In our framework, vehicles are provided with an ability to collect and share information

about local data traffic characteristics they observe. Traffic information obtained from different locations gives vehicles a sufficiently detailed overview of the network condition. To account for changes in network connectivity, this information is *periodically* updated. A key challenge here is to keep the overhead involved in collecting and sharing this information minimal. Our framework uses a lightweight traffic information propagation system where vehicles exchange concise summaries of packet traffic information based on local observations. For example, the average number of packets entering and leaving the system can be useful in analyzing the queuing delay. The amount of information in such summaries and the frequency with which they are updated is chosen in such a way that vehicles always have sufficient information to adapt to changes in the underlying environment.

*Distributed path selection:* Vehicles make use of collected local traffic statistics in making high-level routing decisions by selecting smaller delay paths over the roadmap. For example, the source vehicle decides the series of streets and directions leading to the destination. This information guides the detailed forwarding decisions made at intermediate relay nodes in finding exact next-hop nodes. Such a distributed mechanism helps reduce the overhead of route maintenance and keeps the amount of control packet overhead low. Note that our framework does not require complex processes such as neighbor discovery and maintenance, unlike many other existing solutions.

*Avoiding oscillations in resource allocation:* Routes selected by source vehicles and the traffic information based on which the routes are selected can become stale over time. The former is due to vehicular mobility, where the source nodes may move into a different road that has a higher speed limit. The latter is due to changes in packet traffic characteristics in a given location. For instance, the amount of traffic can change suddenly due to unexpected events such as an accident. Such stale paths are detected and eliminated using information during periodic traffic update. A potential problem of such route updates is that a node may experience oscillations in terms of the selected route to the destination. For example, nearby nodes may receive the same update information and decide to select the same path. To avoid this problem, the selected path should be changed only when a different path can provide a substantial benefit in terms of path quality.

## LD-CROP PROTOCOL OVERVIEW

Based on our proposed framework, we introduce LD-CROP, which has two types of communication processes: *beacon advertisement* and *data delivery*. In beacon advertisement, BSs periodically broadcast beacons to vehicles where as vehicles communicate to BS using minimum delay path via Data Delivery process (Fig. 1).

In this article we assume that BSs are installed at fixed locations along the road. Both vehicles and BSs have unique IDs, and they communicate using wireless radios. We assume an ideal channel for wireless communication. Vehicles are able to find their locations using the Global Positioning System (GPS), and they are able to find street-

level information using digital roadmaps. The road structure is based on urban environment grid networks, which include multiple intersections. Therefore, it is possible to receive the same packet from multiple paths. In this article we refer to intermediate nodes as relay nodes, and we use the term node and vehicle interchangeably.

### BEACON ADVERTISEMENT

Each beacon consists of a BS id (BSid), sequence number (SEQ), the route it has traveled so far (PATH), time to live (TTL), and complete path quality (CPQ). Each beacon is uniquely identified by its BSid and SEQ, which are added to the beacon by the BS. When a vehicle receives the beacon, it records PATH into its *pathTable*. It also appends its current location to PATH and adds local traffic statistics it has collected to CPQ. The vehicle then decrements TTL and rebroadcasts the updated beacon to other vehicles outward from the BS. If a node receives the same beacon multiple times but with a different value of PATH, it means that the node has multiple different paths to the BS.

Each relay node rebroadcasts the beacon according to the UMB [8] protocol we proposed as part of our earlier research. UMB is an IEEE 802.11-based directional multihop broadcast protocol. In order to reduce the flooding overhead, UMB forwards the packet to the farthest node from the source of information within the relay node's transmission range. The received PATH is converted into a sequence of (street, direction) pairs. These pairs can easily be deduced from the location information present in the PATH using a digital roadmap. Along with the PATH, we also store other information such as BSid, SEQ, and CPQ in each entry of *pathTable*. Furthermore, the entries are tagged with a last update timestamp (LUT) so that only *fresh* routes are present at any time. Stale routes whose LUT is greater than a threshold ( $\theta$ ) are removed from the *pathTable* at the time of route selection. Such a mechanism is very important, especially because of vehicular mobility.

Before forwarding the beacon, the vehicle appends its current location to the PATH and updates CPQ with the local packet traffic information it has collected so far. CPQ includes the following three statistics:

- *Service time*: The sum of channel contention time and actual transmission time.
- *Interarrival time*: The time difference between two consecutive arrivals into the packet queue.
- *Packet train size*: The average number of DATA packets sent in one transmission period. The terms *batch* and *packet train* are used interchangeably throughout this article.

More specifically, we maintain five different statistics: expected packet interarrival time, variance in packet interarrival time, expected batch service time, variance in batch service time, and expected batch size. Note that these five statistics capture the data traffic status around the vehicle's current location. We compute these statistics by taking a moving average of a fixed number (a window) of recently collected values. There are two important factors one must consider here: the

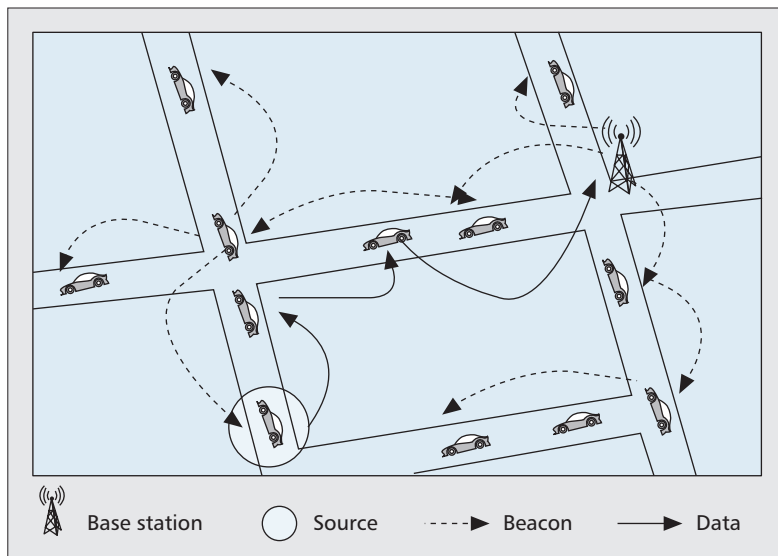


Figure 1. System architecture.

*quality* and *freshness* of data traffic statistics. While the quality is related to the vehicle that is collecting the data, the freshness affects other vehicles that make use of that data. The quality is governed by the window size, which is a function of the vehicle's speed and the distance it travels. On the other hand, the freshness is correlated to the LUT value in *pathTable*, which in turn is related to the beacon interval. For better analysis, we collect the arrival time statistics over individual packets and the service time statistics over the entire transmission period, which includes packet trains. The use of these statistics is elaborated later.

### ROUTING FUNCTIONS

The goal here is to find a best route that has minimum delay. We leverage the traffic information (CPQ) in each entry of *pathTable* to estimate the path delay. It is important to note that the traffic information is on a per relay node basis, whereas the PATH is combined over relay nodes, as explained earlier. The delay associated with a relay node is a combination of queuing delay and service delay, which are captured in the parameters that vehicles collect. For simplicity, we assume that the delays at different relay nodes are only a function of the gathered statistics at a nearby sampling point.

We model each relay node as a G/G/1 queuing system. We allow the sender node to bundle and send multiple packets with the same path as a single batch. We therefore leverage the theory developed by Curry and Deuermeyer for G/G/1 batch departure systems [14].

For a path  $P = (r_1, r_2, \dots, r_k)$  with  $k$  relay nodes, the load information is obtained from CPQ in the *pathTable*. We estimate the total delay over the entire path  $P$  as the sum over estimated delays at each relay node (i.e.,  $D[P] = \sum_{1 \leq i \leq k} D[r_i]$ ). Once the estimated path delay  $D[P]$  is calculated for the selected paths from the routing table, the path with minimum estimated delay is chosen as the *source route*,<sup>1</sup> and is attached to the data packet. The packet is then sent to the queue in the MAC function, which we describe next.



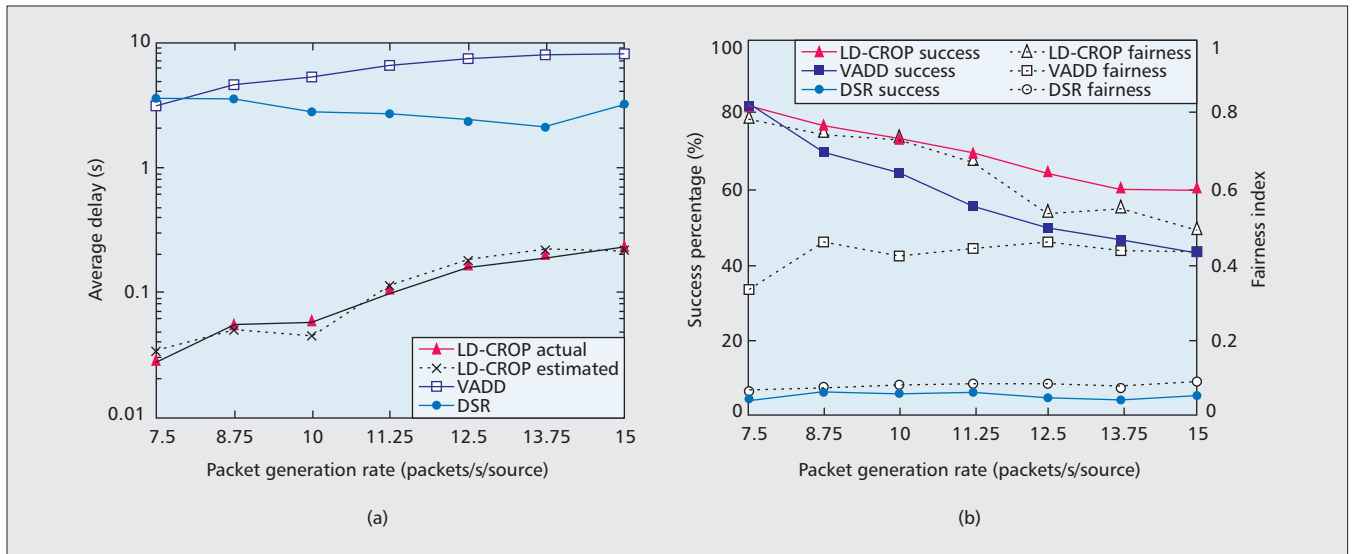


Figure 2. Performance comparison as packet generation rate is varied, transmission range = 300 m: a) average delay; b) success percentage and fairness index.

### MAC FUNCTIONS

In our MAC protocol, every packet passes through a *contention period* before it is transmitted on the channel. The channel contention mechanism is based on IEEE 802.11. Vehicle A contends for the channel by sending a forward relay request (FRREQ) packet, which includes the sender id, the target BSid, and the complete path selected from *pathTable*. The receivers of FRREQ evaluate whether or not they are *qualified* to send a reply message, forward relay reply (FRREP). Qualified receivers wait for a backoff (BO) period and send an FRREP packet.

Since the path associated with data packets contains only street and direction information, the protocol identifies the best relay node at each hop based on certain conditions — *receiver selection*. Assume that vehicle A sends FRREQ ( $f_q$ ) with the associated path  $P_{f_q}$ . If a receiver already has packets with the same path ( $P_{f_q}$ ), it is considered a *hyper relay node* (HRN). HRNs can send packets that have the same path in a bundle, which is called a packet train: a sequence of FRREQ, FRREP, DATA, ACK, ..., DATA, ACK. This mechanism helps reduce collision overhead and transmission delay. The size of a packet is set to a predefined constant value, and there is a waiting time for each packet train transmission to ensure fairness. HRNs select the smallest BO value since this can reduce average packet delay.

All non-HRN vehicles select BO values based on two parameters: *direction* and *distance*. Vehicles traveling in the same direction of the packet (given by  $P_{f_q}$ ) are given higher preference to become relay nodes. Vehicles that are far from the source (sender of FRREQ) have a better chance to become relay nodes than vehicles that are closer.

<sup>1</sup> To avoid oscillations between paths, a source route is changed only when the new minimum delay path has at least 10 percent smaller delay than the currently used one, provided that the current path has not expired.

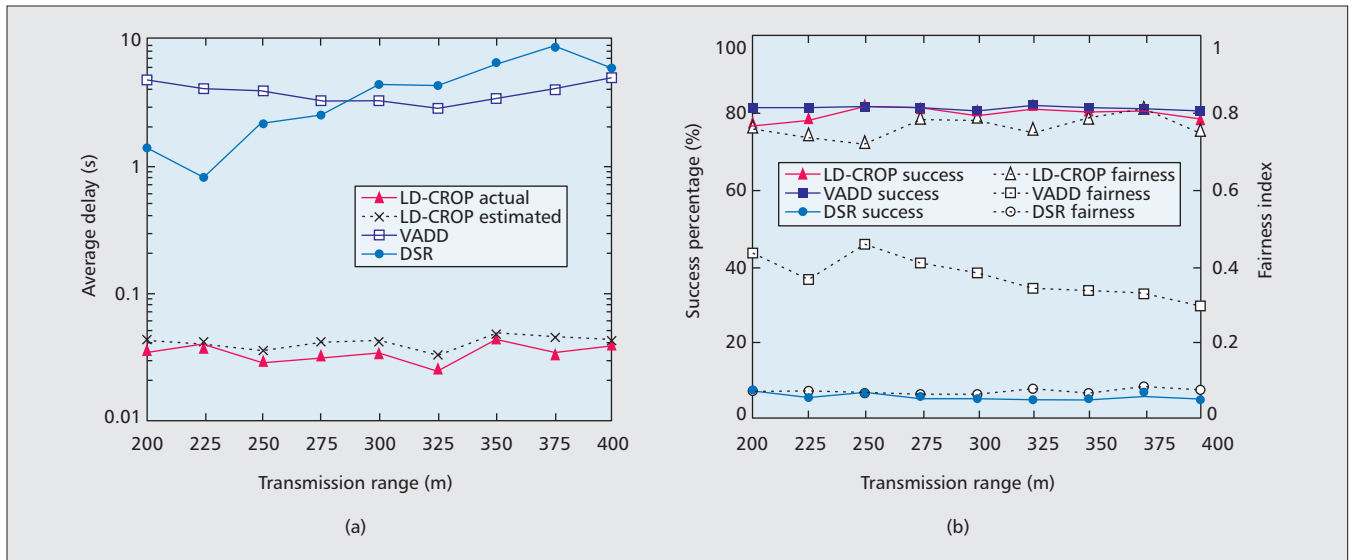
In case of an FRREP collision, the sender retransmits the FRREQ by following a binary exponential BO procedure. Only the receivers who sent FRREP in the previous round contend again by selecting a random BO value from  $[0, K]$  where  $K$  is a predefined constant, instead of repeating all distance computations. By doing so, we try to reduce the probability of collisions that may otherwise occur due to deterministic distance calculations. Other receivers that did not contend remain idle until the data packet is successfully transmitted.

### EVALUATION

#### SIMULATION SETUP

We implemented LD-CROP in NS-2. To evaluate our protocol, we consider a grid road network with one BS. The simulation area is  $2400 \times 2400$  m and consists of three roads in the north/south direction and three roads in the east/west direction. Each road is separated by 600 m and has two lanes running in opposite directions. The BS is placed at the (1800, 1200) location. We randomly place 440 vehicles over the entire road network so that each street has a vehicle density of about 15 vehicles/km/lane. The vehicle speeds are selected from a Gaussian distribution with mean 36 km/h and standard deviation of 5 km/h. For simplicity, we restrict the vehicles' movement to linear paths, and we do not allow lane changing.

The BS is set to send a new beacon every 20 s. The data packets are generated by 20 randomly selected sources. Each source generates data packets based on a Poisson distribution (default mean is 7.5 packets/s). The data packet size is 500 bytes, and maximum packet train size is 5 packets. The data rate is 3 Mb/s, and radio frequency is 5.9 GHz (DSRC). The default transmission range of vehicles is 300 m. While calculating local traffic statistics, we compute the moving average over a window of 25 entries. The entire simulation is run for 100 s, and the results are averaged over 15 independent runs.



**Figure 3.** Performance comparison as transmission range is varied, packet generation rate = 7.5packets/s/source: a) average delay; b) percentage success and fairness index.

### SIMULATION RESULTS

We consider four different metrics in our evaluation: estimated packet delay ( $D[P]$ ); actual packet delay, which is the actual end-to-end delay; success percentage, which represents the fraction of packets that are successfully transmitted to the BS; and fairness index [15], which is a measure that captures how bandwidth is partitioned across different packet flows. Fairness index is computed as

$$\frac{(\sum x_i)^2}{(n \cdot \sum x_i^2)},$$

where  $n$  is the total number of flows and  $x_i$  is the throughput of flow  $i$ . We compare our results with VADD [4]-802.11 and DSR-802.11 protocol pairs. DSR results are used for baseline comparison. Overall, our approach outperforms the DSR-802.11 protocol pair in terms of packet delay, success rate, and fairness. DSR is designed for MANETs where node movement is much slower than that of vehicles.

We first change the rate at which 20 source nodes generate packets by adjusting the mean of Poisson distribution. Figure 2a demonstrates the effectiveness of our delay estimation model. Since our model takes the real-time packet traffic statistics into account, the difference between estimated and actual delay is very small. On average, estimated delay is within 3 percent of actual delay. As the packet generation rate increases, average packet delay of both VADD and LD-CROP increases due to increased contention. However, LD-CROP consistently performs better than VADD at all packet generation rates — note that the y-axis is shown in log scale. This difference is mainly due to the following reasons. First, our model makes use of the packet train mechanism to bundle packets traveling in the same direction, resulting in less contention. Our empirical result shows that the average packet train size is 3.5 when the packet rate is more than 10 packets/s/source. Second, our model always routes the packets along smallest delay paths estimated using real-time traffic statis-

tics. In contrast, VADD uses preloaded information for path delay estimation.

As shown in Fig. 2b, the success rate of LD-CROP is higher than VADD at all packet loads. The number of dropped packets for VADD continues to increase as the network load increases. This is because VADD routing determines the next relay node based on neighborhood lists, and MAC simply tries to send the packet to the selected relay node. LD-CROP, however, can adapt to changes in packet traffic conditions due to our cross-layer design; the routing function determines the best path and direction, and the MAC function finds the best relay node in that direction. Similarly, the fairness guaranteed by LD-CROP is always greater than or equal to the fairness provided by VADD (Fig. 2b). VADD shows poor fairness since packets with long paths are likely to choose routes with high congestion.

We observed similar trends in average delay as we vary the transmission range (packet rate is fixed at 7.5 packets/s/source) (Fig. 3a). At high transmission ranges, every vehicle can reach out to a farther distance. VADD and DSR show significant increases in end-to-end packet delay due to high contention overhead and interference, while the delay in our approach does not change as much. Figure 3b shows that the change in percentage of success of all protocols is minimal as the transmission range is varied. By distinguishing multiple paths to the BS and collecting packet traffic data on these paths, our protocol is able to deliver packets at much smaller delay and with a much better success rate. Figure 3b illustrates that all vehicles receive a fair share of the system resources under LD-CROP.

## OPEN RESEARCH CHALLENGES AND CONCLUSIONS

In this article we introduce a framework for delay-aware data delivery in VANETs. The basic principles of this framework include capturing actual data traffic conditions in the network, delegation of route decisions to sources, and leav-

Unlike other communication solutions that may or may not be realized, standards have de facto binding properties. Therefore, it is imperative that standards incorporate sufficient flexibility to allow different communication protocol implementations to foster progress in VANETs.

ing detailed forwarding functions to intermediate nodes. With these guiding principles, we also introduce a new cross-layer protocol, LD-CROP, that routes packets over low-delay paths. The route selection mechanism of LD-CROP relies on a delay estimation model that uses observed data traffic statistics propagated in service advertisement beacons of BSs. We also propose a mechanism for selecting the best next relay node as part of the MAC protocol. Through a simulation study, we showed that our delay estimation model can accurately predict the delays, and our system delivers good performance in terms of end-to-end delay, success rate, and fairness. Thus, our approach can be used in delay-sensitive applications such as emergency warning systems and real-time systems.

While LD-CROP contains a number of very desirable characteristics for data delivery in VANETs, several very important problems still require further research. These challenges include, but are not limited to, the following:

**Provable service guarantees:** Applications envisioned for VANETs are associated with diverse requirements in delay, throughput, and reliability. Yet existing solutions, including standardization efforts, rely on ad hoc methods of data delivery that do not consider quantitative metrics. To ensure healthy growth and widespread acceptance of VANET applications, communication frameworks that support provable service guarantees are required. Existing theoretical approaches to wireless networks, such as (near)-optimal scheduling schemes, constitute good starting bases for further research.

**Truly cross-layer solutions:** In environments like VANETs, communication solutions must be designed considering various dimensions, including application requirements, protocol considerations, vehicle mobility, spectral availability, and other physical constraints. Existing cross-layer solutions generally consider a limited subset of these aspects in their design. However, these dimensions all interact with each other, and truly cross-layer solutions should consider these interactions among an extensive set of dimensions in their design.

**Accurate vehicle and data traffic models:** Models of vehicular movement in urban and highway settings have been investigated in the literature based on real traffic traces and analytical models of a recursive nature. Real traffic traces cannot be used for design of communication solutions as they do not provide analytically tractable solutions. On the other hand, existing analytical models are far from matching real traffic traces. In an orthogonal direction, data traffic models for VANETs are far from being of major significance. Further research is necessary to develop models for vehicle and data traffic to support the design and evaluation of communication solutions for VANETs.

**Protocol-level rigidity of standards:** Standardization efforts on communication solutions for VANETs are mainly driven by spectral usage and security considerations. While such considerations are definitely well justified, solutions pertaining to the MAC layer and above are generally derived from existing standards without regard to their expected impact on and suitability to application requirements. Unlike other communication solutions that may or may not be

realized, standards have de facto binding properties. Therefore, it is imperative that standards incorporate sufficient flexibility to allow different communication protocol implementations to foster progress in VANETs.

## REFERENCES

- [1] H. Hartenstein et al., "Position-Aware Ad Hoc Wireless Networks for Inter-Vehicle Communications: The Fleetnet Project," *Proc. 2nd ACM MobiHoc '01*, 2001, pp. 259–62.
- [2] F. Li and Y. Wang, "Routing in Vehicular Ad Hoc Networks: A Survey," *IEEE Vehic. Tech. Mag.*, vol. 2, no. 2, June 2007, pp. 12–22.
- [3] C. Lochert et al., "A Routing Strategy for Vehicular Ad Hoc Networks in City Environments," *Proc. IEEE Intelligent Vehic. Symp.*, 2003, pp. 156–61.
- [4] J. Zhao and G. Cao, "VADD: Vehicle-Assisted Data Delivery in Vehicular Ad Hoc Networks," *IEEE Trans. Vehic. Tech.*, vol. 57, no. 3, 2008, pp. 1910–22.
- [5] J. Jeong et al., "TBD: Trajectory-Based Data Forwarding for Light-Traffic Vehicular Networks," *Proc. 29th ICDCS*, 2009.
- [6] T. Taleb et al., "An Efficient Vehicle-Heading based Routing Protocol for VANET Networks," *IEEE WCNC*, vol. 4, Apr. 2006, pp. 2199–2204.
- [7] V. Nambodiri and L. Gao, "Prediction-based Routing for Vehicular Ad Hoc Networks," *IEEE Trans. Vehic. Tech.*, vol. 56, no. 4, 2007, p. 2332.
- [8] G. Korkmaz, E. Ekici, and F. Ozguner, "Black-Burst-based Multihop Broadcast Protocols for Vehicular Networks," *IEEE Trans. Vehic. Tech.*, vol. 56, no. 5, pt. 2, 2007, pp. 3159–67.
- [9] T. Osafune, L. Lin, and M. Lenardi, "Multi-Hop Vehicular Broadcast (MHVB)," *7th Int'l. Conf. ITS*, June 2006, pp. 757–60.
- [10] H. Menouar et al., "A Survey and Qualitative Analysis of MAC Protocols for Vehicular Ad Hoc Networks," *IEEE Wireless Commun.*, 2006, pp. 30–35.
- [11] G. Korkmaz, E. Ekici, and F. Ozguner, "A Cross-Layer Multihop Data Delivery Protocol with Fairness Guarantees for Vehicular Networks," *IEEE Trans. Vehic. Tech.*, vol. 55, no. 3, 2006, pp. 865–75.
- [12] B. Wiegel, Y. Gunter, and H.P. Grossmann, "Cross-Layer Design for Packet Routing in Vehicular Ad Hoc Networks," *IEEE 66th VTC-Fall*, 2007, pp. 2169–73.
- [13] W.R. Chang, H.T. Lin, and B.X. Chen, "TrafficGather: an efficient and scalable data Collection Protocol for Vehicular Ad Hoc Networks," *5th IEEE CCNC '08*, 2008, pp. 365–69.
- [14] G.L. Curry and B.L. Deuermeyer, "Renewal Approximations for the Departure Processes of Batch Systems," *IEE Trans.*, vol. 34, no. 2, 2002, pp. 95–104.
- [15] R. Jain, D. Chiu, and W. Hawe, "A Quantitative Measure of Fairness and Discrimination for Resource Allocation in Shared Computer Systems," *DEC Research Rep. TR-301*, 1984.

## BIOGRAPHIES

BOANGOAT JARUPAN (bea@ece.osu.edu) received her B.S. and M.S. degrees in electrical and computer engineering from The Ohio State University in 1997 and 2000, respectively. She is pursuing her Ph.D. degree in electrical and computer engineering at the same university. Her research interests include mobile ad hoc communication systems with an emphasis on vehicular networks. She is currently working on cross-layer protocol design and multihop communication models for delay-sensitive applications and intersection collision warning systems.

EYLEM EKICI received his B.S. and M.S. degrees in computer engineering from Bogazici University, Istanbul, Turkey, in 1997 and 1998, respectively. He received his Ph.D. degree in electrical and computer engineering from Georgia Institute of Technology, Atlanta, in 2002. Currently, he is an associate professor in the Department of Electrical and Computer Engineering of The Ohio State University. His current research interests include wireless sensor networks, vehicular communication systems, and cognitive radio networks, with a focus on routing and medium access control protocols, resource management, and analysis of network architectures and protocols. He is an associate editor of *Computer Networks Journal* (Elsevier) and *ACM Mobile Computing and Communications Review*.