Infrastructure-Assisted Routing in Vehicular Networks

Yuchen Wu¹, Yanmin Zhu¹,²*, and Bo Li¹,³

¹Department of Computer Science and Engineering, Shanghai Jiao Tong University; ²Shanghai Key Lab of Scalable Computing and Systems; ³HK University of Science and Technology

Abstract—Deploying roadside access points (APs) or an infrastructure can improve data delivery. Our empirical results from real trace driven simulations show that deploying APs produces up to 5x performance gain in delivery ratio and reduces delivery delay by as much as 35% with simple routing. However, we also find that buffer resources at the APs become a critical factor and poor buffer allocation leads to marginal performance gain for inter-vehicle routing. Motivated by this important observation, we investigate the optimal infrastructure-assisted routing for inter-vehicle data delivery. It remains a challenging issue for two major reasons. First, the addition of APs dramatically changes delivery opportunities between vehicles, which has not been well understood by existing work. Second, packet forwarding and buffer allocation are inter-dependent and should be addressed together. To tackle the challenges, we first characterize packet delivery probability as a function of predicted vehicle trajectories and AP locations. Then, we formulate the coexisting problem of packet forwarding and buffer allocation as an optimization problem and show that it is a knapsack problem. We design a global algorithm to solve this optimization problem. For more realistic settings, we propose a distributed algorithm for packet forwarding and buffer allocation in which each vehicle and the APs make decisions locally. Through trace-driven simulations, we demonstrate that the distributed algorithm steadily outperforms other alternative approaches under a wide range of network configurations.

Keywords: Infrastructure Assisted, Vehicular Network, Buffer Allocation, Packet forwarding

I. INTRODUCTION

Wireless vehicular networks have recently received increasing attention and a wide spectrum of existing applications have been envisioned, such as driving safety [5] and content sharing [28]. Inter-vehicle routing is a fundamental service required by a vehicular network, which is particularly challenging. A few unique challenges have been identified. Two vehicles can communicate with each other when their distance is smaller than the communication range. Recent study shows that the duration time for a moving vehicle encountering a fixed point can be as short as 10 seconds on average [22]. Vehicles are moving fast, this makes the network topology dynamic and changing quickly over time. It is often difficult to find a connected path between any pair of vehicles in a vehicular network.

An infrastructure of roadside units or access points (APs) have become ubiquitously available in urban areas [9, 27]. Roadside APs can communicate with passing vehicles, store and forward data packets. Roadside APs are usually connected via a wired network (e.g., Internet). It has been widely accepted that such an infrastructure can improve data delivery in vehicular networks.

To understand potential benefits of an infrastructure for inter-vehicle data delivery, we have conducted comprehensive experiments with real vehicular traces collected from around 4000 taxis in Shanghai, China over duration of nearly two years. Our empirical study shows that the deployment of APs can bring up to 5x performance gain in delivery ratio and reduce delivery delay by up to 35% with simple routing protocols in some configurations. In the meanwhile, however, we also find that buffer resources at the APs become a critical factor. The demand for buffers quickly increases as there are more packets and vehicles. In a real-world setting, buffers at the APs cannot be arbitrarily large, which introduce high deployment and maintenance cost. Given buffers of fixed size, data packets can quickly use up the buffers. This necessitates buffer allocation management at the APs. In comparison, the required buffer size for a vehicle is relatively small. Compared with the infrastructure, the opportunity of a vehicle encountering other vehicles is far smaller. In addition, our experimental results show that poor buffer allocation leads to marginal performance gain for inter-vehicle routing.

Motivated by this important observation, we investigate the optimal infrastructure-assisted routing for inter-vehicle data delivery. It remains a challenging issue for two major reasons. First, the addition of APs dramatically changes delivery opportunities between vehicles. As an example illustrated in Figure 1, vehicle A wants to send a packet to B, which are separate by a long distance. There is no communication opportunity for vehicle A and B since there is no contact point along their trajec-

Figure 1: Illustration of a vehicular network with roadside APs in the downtown region of Shanghai, China. A vehicle can deliver a data packet to another vehicle by forwarding the packet to an AP and a different AP can then forward the packet to a relay vehicle.
tories. However, with the APs, vehicle $A$ can first forward the packet to AP$_1$. Vehicle $B$ encounters AP$_2$ at a later time. Thus, the packet is delivered to $B$ by the relay through AP$_1$ and AP$_2$. It has not been well understood by existing work how the addition of the APs impacts inter-vehicle data delivery.

Second, packet forwarding and buffer allocation are inter-dependent and should be addressed together. A forwarding decision should consider potential influence by the APs and the buffer allocation must take into account future forwarding opportunities of the packets. We will show the co-existing problem of packet forwarding and buffer allocation is still NP hard even if the complete trajectories of the vehicles are impractically assumed to be known in advance.

A few studies [6, 15] have considered data delivery when the vehicular network is supported with an infrastructure. In [15], throwboxes are introduced for helping forward data between mobile clients in a delay tolerant network (DTN). The main issue under consideration is power saving of the throwboxes. In [8-9], the authors consider one way direction routing, either from AP to vehicle or vehicle to AP. Roadside units are assumed to be not connected and have infinite buffer sizes. In [6], a vehicle is supposed to request a packet from an AP and the AP sends a reply to the vehicle. It does not address the inter-vehicle routing problem. As a result, few existing work considers the inter-vehicles routing supported by an infrastructure with limited buffers.

To tackle the aforementioned challenges, we first characterize packet delivery probability as a function of predicted vehicle trajectories and AP locations. Then, we formulate the coexisting problem of packet forwarding and buffer allocation as an optimization problem and show that it is a knapsack problem. We design a global algorithm and a distributed algorithm for making packet forwarding and buffer allocation decisions. These algorithms require no knowledge of future vehicle trajectories. Trace-driven simulations have been conducted and comparative study is performed. Results show that our proposed algorithms steadily outperform other algorithms.

The rest of the paper is organized as follows. The next section reviews related work. In Section III, we show our empirical study results, outlining the potential performance gain of infrastructure deployment and the issue of buffer allocation. Section IV describes the problem and analyzes delivery probability of a packet. In Section V, we describe the designs of a global algorithm and a distributed algorithm. Evaluation results are presented in Section VI. The paper concludes in Section VII.

II. RELATED WORK

Data delivery is important to vehicular networks and a number of routing algorithms have been proposed. Existing routing algorithms for vehicular networks can be divided into two categories.

One category of routing algorithms [1, 5, 7, 10, 12-14, 23, 25] considers inter-vehicle packet delivery in a vehicular network in which there is no infrastructure support. Algorithms of this category try to exploit the features associated with a vehicular network, such as expected delay [10], geometric distance [14, 17-18], inter-contact time [12, 16, 26] or encounter probability [7, 13, 25]. MAC layer protocols for broadcast [5] are also proposed. Some algorithms for delay tolerant networks (DTNs) have been leveraged in vehicular networks. Epidemic [1] is a typical routing algorithm in DTNs. According to this strategy, a node always forwards packets to every node it encounters. With random walk [21] a node always forwards packets to every node it encounters and keeps no copy of these packets.

The other category of routing algorithms [3, 8-9, 11, 15] considers a vehicular network with an infrastructure. TBD [8] considers data delivery from vehicles to fixed APs. A delay model of packets routing along roads is set up in this algorithm and the path with minimum delay can be found with the help of the real-time traffic condition information. TSF [9] considers data delivery from APs to vehicles. In this model, future trajectories of vehicles are assumed to be available, so that certain roadside units can be selected as relays which in term forward packets to the destination vehicles. These roadside units are not connected and there is no buffer limit for them. Some recent study on broadcast routing in vehicular networks [11] also mentions the buffer allocation problem on roadside units. They focus on data dissemination and no buffer allocation scheme is proposed. Throwboxes [15] are employed for helping relay messages in delay tolerant networks, which can hold packets and later forward the packets to other nodes. The main design focus of throwboxes is sleep scheduling for power savings. In [20], an algorithm is proposed for deciding the locations for AP deployment.

Some other studies [2, 4, 19, 22, 27, 29] focus on the feasibility and performance of Internet access on mobile vehicles.

In summary, a few of existing studies have considered the infrastructure for data delivery in vehicular networks but most of them assume one way data communication, either from vehicle to infrastructure or from infrastructure to vehicle. Some inter-vehicle algorithms have been proposed but they do not exploit the help of infrastructure. In this work, we complement these existing algorithms by exploring infrastructure-assisted routing for inter-vehicle data delivery.

III. EMPIRICAL STUDY

This section presents our empirical study based on real trace-driven experiments, and study potential performance gain by deployment of APs. In addition, we investigate the buffer demand and buffer allocation at the APs.

A. Trace-driven Simulation Study

To understand the potential impact of the APs, we conduct trace-driven simulation study. A large-scale dataset of real GPS traces from around 4,000 taxies operational in the urban area in Shanghai, China [24]. The traces span duration of nearly two years, from January 2006 to December 2007. The trace of a taxi is a sequence of positions (in longitude and latitude) tagged with timestamp.

During simulation, the APs are placed on some road intersections using the road map. Those locations with large frequencies of vehicle transverse are selected. Note that the dep-
The previous study has revealed that the required buffer size of the APs grows quickly over time. In the real world, however, the total buffer size cannot increase arbitrarily as this may cause significant costs for deployment and maintenance. We next study the effect of buffer allocation schemes when the buffer size available at the APs is fixed. We set the buffer size restriction on the buffer size and hence there is no need for a buffer allocation scheme.

In Figure 2, we plot delivery ratio of the two routing algorithms as the number of APs is varied. We can find that the delivery ratio of either of the routing algorithm can be significantly improved when more APs are added. When there is no infrastructure (i.e., number of APs equals zero), the delivery ratio is as low as 54% for flooding and 10% for random walk. The delivery ratio is increased to 75% and 68% when the number of APs is 50. In Figure 3, we plot delivery delay of the two routing algorithms as the number of APs is varied. Similarly, we find that the delivery delays of both routing algorithms dramatically decrease as more APs are added.

In summary, there is around 5x performance gain in delivery ratio and delivery delay is reduced by 35% for random walk.

We also notice that when there are over 40 APs, the increase in delivery ratio becomes marginal as the number of APs increases. Thus, in a realistic large-scale urban area a relatively small number of APs suffices. This suggests the deployment cost of the APs can be bounded.

C. Demand for Buffers

For the study of the demand for buffers at the APs, we assume the buffer size at each AP is infinite and measure the number of data packets buffered at the APs. To increase the delivery performance of the epidemic algorithm, when a packet is buffered by one AP, it is then duplicated at the other APs.

For comparative study, we derive the minimum buffer size required for the optimal data delivery. To this end, we first run the flooding algorithm. For each packet, we identify the best path that reaches the destination earliest. Along the path, we extract the arrival time and the departure time of the packet to and from the APs. By this means, we can calculate the minimum buffer size at any time point.

In Figure 4, we plot the ratio of required buffer size to the total number of packets in the network over time. We can find that the buffer size required by the epidemic algorithm grows very quickly. In comparison, the minimum buffer size for the optimal data delivery first grows and then becomes relatively steady. Note that the two curves correspond to the same delivery performance. In Figure 5, we plot the required buffer size as the packet arrival rate is varied. We can find that as the packet arrival rate increases, both the buffer size required by epidemic and the minimum buffer size increase. However, the minimum buffer size increases much slower than the buffer size required by the epidemic algorithm. We also investigate the required buffer size of a vehicle and find it is far smaller than that of the infrastructure. The typical ratio is only 1:200. This is attributed to the fact that the number of contacts of a vehicle is far smaller than that of the infrastructure.

D. Effect of Buffer Allocation

To study the improvement of packet delivery, we compare two scenarios: with and without infrastructure. To this end, we vary the number of APs from 0 to 50. We also use different packet arrival rates ($\lambda$). In this set of simulations, there is no

![Figure 2: Delivery ratio vs. number of APs for different routing algorithms (infinite buffer).](image)

![Figure 3: Delivery delay vs. number of APs for different routing algorithms (infinite buffer).](image)

![Figure 4: Ratio of required buffer size to total number of packets in the network over time.](image)

![Figure 5: Required buffer size vs. packet arrival rate.](image)

![Figure 6: Delivery ratio vs. number of APs for different buffer allocation algorithms (limited buffer).](image)

![Figure 7: Delivery delay vs. number of APs for different buffer allocation algorithms (limited buffer).](image)
to be 20% of the total packets generated in one hour. The epidemic routing algorithm is used. We compare FIFO buffer replacement and optimal replacement (derived as introduced in the previous subsection).

In Figure 6 and Figure 7, we plot delivery ratio and delivery delay for different buffer allocation schemes, respectively. We can find that the optimal buffer replacement scheme produces much better performance than FIFO. When the number of AP increases, the total buffer size also increases. As more buffer resources become available, the performance increase of FIFO becomes marginal, compared with the optimal scheme.

In summary, the use of APs can improve delivery performance significantly. Moreover, buffer resources are an important factor and a poor buffer allocation scheme may result in low delivery performance.

IV. PROBLEM DESCRIPTION AND ANALYSIS

A. Network Model and Assumptions

We consider an infrastructure-based vehicular network, which consists of two parts: the set $N$ of all the moving vehicles and the set $U$ of the connected APs.

The location of an AP, $j$, is denoted by $u_j$. Each AP has a buffer of a fixed size $B$, i.e., it can hold at most $B$ data packets. We assume that the deployment of APs is given and the locations of the APs are fixed.

The position of a vehicle is changing over time. Let $c_i(t)$ stand for the position of a node $i$ at time $t$. When we consider slotted time, the trajectory of a node $i$ is a sequence of positions, denoted by

$$T_i = \langle c_i(0), c_i(1), ..., c_i(T), ..., \rangle.$$  

(1)

When the distance between two nodes (AP or vehicle) is smaller than the communication range $D$, the two nodes can communicate with each other. In reality, encounter durations between two vehicles can be different. For simplification, we assume that each encounter of two vehicles can support the delivery of one data packet. This also takes into account the transmission capacity of the link between the two vehicles.

The APs are connected via a wired network. We assume that that the bandwidth $w$ for the connection between any pair of APs is fixed. In addition, it takes a noticeable delay for an AP to locate a data packet in its buffers and send it to another AP. Since the encounter duration is as short as several seconds in reality, it is practical to assume that an AP can only send a data packet in its own buffer to a passing vehicle within one contact.

We assume the buffer of a vehicle is sufficiently large. This is reasonable as it has a relatively limited opportunity of encountering other vehicles, compared with the whole infrastructure of the APs. Each vehicle generates data packets over time. We consider unicast data delivery, i.e., each packet $p$ has a single source, $\delta(p)$, and a single destination $\psi(p)$. For simplification of discussion, the packets are of equal size. Let $\Phi$ denote the set of all packets. TTL stands for the Time-To-Live of a packet.

Let the set of data packets buffered at AP $j$ be denoted by $b_j$. A packet, $p$, buffered in AP $j$ can be represented by

$$p \in b_j, j \in U.$$  

(2)

The performance of data delivery in the vehicular network can be measured by delivery ratio and delivery delay.

B. Problem Formulation

The goal is to find a route from the source of each packet, $p$, to its destination. The route for $p$, denoted by $R_p$, is essentially a sequence of relays that can be a vehicular node or an AP,

$$R_p = \{r_0, r_1, ..., r_{|R_p|}, r_{|R_p|}\}, r_i \in N \cup U.$$  

(3)

Apparently, $r_0 = \delta(p)$ and $r_{|R_p|} = \psi(p)$.

Since the vehicular network suffers frequency topology disconnection, a packet may be stored at a vehicle or an AP. If the packet is held by a vehicle, then it is carried while the vehicle is moving. If it is stored at an AP, it is buffered there for certain duration before it is explicitly discarded. We define $R_p(\tau, r)$ to indicate if relay $r$ holds packet $p$ at time $\tau$, which is defined by:

$$R_p(\tau, r) \triangleq \begin{cases} 1, & \text{if } r \text{ holds } p \text{ at } \tau \\ 0, & \text{if } r \text{ doesn't hold } p \text{ at } \tau \end{cases}.$$  

(4)

Then, the buffer constraint of an AP can be represented by:

$$\sum_{p \in \Phi} R_p(\tau, r) \leq B, \forall r \in U, \tau > 0.$$  

(5)

The main objective of data delivery in vehicular networks is to deliver data packets with short delay and high delivery ratio. Delivery ratio is defined as the ratio of the successfully delivered packets to the total number of packets. For a given routing algorithm, $Y$, the delivery of packet $p$ is associated with a random variable $\rho_p$, indicating whether the packet is successfully delivered,

$$\rho_p \triangleq \begin{cases} 1, & \text{if } p \text{ is delivered} \\ 0, & \text{otherwise} \end{cases}.$$  

(6)

The random variable, $\rho_p$, is dependent on the movements $\{T_i\}$ of all the nodes, forwarding scheme, buffer allocation and AP locations. As motioned before, we do not consider the placement problem of APs and hence assume it is given. The movement trajectories of the vehicles play an important role in delivery performance. However, the vehicles move at their own wills and therefore the vehicular movements are out of control of network design. Thus, when the forwarding scheme and buffer allocation are both given, the random variable, $\rho_p$, is determined by the trajectories of the nodes. Considering the uncertainty with node trajectories, we consider the expectation of $\rho_p$ over all possible trajectories,

$$\mathbb{E}[\rho_p] \in [0, 1].$$  

(7)

Now, we are able to formally describe the data delivery problem. Given the fact that vehicles move along random trajectories, we try to design the forwarding scheme and buffer allocation scheme with the following objectives. The objective of maximizing the delivery ratio is given by:

$$\max \sum_{p \in \Phi} \rho_p / |\Phi|.$$  

(8)
For simplification, we assume that the delivery of a packet is independent of each other. Since $|P|$ is constant for a given set of packets, it is equivalent to consider the following objective,

$$\max E\left[\sum_{c \in S} \rho^c\right] \leftrightarrow \max \sum_{c \in S} E[\rho^c] \tag{9}$$

After formally stating the problem of data delivery, we next prove that this problem is NP-hard even if the complete information about node movement and data packets is assumed available.

**DEFINITION 1 (Optimal Data Delivery, ODD):** The ODD problem is to find the packet forwarding scheme and the buffer allocation scheme so that objective (9) is achieved, given the deployment of APs.

**THEOREM 1:** When the complete trajectories of the vehicles are given, the ODD problem is still NP-hard.

Because of the page limit, the proof is omitted. The basic idea of proof is to reduce the edge disjoint path (EDP) problem to the ODD problem.

**C. Predictive Trajectory Model**

The future trajectory of a vehicle is usually not available. However, it is important to data delivery in vehicular networks, which has been shown previously. We propose a predictive trajectory model to estimate the future trajectories. This model is motivated by the previous study that shows the mobility of a vehicle has a strong pattern and this pattern can be used to predict the future locations of a vehicle [25].

As proposed in [25], we characterize the mobility pattern of a vehicle, $n$, as a $k$-order Markov Chain. Its transition matrix $X_n$ is defined as follows. An element $x$ in $X_n$ is the probability of the transition from a $k$-dimensional location sequence to a given location,

$$x_{c_1,...,c_k,c_{k+1}} = \Pr(c_{k+1}|c_1,...,c_k). \tag{10}$$

The transition matrix of a vehicle can be built by analyzing its historical trace.

The future trajectory is not certain. Thus, we use the concept of trajectory bundle, denoted by $\{P^c_e(t)\}|c \in S|$ to represent the possible future trajectories of a vehicle, where $P^c_e(t)$ is the probability of node $n$ to appear at location $c$ at time $t$ and $S$ stands for the whole space of possible locations.

The trajectory bundle of a vehicle can be computed iteratively. First step is to set up the initial value. Then, given the previous $k$ locations of node $n$ and its transition matrix, the probability distribution of the next location can be calculated by

$$P^c_e(t) = \sum_{c_1,...,c_k} x_{c_1,...,c_k,c_e} \times \prod_{i=1}^{k+1} P^c_e(t-k+i). \tag{11}$$

**D. Encounter Probabilities**

After deriving the trajectory bundle for each vehicle, we are able to derive the encounter probability of any pair of vehicles and the encounter probability of a vehicle with the infrastructure (i.e., any of the APs).

**Theorem 2:** Given the trajectory bundles of nodes $i$ and $j$, the probability $\epsilon_{ij}$ of node $i$ encountering node $j$ is,

$$\epsilon_{ij} = 1 - \prod_{t=0}^{TTL}[1 - \sum_{c=0}^{TTL} P^c_e(t)'.P^c_j(t)] \tag{12}$$

This theorem is given by [25].

**Theorem 3:** Given the trajectory bundles of nodes $i$, the probability $\epsilon_i$ of node $i$ encountering the infrastructure is,

$$\epsilon_i = 1 - \prod_{j \in \mathcal{F}}[1 - P^c_i(t)]. \tag{13}$$

Proof. The probability of node $i$ encountering AP $f$ equals the probability that $f$ appears in the communication coverage of AP $j$, denoted by $\epsilon_{ij}$. Note that the probability that node $i$ appears in $u_j$ at time $t$ is $\epsilon_{ij}(t) = P^c_i(t')$. And $\epsilon_{ij}$ is the probability that this node appears in $u_j$ at any time. It can be calculated by

$$\epsilon_{ij} = 1 - \prod_{t=0}^{TTL}[1 - P^c_j(t)]. \tag{14}$$

The probability that node $i$ meets no AP is

$$\prod_{j \notin \mathcal{F}}[1 - \epsilon_{ij}] = \prod_{j \notin \mathcal{F}}[1 - P^c_i(t)]. \tag{15}$$

Then, $\epsilon_i$ can be calculated by

$$\epsilon_i = 1 - \prod_{j \notin \mathcal{F}}[1 - \epsilon_{ij}]. \tag{16}$$

Combining (15) and (16) gives the encounter probability of a vehicle encountering the infrastructure.

**E. Delivery Analysis**

The delivery probability of a packet is crucial for making packet forwarding and buffer allocation decisions. We next show how to derive this delivery probability based on the encounter probability of two vehicles and the encounter probability of a node with the infrastructure.

The route $\mathcal{R}_p$ of packet $p$ is a sequence of relays, $\mathcal{R}_p = \{r_0,r_1,...,r_i,...,r_k\}$. The delivery state transition diagram of packet $p$ is illustrated in Figure 8. From the diagram, we can find that the route may include the infrastructure 0 or several times. The transition probabilities between different states can be calculated as introduced previously.

We then study the probability $\rho^c_p$ that packet $p$ is successfully delivered by route $\mathcal{R}$. The route has $k$ links, denoted by $r_i \rightarrow r_{i+1}, 0 \leq i \leq k - 1$. Let $Pr(r_i \rightarrow r_{i+1})$ denote the probability of packet $p$ being successfully forwarded from $r_i$ to $r_{i+1}$. Then, the probability of packet $p$ is delivered through $\mathcal{R}$ is,

$$\rho^c_p = \prod_{i=0}^{k-1} Pr(r_i \rightarrow r_{i+1}). \tag{17}$$

The probability $Pr(r_i \rightarrow r_{i+1})$ depends on the nature of the link. There are in total four cases.

Case 1: The link is from a vehicle to another vehicle, i.e., both $r_i$ and $r_{i+1}$ are a vehicle node. It is easy to compute,

$$Pr(r_i \rightarrow r_{i+1}) = \epsilon_{r_i,r_{i+1}}, r_i, r_{i+1} \in N. \tag{18}$$

Case 2: The link is from a vehicle to an AP. That is, $r_i$ is a vehicle node and $r_{i+1}$ is an AP. This case happens a vehicle forwards packet $p$ to one of the APs. The probability of this link is,

$$Pr(r_i \rightarrow r_{i+1}) = \epsilon_{r_i,r_{i+1}}. r_i \in N, r_{i+1} \in U. \tag{19}$$
Case 3: The link is from an AP to a vehicle, i.e., \( r_i \) is an AP and \( r_{i+1} \) is a vehicle. This case happens when \( r_i \) holds packet \( p \) in its buffer and forwards it to \( r_{i+1} \). The probability of this link is,

\[
Pr(r_i \rightarrow r_{i+1}) = Pr(r_i) \times \varepsilon(r_i, r_{i+1}), r_i, r_{i+1} \in U, r_{i+1} \in N.
\]

where \( Pr(r_i) \) is the probability that AP \( r_i \) holds packet \( p \), which is dependent on the buffer allocation scheme at the infrastructure. In the ideal case, it equals to one if the AP always keeps this packet.

Case 4: The link is between two APs. This case happens when AP \( r_i \) moves or copies packet \( p \) to AP \( r_{i+1} \). This is decided by the buffer allocation scheme. The probability of this link is,

\[
Pr(r_i \rightarrow r_{i+1}) = Pr(r_i) \times Pr(r_i \Rightarrow r_{i+1}), r_i, r_{i+1} \in U,
\]

where \( Pr(r_i \Rightarrow r_{i+1}) \) denotes the probability that packet \( p \) is moved or copied to AP \( r_{i+1} \).

In essence, for packet \( p \), the buffer allocation at the infrastructure determines whether one buffer unit is allocated to it and the time length of this unit available to the packet. Thus, the buffer allocation to packet \( p \), denoted by \( \Omega_p \), can be modeled by a set of 2-tuples:

\[
\Omega_p = \{(x_u, t_u) | u \in U\},
\]

where \( x_u \in \{0,1\} \) indicates whether to allocate a buffer unit of AP \( u \) and \( t_u \in (0,TTL) \) indicates how long to store it. In a practical setting, \( t_u \) is usually known until this packet is replaced.

Thus, \( Pr(r_i) \) and \( Pr(r_i \Rightarrow r_{i+1}) \) are expressed as,

\[
Pr(r_i) = f(x_{r_i}, t_{r_i}),
\]

\[
Pr(r_i \Rightarrow r_{i+1}) = g(x_{r_i}, t_{r_i}, x_{r_{i+1}}, t_{r_{i+1}}), r_i, r_{i+1} \in U.
\]

For analysis simplification, we assume \( t_u \) is sufficiently long and \( x_u = 1 \) so that we get the maximum contribution made by the APs to the packet delivery.

With all the probabilities, the delivery probability through \( \rho_p^2 \) can be calculated. The expectation of delivery probability \( \mathbb{E}[^\rho_p] \) of \( p \) through all possible routes can be calculated by

\[
\mathbb{E}[^\rho_p] = 1 - \prod_{\text{all } R} (1 - \rho_p^2).
\]

V. ALGORITHMS

We first design a global algorithm for buffer allocation and packet forwarding that assumes the global knowledge about packets and vehicles. Then, we propose a distributed algorithm with which each vehicle and the APs make buffer allocation and packet forwarding locally. The global algorithm serves a baseline for performance study.

A. Global Algorithm

The overall goal of the network is to maximize the expected packet delivery ratio, which is the sum of the delivery probability of all packets. With predicted trajectories of vehicles, we derived the delivery probability of a packet which is dependent on the joint operation of buffer allocation and packet forwarding. The central idea of the global algorithm is to select the best operation of buffer allocation and packet forwarding that maximizes the increment of the overall expected delivery ratio.

To facilitate description of the global algorithm, we define some notations. Let \( C_F \) and \( C_B \) denote the set of the currently available packet forwarding and buffer allocation choices, respectively. Let \( \Theta \) denote the current expected overall delivery ratio. We define an indicator variable \( y_q \) for each operation choice \( q \in C_F \cup C_B \), \( y_q \in \{0,1\} \). If \( y_q \) takes 1, then \( q \) is performed; otherwise, it is not performed.

For a candidate network operation \( q \in C_F \cup C_B \), the increment \( \Delta \Theta_q \) is defined by,

\[
\Delta \Theta_q = \Theta \{\text{perform } q\} - \Theta.
\]

Then, the global algorithm is to select such a set of operations from the currently available operation choices with the objective of maximizing the sum of performance increments.

\[
\max \sum_{q \in C_F \cup C_B} \Delta \Theta_q \times y_q
\]

s.t. \( b_j \leq B \forall j \in U \)

\[d(l_{i\rightarrow j},l_{a\rightarrow b}) > D, \forall i,j,a,b \in N \cup U\]

where \( l_{i\rightarrow j} \) and \( l_{a\rightarrow b} \) are two active links and \( d(l_{i\rightarrow j},l_{a\rightarrow b}) \) is the minimum distance between \( i \) and \( b \) or \( a \) and \( j \). This is a knapsack problem and can be solved in pseudo-polynomial time.

Our global algorithm adopts a greedy approximation algorithm which guarantees that the result is no less than half of the optimal solution. It first calculates the delivery probability increment for each candidate choice. Then, it iteratively selects the best choice. Each time a choice is selected, those choices violating the constraints on buffer size and link interference are eliminated.

B. Distributed Algorithm

In a practical setting, any individual vehicle has no access to the global knowledge. Similarly, it is also impractical for the APs to have the global knowledge. Thus, it is necessary to design distributed algorithm for vehicles and the APs. The APs are well connected by a wired network and hence they can quickly exchange information with each other. As a result,
each AP can be considered to have the complete information about the other APs.

The distributed algorithm consists of three major components. The first component deals with the fusion of the current locations of the vehicles. With vehicle locations, we are able to predict future trajectories of the vehicles which are necessary for computing the delivery probability of a packet. The second component concerns packet forwarding which is needed by both vehicles and APs. The third component deals with the buffer allocation at the APs.

1) Knowledge Fusion

The required knowledge for packet forwarding and buffer allocation can be divided into two parts: static and dynamic. The static knowledge includes the total number of vehicles, locations of APs and mobility patterns of each vehicle. As such knowledge is stable; it can be pre-distributed to the vehicles and the APs. The dynamic knowledge includes the set of existing packets and the current locations of the vehicles. Since a node cannot directly help the delivery of a packet that it does not carry, it brings marginal benefit to exchange the set of existing packets. Thus, the key knowledge that should be shared among the vehicles is their current locations.

We propose a method for knowledge fusion among the vehicles. Each vehicle maintains two tables. One table contains the locations of the vehicles and each location is associated with a timestamp. The other table is the list of packets carried by the vehicle. Each time two vehicles meet each other, they exchange the two tables. On receiving the two tables, the vehicle first compares the two location tables and updates its own table if one entry is not present in its own table or has a more update timestamp. Next, the packet list is then used in packet forwarding and the main purpose is to avoid sending a packet when the other vehicle has already kept it. The APs perform in the same way. We leave reducing the warm-up time of the table and efficiency management about the size of the table as further work.

2) Packet Forwarding

Both the APs and the vehicles need a packet forwarding scheme. We design a packet forwarding algorithm which is invoked each time an encounter happens. The central idea of the packet forwarding algorithm is to sort all packets according to their potential increments in delivery probability. The packet with the maximum increment is selected. In addition, each packet is associated with an index that helps prevent the packet being forwarded to a node providing a lower delivery probability for the packet.

When vehicle $i$ encounters another vehicle $j$, it first exchanges the two tables with $j$ as introduced previously. Next, it updates the delivery probability of every packet $p$ on it. This new probability can be calculated according to (24) based on the updated table of vehicle locations. Then, it computes the potential increment in delivery probability for each packet $p$ on it, assuming that it was forwarded to vehicle $j$. Then the packet with the maximum potential increment is selected. Vehicle $i$ and $j$ make the forwarding decisions independently, but in a time slot only one packet transmission is possible. The computation between $i$ and $j$ can be resolved in many ways. A simple solution is to utilize the media access control protocol. The one that gains the channel access first wins. One vehicle that senses a busy channel drops the transmission attempt.

The index for each packet $p$ is initialized to the probability that the source node of the packet encounters the APs multiplied by the probability that the destination node of the packet encounters the APs. The packet will not be forwarded to $j$ if it provides a delivery probability lower than the index. Each time a packet $p$ is forwarded from $i$ to $j$, its index is updated to the delivery probability of $p$ on node $j$.

3) Buffer Allocation

The APs use the buffer allocation scheme to decide the buffer resource allocation.

We design a buffer allocation algorithm that is invoked each time an AP receives a new packet. In this algorithm, it is possible to place multiple copies in different APs. The AP determines the number of copies and the set of APs that will buffer this new packet. In the meanwhile, some existing packets are eliminated from the buffers at these APs. The algorithm introduces a parameter $\theta$ to control the maximum number of copies that a copy can be put in the AP buffers. Note that it is not necessary that each packet buffered in the infrastructure has $\theta$ copies.

We next explain the details of the buffer allocation algorithm. Suppose an AP receives packet $p$. The AP computes the delivery probability $\rho_p^a$ of packet $p$ if the packet is buffered at each AP, $u$. Let $\rho_p^u$ denote the smallest delivery probability out of the packets in AP $u$. The APs are sorted in descending order of $\rho_p^u - \rho_p^a$. Then, the top $\theta$ APs are selected. Since $\rho_p^u - \rho_p^a$ may be negative, APs with negative results are discarded. For the remaining APs, packet $p$ is added to their buffers. On each of these APs, the packet corresponding to $\rho_p^u$ is dropped.

VI. EVALUATION

A. Methodology and Simulation Setup

To study the performance of our approach, we conduct real drive driven simulations. Two performance metrics are considered, i.e., delivery ratio and delivery delay. The real vehicular traces are used for simulations, as introduced in Section III.

The following parameters are used in the simulations by default if not specified elsewhere. The packet arrival rate is 2 packets per minute. Packets are of the same size. The source and destination of a packet are randomly selected from the vehicles. The TTL is set to 2 hours. There are 30 APs. The deployment of the APs is fixed, in which the APs are placed at the intersections of main roads. In the network there are 800 vehicles. The buffer size at each AP is 24,000 packets. The link bandwidth is one packet per second. The communication range is 300 m for both vehicles and APs.

For performance study, we consider several important variables, including packet arrival rate, number of vehicles, number of APs and buffer size.
To facilitate comparative study, we consider several alternative algorithms. For packet forwarding, we compare with two algorithms.

- **RW**: With random walk (RW), a vehicle randomly selects a packet and forward to a next hop from the set of available neighbors.
- **N-RW**: N-RW [1] is an improved algorithm of the epidemic routing algorithm, in which the packet has n copies each takes a random walk.

For buffer allocation, we consider two algorithms.

- **FIFO**: For each received packet, FIFO randomly selects an AP to buffer it. If the buffer full, it drops the packet that enter the buffer earliest.
- **DIS**: For each packet, DIS selects an AP which has the minimum distance to the destination. If the buffer is full, it drops the packet with the maximum distance to its destination.

### B. Comparative Results

We compare our approach with other alternative methods and examine the effects of different parameters. For ease of presentation, we call our own approach OIA.

We first study the performance comparison as the packet arrival rate is varied. In Figure 9 and in Figure 10, we plot delivery ratio and delivery delay with different packet arrival rate, respectively. We find that both the global and the distributed algorithms outperform other algorithms. When the packet arrival rate grows higher, the delivery ratios of all algorithms drop. This is because the buffer size and number of APs are fixed and hence the total buffer is fixed. A higher packet arrival rate means more packets in the network which require more buffer to hold and higher link capacity. Therefore, the performances of all the algorithms fall on high packet arrival rate.

We then compare the algorithms as we vary the number of APs. In Figure 11 and Figure 12, our two algorithms have higher delivery ratios and smaller delays compared with other algorithms. More APs mean greater connectivity and more storage. Thus, the gap between our algorithms and other algorithms is smaller with more APs being deployed.

We next make performance comparison as we vary the number of vehicles. In Figure 13 and Figure 14, we find that the performances of our algorithms are better than all the other four algorithms as expected. When there are a small amount of vehicles in the network, because of poor connectivity the performance of all the algorithms is low. With the growing number of vehicles, better connectivity leads to better performance. Because of fixed buffer size and packet arrival rate, when the number of vehicles is large enough, the problem of buffer limitation is revealed and the performances drop. This result represents the influence of vehicle density, while it is hard to directly control the density when using real vehicle traces.

We finally make performance comparison as buffer size is varied. In Figure 15 and Figure 16, we find that the performance of all the algorithms increases when more buffers are used. In addition, the gap between our algorithms and other algorithms is smaller with a large buffer size because the problem of limited buffer is not as critical as that with small buffers.

### C. Impact of AP Deployment

We study the impact on the performance of our algorithms as the locations of APs are different. Although the optimal AP deployment is difficult to obtain, we use several different configurations for AP locations for this study.

We design four configurations of AP locations. The first configuration has locations randomly selected in the urban area of Shanghai. For the second, we select the locations only in the center area of the city (12% of the total area). Roads are dense in this area. For the third, locations are selected from the urban area excluding the city center. For the last configuration, we manually choose the locations in the whole urban area. During simulation, the buffer size at each AP is 24,000.
In Figure 17 and Figure 18, we can see that our algorithms outperform other algorithms for all the four configurations of the AP locations. Our algorithms have largest the performance advantage when the configuration with AP locations randomly selected. As a better configuration of AP locations is used, the performance advantage becomes smaller. These results show that our algorithms adapt to various AP deployments.

VII. CONCLUSION

In this paper we have investigated the optimal infrastructure-assisted routing for inter-vehicle delivery in vehicular networks. Our empirical study shows that deploying APs can improve data delivery in vehicular networks dramatically. In addition, buffer resources at the infrastructure are an important limiting factor. The empirical results suggest that effective buffer allocation is necessary. In this work we have formulated the coexisting problem of packet forwarding and buffer allocation as an optimization problem, which is shown to be a knapsack problem. We then designed the global algorithm and the distributed algorithm. Through comprehensive simulations, we demonstrate that our algorithms steadily outperform other algorithms under a wide range of network configurations.

ACKNOWLEDGMENT

This research is supported by Shanghai Pu Jiang Talents Program (10PJ1405800), Shanghai Chen Guang Program (10CG11), NSFC (No. 61170238, 60903190, 61027009, 60970106, 61170237), 973 Program (2005CB321901), MIIT of China (2009ZX03006-001-01), Doctoral Fund of Ministry of Education of China (20100073120021), 863 Program (2009A012201 and 2011AA010500), and HP IRP (CW267311), NSFC/RGC (N_HKUST610/11), HKUST (RPC11E929, SRFI11EG17-C and SB109/10.EG01-C), Huawei Technologies Co. Ltd. (HUAW18-15L0181011/PN), and Guangdong Bureau of Science and Technology (GDT11EG06). In addition, it is partially supported by the Open Fund of the State Key Laboratory of Software Development Environment (Grant No. SKLSDE-2010KF-04), Beijing University of Aeronautics and Astronautics.

REFERENCES