This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

Trajectory Improves Data Delivery in Urban Vehicular Networks

Yanmin Zhu, *Member*, *IEEE*, Yuchen Wu, and Bo Li, *Fellow*, *IEEE*

Abstract—Efficient data delivery is of great importance, but highly challenging for vehicular networks because of frequent network disruption, fast topological change and mobility uncertainty. The vehicular trajectory knowledge plays a key role in data delivery. Existing algorithms have largely made predictions on the trajectory with coarse-grained patterns such as spatial distribution or/and the inter-meeting time distribution, which has led to poor data delivery performance. In this paper, we mine the extensive datasets of vehicular traces from two large cities in China, i.e., Shanghai and Shenzhen, through conditional entropy analysis, we find that there exists strong spatiotemporal regularity with vehicle mobility. By extracting mobility patterns from historical vehicular traces, we develop accurate trajectory predictions by using multiple order Markov chains. Based on an analytical model, we theoretically derive packet delivery probability with predicted trajectories. We then propose routing algorithms taking full advantage of predicted probabilistic vehicular trajectories. Finally, we carry out extensive simulations based on three large datasets of real GPS vehicular traces, i.e., Shanghai taxi dataset, Shanghai bus dataset and Shenzhen taxi dataset. The conclusive results demonstrate that our proposed routing algorithms can achieve significantly higher delivery ratio at lower cost when compared with existing algorithms.

Index Terms—Vehicular networks, probabilistic trajectory, routing, prediction, Markov chain

1 Introduction

A vehicular network is a network of vehicles which communicate with each other via short-range wireless communications [1]. Vehicles can therefore communication with each other either directly when there meet each other or through multihop transmissions. Vehicles can act as powerful sensors and form mobile sensor networks [2, 3]. Vehicular networks have many appealing applications, such as driving safety [4], intelligent transport [5, 6], infrastructure monitoring [7, 8]. and urban monitoring [9].

Today 3G networks are getting more and more popular and ubiquitous access to 3G is possible. Moving vehicles can also communicate with each other via 3G. The 3G communication has advantage of ubiquitous access and shorter delay. Compared with multihop inter-vehicle communication, however, communication via 3G has limitations. First, although 3G communication becomes more and more cheaper, the cost of 3G communication is still high. In Shanghai, China, for example, the rate of 3G data communication is around 1 USD for only 30 MByte. By contrast, ad hoc vehicular communication is for free. Second, the bandwidth of inter-vehicle communication, realized through DSRC or 802.11, can be higher than that of 3G. Finally, many real-time applications, e.g., emergent message dissemination, are time critical and direct inter-vehicle communications may be more suitable.

Efficient inter-vehicle data delivery is of central importance to vehicular networks and such importance has been recognized by many existing studies [10-16]. In this paper we focus on such vehicular networks that are sparse and do no assume that all vehicles on the road are member nodes of the vehicular network. Such sparse vehicular networks feature infrequent communication opportunities. Inter-vehicle data delivery may introduce nonneligible delivery latency because of frequent topology disconnection of a vehicular network. Thus, we should stress that the inter-vehicle communication in vehicular network are suitable for *those applications which can tolerate certain delivery latency.* For example, in the context of urban sensing, vehicles continuously collect useful information, such as road traffic conditions and road closures. A vehicle may send a query for a specific kind of information and the one that has the information should respond the querying node with the data. Such communication require multi-hop data delivery in vehicular networks. Other examples of such applications include peer-to-peer file sharing, entertainment, advertisement, and file downloading.

There is a great deal of uncertainty associated with vehicle mobility. Vehicles move at their own wills. It is difficult, if not impossible, to gain the complete knowledge about the vehicular trace of future movement, i.e., the position of the vehicle at a given point in time. For routing in a vehicular network, a relay node must decide how long a packet should be kept and which node a given packet should be forwarded to. Existing study [17, 18] shows that it is possible to find an optimal routing path when the knowledge of future node traces is available, which is NP-hard though. However, it is impractical to have prior knowledge about future traces of nodes.

The knowledge of future vehicular trajectories plays a key role for optimal data delivery. Existing routing algorithms heavily rely on prediction of vehicle mobility. However, they have adopted only simple mobility patterns, such as the spatial distribution and inter-meeting time distribution, which support coarse-grained predictions of vehicle movements. Some algorithms [14, 19] assume random mobility in which vehicles move randomly in an open space or a road

• *Manuscript received on June 9, 2012, revised on November 28, 2012 and on March 25, 2013.*
• *Yanmin Zhu and Yuchen Wu are Shanghai Key Lab of Scalable Computing and Systems, and the Department of Computer Science and Engineering at Shanghai Jiao Tong University. Their emails are {yzhu, eaufavor}@sjtu.edu.cn.*
• *Bo Li is with the Department of Computer Science and Engineering at Hong Kong University of Science and Technology. His email is bli@cse.ust.hk.*

Digital Object Indentifier 10.1109/TPDS.2013.118 1045-9219/13/$31.00 © 2013 IEEE

network. This model is simple but far from the reality. Some other algorithms assume simple mobility patterns such as exponential inter-meeting times and regular spatial distributions. As a result, prediction results based on these simple patterns are of limited value to efficient data delivery in vehicular networks. In addition, many of existing algorithms ignore the fact that links in a vehicular network have unique characteristics [16]. On the one hand, a link is typically short-lived. This suggests that the capacity of the link is limited. Thus, the order for forwarding packets becomes important. On the other hand, links in a densely populated area may interfere with each other. This indicates that link scheduling becomes necessary.

To overcome the limitations of existing algorithms, this paper proposes an approach to exploiting the hidden mobility regularity of vehicles to predict future trajectories. By mining the extensive dataset of vehicular traces from more than 4,000 taxis in Shanghai, China, we show that there is strong spatiotemporal regularity with vehicle mobility. More specifically, our results based on conditional entropy analysis demonstrate that the future trajectory of a vehicle is greatly correlated with its previous trajectory. Thus, we develop multiple order Markov chains for predicting future trajectories of vehicles. With the available future trajectories of vehicles, we propose an analytical model and theoretically derive the delivery probability of a packet.

Since the optimal routing problem with given vehicle trajectories is still NP-hard, we develop an efficient global algorithm for computing routing paths when predicted trajectories are available. For more practical situations, we develop a fully distributed algorithm which needs only localized information. The two algorithms jointly consider packet scheduling and link scheduling. We evaluate the algorithms with extensive trace driven simulations, based on the trace dataset collected in Shanghai and Shenzhen. The results demonstrate that our algorithm considerably outperforms other algorithms in terms of delivery probability and delivery efficiency.

The rest of the paper is organized as follows. In Section 2, we formally state the problem. In Section 3 we reveal spatiotemporal regularity with vehicle mobility. Section 4 presents the analytical model and a global algorithm. Section 5 details the design of our distributed algorithm. We further discuss some design issues in Section 6. Performance results are presented in Section 7. We discuss related work in Section 8. The paper is concluded in Section 9.

2 System Model and Problem Statement

We have adopted a system of notations throughout the paper. A table of notations have been summarized, which can be found in the supplementary material.

2.1 System Model

The vehicular network is modeled as a set of nodes, N. When two nodes, i and j, are in the communication range (denoted by D), i.e., $d_{ij} < D$, there is a link between the two nodes and they can communicate with each other while the link exists.

The position of node n at time τ is denoted by $p_n(\tau)$. The time is slotted. Therefore, the trajectory of node n is a sequence of positions, denoted by

$$T_n = \left\langle p_n(0), p_n(1), \ldots p_n(\tau) \right\rangle. \quad (1)$$

The distance between two nodes, i and j, at time τ is denoted by $d_{ij}(\tau)$.

Links in the network are changing with the relative positions of the nodes. The set of all possible links at time τ is denoted by $L(\tau)$,

$$L(\tau) = \{ l_{i,j} \mid d_{i,j}(\tau) \le D; \forall i, j \in N \}. \quad (2)$$

Links are assumed to have the same capacity in theoretical analysis.

The vehicular network tries to deliver a set of packets, denoted by Φ, and the packets are of equal size. A packet has a source, $\delta(p)$, and a destination, $\psi(p)$. A packet is copied and forwarded from one node to another if there is a link between them. A packet group, $\theta(p)$, is introduced to denote all the copies of packet p in the network. The size of packet group, $|\theta(p)|$, increases when there is a new copy-forward process of packet p in the network.

2.2 Problem Formulation

The main goal of data delivery is to move the packets from their sources to respective destinations. The delivery performance objectives include delivery ratio, delay and efficiency. In the following, we first analyze the properties of individual packets, and then show the objectives in a global view.

The delivery probability is the chance of a packet successfully delivered from its source to its destination. The delivery probability of packet p, denoted by ρ_p, depends on the routing strategy, Υ, and the trajectories of nodes,

$$\rho_p = f_\Upsilon(\delta(p), \psi(p), \{T_n \mid n \in N\}). \quad (3)$$

With given trajectories, ρ_p can be calculated at the beginning and it does not change over time. But in the real world where the future traces are uncertain, this delivery probability can be considered as a random variable.

The delay of a packet, denoted by σ_p, is defined as the total transmission time spent for the network to deliver the packet to its destination. Analysis of delay is similar to that of delivery probability. The delay consists of two parts. One part is the time already spent, and the other is time to spent,

$$\sigma_p(\tau) = \tau + \Gamma_\Upsilon(\delta(p), \psi(p), \{T_n\}). \quad (4)$$

When $\sigma_p(\tau) = \tau$, the delivery of the packet is complete.

The cost of the transmission of a packet, denoted by ς_p, is defined as the total number of times that a packet is forwarded. A new copy is created when a packet is forwarded. Thus, the total number of copies indicates the delivery cost. The cost is therefore defined as

$$\varsigma_p = \left|\theta(p)\right| . \tag{5}$$

One of the common objectives of vehicular networks is to maximize *delivery ratio*. The delivery ratio is defined as the proportion of successfully delivered packets to the total packets to be transmitted. The number of total packets is often omitted in calculations because it is fixed and not affected by routing strategies. Thus, one objective can be given by

$$\max \sum\nolimits_{p\in\Phi} \mathbb{E}[\rho_p] . \tag{6}$$

Thus, the problem is to find the optimal routing of the packets through the vehicular network that meets (6).

Two other common objectives are minimization of delay and minimization of total cost, which are given respectively by

$$\min \sum\nolimits_{p\in\Phi} \mathbb{E}[\sigma_p] \text{, and } \min \sum\nolimits_{p\in\Phi} \mathbb{E}[\varsigma_p]. \tag{7}$$

The efficiency is defined as the ratio of total successfully delivered packets to the total cost, given by

$$\ell = \sum\nolimits_{p\in\Phi} \mathbb{E}[\rho_p] \Big/ \sum\nolimits_{p\in\Phi} \mathbb{E}[\varsigma_p] . \tag{8}$$

A high efficiency indicates that one algorithm achieves high delivery ratio at a low cost.

2.3 Complexity Analysis

Without loss of generality, we take the objective of maximizing the delivery ratio for example. Objective (6) can be written as,

$$\max \sum\nolimits_{p\in\Phi} \mathbb{E}[f_\Upsilon(\delta(p), \psi(p), \{T_n \mid n\in N\})] . \tag{9}$$

Theorem 1: *This routing problem with objective (9) when the vehicular traces and the set of packets to transmit are given is NP-hard.*

The proof of this theorem can be found in the supplementary material of the paper.

In practice, the complete knowledge is hard to be obtained because the future knowledge of vehicular traces is usually unavailable. Following this practical constraint, the trajectory of a node is divided into two parts: the past and the future, denoted by

$$\{T_n\} = \{T_n |_{\tau\le t}\} \cup \{T_n^{'} |_{\tau > t}\} . \tag{10}$$

Let $\lambda_t(p)$ to denote the current position of the packet. The objective at time t for routing becomes

$$\begin{aligned} &\max \sum\nolimits_{p\in\Phi} \mathbb{E}[\rho_p] \\ &= \max \sum\nolimits_{p\in\Phi} \mathbb{E}[f_\Upsilon(\lambda_t(p), \psi(p), \{T_n\}, \{T_n^{'}\})]. \end{aligned} \tag{11}$$

The second part of the trajectory is different from the first part. The first part has been fixed while the second part is not fixed and unknown at the time of being.

3 Spatiotemporal Regularity Analysis

In this section, we quantitatively reveal the spatiotemporal regularity with vehicle mobility. The quantitative analysis is based on mining the large dataset of vehicular traces of more than 4,000 taxis in Shanghai, China, which have a duration of more than two years. The GPS trace of a taxi, equipped with a GPS receiver, was collected by the vehicle periodically sending its instant position to a data collection center at an interval from 10 seconds to several minutes. The taxis operates in the whole urban area of Shanghai, which covers an area of over 120 square kilometers.

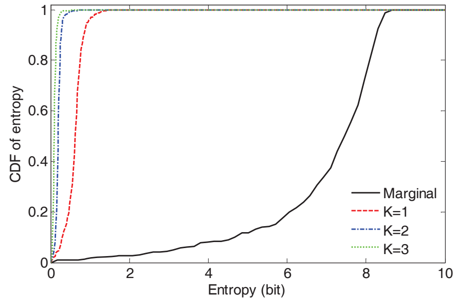


Figure 1: The CDFs of marginal entropy and conditional entropies of a vehicle positional state. The grid size is 200 m×200 m, and the number of vehicles is 500.

For simplification of discussion, the whole space is divided into Q small grids., and is denoted by S,

$$S = \{s_0, s_1, .., s_{Q-1} \mid s_i \cap s_j = \varnothing\} . \tag{12}$$

where Q is the total number of grids. The time is slotted. The location of a vehicle at a given time is considered as a random variable which takes state values from the grid space. Let S_i denote the random variable for vehicle i. We reveal the spatiotemporal regularity by computing the marginal and the conditional entropies of S_i given the previous K states.

For vehicle i, suppose we have observed its states for L time slots. The state sequence of the vehicle can be denoted by a vector $T_i =< s_0, s_1, .., s_{L-1} >$, where $s_j \in S, 0 \le j \le L$-1 is the positional state of vehicle i at time slot j. Suppose that s_j appeares o_j times in the vector of T_i, $0 \le j \le L$-1. Thus, the probability of vehicle i taking state s_j can be computed as o_j/L. Then, the marginal entropy of S_i is

$$H(S_i) = \sum\nolimits_{j=0}^{Q-1} (o_j / L) \times \log_2 \frac{1}{o_j / L} . \tag{13}$$

Next, we compute the conditional entropy of S_i given its immediately previous state S_i^1 which has the same distribution with S_i. The conditional entropy is,

$$H(S_i \mid S_i^1) = H(S_i, S_i^1) - H(S_i) . \tag{14}$$

To derive the conditional entropy, we have to derive the entropy of the joint random variable (S_i, S_i^1) By using the state sequence T_i, we derive a sequence of 2-tuples, $T_i^1 =< (s_0, s_1), (s_1, s_2), \ldots, (s_{L-2}, s_{L-1}) >$. By counting the occurrences of (s_{j-1}, s_j), denoted by $o_{j-1,j}$, we can get its probalibty. Thus, the joint entropy of the two dimensional random variable of (S_i, S_i^1) is,

$$H(S_i, S_i^1) = \sum\nolimits_{j=1}^{Q-1} \frac{o_{j-1,j}}{L-1} \times \log_2 \frac{1}{o_{j-1,j} / (L-1)} . \tag{15}$$

By generalizing the previous computation, we can compute the conditional entropy of S_i given its immediately previous K states $S_i^1, S_i^2, .., S_i^K$

$$H(S_i \mid S_i^1, S_i^2, ... S_i^K) . \tag{16}$$

In essence, by showing the marginal entropy we reveal spatial regularity, which characterizes the uncertainty of a vehicle residing a location in the space. By showing the conditional entropy given the previous states, we reveal the spatiotemporal regularity of vehicle mobility. This characterizes the uncertainty of a vehicle residing a specific location when its previous states are given. This demonstrates the correlation of a vehicle's positional states over different times.

In Figure 1, the cumulative distribution functions (CDFs) of the marginal and the conditional entropies for $K = 1, 2, 3$ are shown. The conditional entropies are significantly smaller than the marginal entropy. This implies that the uncertainty of the positional state becomes smaller when the previous states are known. We also find that when K becomes larger, the entropy continues to decrease. This suggests that more previous states help further reduce uncertainty. However, the improvement quickly stalls as K increases. This gives the guidance to the order selection for trajectory prediction using multiple order Markov chains.

4 Design of Global Algorithm

4.1 Overview

The global metric of a packet changes after intermediate transfers over time. Consider that a packet, p, is forwarded from node i to j at time t. After the forwarding, the increment in the global metric becomes

$$\Delta\Theta_{p,N.t} = \Theta_{p,N,t}(j) - \Theta_{p,N,t}(i)\,. \tag{17}$$

And the original objective becomes

$$\max \sum\nolimits_{p\in\Phi} \left(\sum\nolimits_{\tau=0}^{t-1} \Delta\Theta_{p,N,\tau} + \Delta\Theta_{p,N,t}\right). \tag{18}$$

Since the sum of increments before time t has been fixed, it can further be written as

$$\max \sum\nolimits_{p\in\Phi} \Delta\Theta_{p,N,t}\,. \tag{19}$$

Item $\Delta\theta$ can be considered as the current metric for packet p.

At time t, there may exist a number of links that may inference with each other, and it is impossible for all the links to be active simultaneously. The routing algorithm must schedule the links, i.e., choose a subset of links from all possible links to maximize the sum of current metrics. Thus, the objective becomes

$$\max \sum\nolimits_{p\in\Phi,\ l\in L(t)} \left(\Delta\Theta_{p,N,t} \times l_{i,j}\right). \tag{20}$$

The optimal routing performs two tasks, i.e., link scheduling and packet scheduling. Before the routing decision can be made, we have to obtain the metric increment of every packet over all possible transfers. When all the increments are available, the optimal routing chooses such a set of links and a set of packet transfers that meet (20).

We should emphasize that even when all $\Delta\theta_{p,N,t}$ are available, finding the best links and packet transfers is still an NP-hard problem. A brief proof is as follows. This problem can be considered as a weighted maximum independent set problem that has been proved to be NP-hard. To solve this problem, a number of existing heuristic algorithms [20] can be used, in which $\Delta\theta_{p,N,t}$ are considered as link weights.

As mentioned before, we take delivery probability maximization as example. The delivery probability of packet, ρ_p, depends on the encounter probability of every two nodes, denoted by $\varepsilon_{i,j}, \forall i,j \in N$. In the following we derive the relationship between metric increments and encounter probability.

The probability for the packet to be delivered in no more than one hop is

$$\rho_p^1 = \varepsilon_{\lambda(p),\psi(p)}\,. \tag{21}$$

Then, for a two hop delivery with the two-hop route of $< \lambda(p), n_1, \psi(p) >$, the probability is

$$\rho_p^{<\lambda(p),n_1,\psi(p)>} = \varepsilon_{\lambda(p),n_1} \times \varepsilon_{n_1,\psi(p)}\,. \tag{22}$$

Thus, the probability for packet being delivered in two hops is

$$\begin{aligned}\rho_p^2 &= 1 - \prod\nolimits_{n\in N, n\neq\lambda(p),\psi(p)} \left(1 - \rho_p^{<\lambda(p),n,\psi(p)>}\right)\\ &= 1 - \prod\nolimits_{n\in N, n\neq\lambda(p),\psi(p)} \left(1 - \varepsilon_{\lambda(p),n} \times \varepsilon_{n,\psi(p)}\right).\end{aligned} \tag{23}$$

The delivery probability of the packet with the h-hop route $< \lambda(p), n_1, .., n_{h-1}, \psi(p) >$ is

$$\rho_p^{<\lambda(p),n_1,..,n_{h-1},\psi(p)>} = \varepsilon_{\lambda(p),n} \times \left(\prod\nolimits_{i=1}^{h-2} \varepsilon_{n_i,n_{i+1}}\right) \times \varepsilon_{n_{h-1},\psi(p)}\,. \tag{24}$$

Then the set of all h-hop routes (denoted by Ξ) which starts from λ and ends at ψ is denoted by

$$\Xi^h(\lambda,\psi) = \{< \lambda, n_1, .., n_{h-1}, \psi >|\ n_i \in N; n_i \neq \lambda, \psi\}. \tag{25}$$

Thus, the h-hop delivery probability is calculated by

$$\rho_p^h = 1 - \prod\nolimits_{r\in\Xi^h(\lambda(p),\psi(p))} \left(1 - \rho_p^{\ r}\right). \tag{26}$$

Therefore, the total delivery probability is

$$\rho_p = 1 - \prod\nolimits_{0<h<H} \left(1 - \rho_p^h\right). \tag{27}$$

Constant H acts as the hop limit.

The previous analysis indicates that encounter probability is the key to computing the overall delivery probability of a packet. Since it is impossible to know future movements of vehicles, the knowledge of encounter probability of each pair of vehicles is not immediately available.

Fortunately, we have observed that there is strong spatiotemporal regularity with vehicle mobility. Based on this observation, we propose the notion of mobility patterns to characterize this regularity. With the mobility pattern, we are enabled to predict the trajectory of a vehicle. With trajectories of vehicles, we can effectively compute the encounter probability of two vehicles. Let the mobility pattern of node n denoted by MP_n. We develop the following mobility pattern for characterizing the spatiotemporal regularity of vehicle mobility. For vehicle i, its mobility pattern is defined as, $M_i = (H, F)$ where H is a vector of past positional states, $H =< H_0, H_1, .., H_{K-1} >$, $K \geq 1$, and F is the future positional state. The mobility pattern characterizes the frequencies of F following H in the vehicle's traces by computing the probability distribution of the two-dimensional random variable M_i. This can be

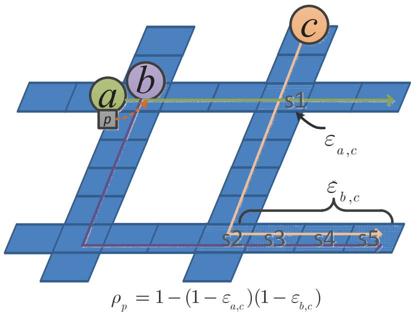


Figure 2: An illustrating example of calculation for encounter probability and delivery probability.

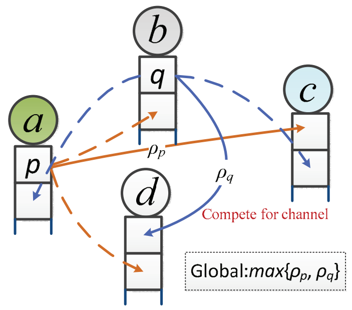


Figure 3: An illustrating example with four nodes for explaining the global algorithm. The links formed among node *a*, *b*, *c* should compete for the media.

achieved by analyzing the historical traces of the vehicle, as discussed in Section 3.

4.2 Prediction of Future Trajectories

Predicting the future trajectory of vehicle i at time t is to compute the future trajectory, $T_i|_{\tau>t}$, given the trajectory before t, $T_i|_{\tau<t}$. A trajectory, T, can be described by a sequence of positional states,

$$T = < s^0, s^1, \cdot\cdot, s^\tau, \cdot\cdot > . \tag{28}$$

Because of the uncertainty of vehicle mobility, there may exist a number of possible future trajectories.

Definition 1*. The set of all possible trajectories of node n is defined as a trajectory bundle, denoted by* TB_n*, which can be characterized by*

$$TB_n = < D^n(1), D^n(2), \cdot\cdot, D^n(\tau), \cdot\cdot > . \tag{29}$$

where $D^n(\tau)$ is the probability distribution of spatial states at future time τ.

$$D^n(\tau) : P_s^n(\tau) \mid s \in S . \tag{30}$$

where $P_s^n(\tau)$ is the probability of the node n appearing in state s at time τ.

To predict the trajectory of a vehicle, it is essentially to calculate the trajectory bundle of the vehicle. For trajectory prediction, we develop multiple order Markov chains for predictions. The key is to establish the matrix of transition probabilities. The transition probabilities are computed on an individual vehicle's basis, since different vehicle possesses different regularities.

For a vehicle, we can easily derive its transition matrix (denoted by X) from its mobility pattern. When we are using a K-order Markov chain, an element, $x_{ij} \in X$, represents the transition probability from H_i to F_j, where H_i is a sequence of positional states, $H_i =< h_0^i, h_1^i, .., h_{K-1}^i >$, and F_j is a single state. Then, $x_{ij} = Pr\,(H_i, F_j)$, where $Pr\,(\cdot)$ is the probability distribution function of the mobility pattern of the vehicle.

By applying the K-order Markov chain, we can compute the trajectory bundle as follows. Given the current trajectory of node n is $T_c^n =< s_{-K+1}, s_{-K+2}, \dots, s_0 >$, the initial distributions for T_c^n are

$$D^n(\tau) : \begin{cases} P_{s_\tau}^n(\tau) = 1 \\ P_s^n(\tau) = 0, s \neq s_\tau \end{cases}, (\tau \le 0) . \tag{31}$$

The distributions of spatial state at future times can be iteratively calculated. For a single trajectory (denoted by$<< s_1, .., s_K >, s >$, its probability is

$$P^n_{<s_1,..,s_K>,s}(\tau) = x_{<s_1,..,s_K>,s} \times \prod_{i=1}^{K} P_{s_i}^n(\tau - K + i) \ . \tag{32}$$

$P_{s_i}^n(t)$ are derived from the previous distributions and $x_{<s_1,..,s_K>,s}$ is defined in the transition matrix. Then, $D^n(\tau)$ can be derived by

$$D^n(\tau) : P_s^n(\tau) = \sum\nolimits_{all<s_i>} P_{<s_i>,s}{}^n(\tau),\ \tau > 0 . \tag{33}$$

4.3 Analysis of Delivery Probability

With the predicted trajectories, we are able to derive the encounter probabilities which are required for computing the eventual delivery probabilities.

Given the trajectory bundles of node i and j, $D^i(\tau)$and $D^j(\tau)$ are known. Let $\varepsilon_{i,j}(\tau)$ denote the encounter probability of the two nodes at time τ. It can be calculated by

$$\varepsilon_{i,j}(\tau) = \sum\nolimits_{s\in S} P_s^i(\tau) \times P_s^j(\tau) . \tag{34}$$

Then, the encounter probability, $\varepsilon_{i,j}$ is given by

$$\varepsilon_{i,j} = 1 - \prod_{\tau=t}^{T} (1 - \varepsilon_{i,j}(\tau)) , \tag{35}$$

where T denotes the max prediction range of time.

If the overall objective is to minimize the delivery delay, the estimated encounter time for the two nodes can be derived by

$$\eta_{i,j} = \sum_{\tau=t}^{T} \tau \times \varepsilon_{i,j}(\tau) \Big/ \sum_{\tau=t}^{T} \varepsilon_{i,j}(\tau) . \tag{36}$$

Therefore, the estimated delay of a packet can be calculated through trajectories in a similar way with delivery probability computation.

An example for calculating encounter probability and delivery probability is shown in Figure 2. There are three vehicles, a, b and c. Vehicle a encounters b while a wants to deliver packet p to c. To simplify the example, we consider only one predicted trajectory for each vehicle, which is shown as a line with an arrow along roads. Thus, the encounter probability between a and c, $\varepsilon_{a,c}$, can be calculated by $\varepsilon_{a,c} = P_{s1}^a \times P_{s1}^c$ where $s1$ is the overlapped grid between the trajectories of a and c and P_{s1}^a stands for the probability of a to appear in grid $s1$. There are several overlapped girds between the trajectories of b and c. The encounter probability, $\varepsilon_{b,c}$, can be calculated by $\varepsilon_{b,c} = 1 - \prod_{s\,\in\{s2,s3,s4,s5\}}(1 - P_s^b \times P_s^c)$.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

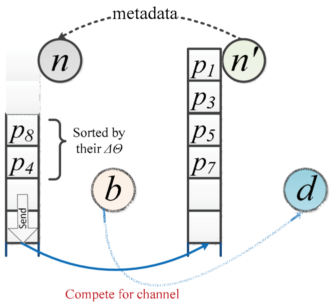


Figure 4: An illustrating example for explaining the distributed algorithm at node n. The link between n and n' may compete for the media with the link between b and d.

If a copies packet p to b, b contributes to packet delivery because it may encounter c as well. In this case, the delivery probability of packet p, when a and b forward it independently, is $\rho_p = 1 - (1 - \varepsilon_{a,c})(1 - \varepsilon_{b,c})$.

4.4 Algorithm Description

The global algorithm is described as follows. According to Subsections 4.2, each vehicle which has a transition matrix obtained from historical data can calculate its future trajectory bundle using its current position. With the trajectory bundles, the encounter probability of every two vehicles is known from (34). Then, the delivery probability of each packet in the network through each possible path can be calculated by (35) and (27). With these probabilities, the global algorithm solves an optimization problem to determine which the best next hop for each packet is. The algorithm repeatedly schedules the delivery until all the packets are delivered.

An illustrating example for explaining the global algorithm is given as follows. Suppose at a small area of the vehicular network, there are only four vehicle, a, b, c and d in each other's communication range as shown in Figure 3. Vehicle a has one packet p and vehicle b has one packet q. Both a and b want to deliver their packets to their destinations. The algorithm first calculates the delivery probabilities of packet p of each possible next hop (b, c, and d) and then finds that c is the best next hop. In the same way, d is the best next hop for packet q. Because the wireless links between a, c and b, d will interfere each other, the algorithm selects the packet with the largest delivery probability to be forwarded.

5 Design of Distributed Algorithm

The global algorithm requires the complete knowledge of all packets and vehicles. More specifically, for each packet, the knowledge of its source and destination, and the current position must be known. For each vehicle, the knowledge of its position and the mobility pattern must be known. These pieces of knowledge are not available in a distributed setting. Thus, we design a practical, distributed algorithm with which a vehicle requires only limited and localized knowledge.

5.1 Overview

The distributed algorithm consists of two fundamental building blocks. In the first block, the new objective for each individual vehicle is designed, since it is impractical for individual vehicles to compute the global objective. In the second block, we define the metadata for vehicles to exchange with each other at meetings.

In the distributed setting, the knowledge of nodes and packets is usually incomplete. Therefore, a global optimization (19) is hard for it is impossible to compute the global metric Θ and the current metric $\Delta\Theta$ (defined in (11) and (17)). In this case, we have to design a new metric based on which an individual vehicle makes routing decisions. Let the incomplete set of nodes and packets be denoted by Φ' and N', respectively. Then, the new objective becomes

$$\max \sum_{p\in\Phi'} \Delta\Theta_{p,N',t} \ . \qquad (37)$$

$\Delta\Theta_{p,N',t}$ becomes the new metric of packet p at time t, which is local and can be computed by individual vehicles for each packet. When making routing decisions, a node maximizes the sum of local metrics of all the packets it knows.

It would be better for a node to have more knowledge about nodes and packets. Since it is difficult for a node to have the complete knowledge, we design a distributed protocol for the nodes to exchanging information when they meet each other. By this way, the knowledge can be propagated throughout the network.

For exchanging information, we define metadata, which include two parts of information. The first part is about the mobility pattern and the most update position of each vehicle (time stamped). The second part is about the packets that the node carries. Note that for a relatively stable set of vehicles, the mobility pattern reflects the regularity of a vehicle's mobility. Thus, it is relatively stable and therefore it is no need to update the mobility patterns frequently.

5.2 Algorithm Description

As a distributed algorithm, each vehicle executes the routing algorithm independently. For each vehicle, a routine procedure is invoked each time it finds a new communication neighbor. For neighbor discovery, it is required that every vehicle periodically broadcasts hello messages so that other vehicles can discover it when it enters their communication ranges.

In the following we describe the procedure, supposing that vehicle n finds another vehicle, n', entering its communication range. The Pseudo code description of this procedure is shown in Figure 5.

Procedure 1 $(n, n', \Lambda_n, \mathcal{M}_n, \mathcal{M}_{n'}, t)$
Variables
n: the node; n': new neighbor; $\mathcal{M}_n$, $\mathcal{M}_{n'}$: metadata;
Φ_n: packets of node n; Λ_n: neighbors of n;
t: current time; N: total nodes

1. $\Lambda_n = \Lambda_n \cup \{n'\}$,
2. n exchanges metadata with n': $n \leftarrow \mathcal{M}_{n'}$, $n' \leftarrow \mathcal{M}_n$
3. n calculates $\Delta\Phi_{n'} = \Phi_n - \Phi_{n'}$ from $\mathcal{M}_n, \mathcal{M}_{n'}$
4. For every packet $p \in \Delta\Phi_{n'}$, calculate metric $\Delta\Theta_{p,N,t}(n') = \rho_p(n') - \rho_p(n)$
5. Sort all packets $p \in \Delta\Phi_i$ according to metric $\Delta\Theta_{p,N,t}(i)$ for all $i \in \Lambda_n$ in the decreasing order
6. Transmit the packets in this order

Figure 5: Pseudo code of the procedure of the distributed algorithm, which is executed upon each contact with a vehicle.

First, node n updates its neighbor set (Λ_n). Then, node n and n' exchange their metadata. The metadata of a node, i, denoted by $\mathcal{M}_i$, include two metadata sets, $\mathcal{P}_i$ and $\mathcal{V}_i$. Set $\mathcal{P}_i$ contains the metadata about the packets that node i carries, including identification and source-destination pair. Set $\mathcal{V}_i$ contains the metadata about the vehicles known to node i, including ID, most update position and mobility pattern.

Next, it recalculates the metrics for all the packets it carries, and sort the packets respect to the new metrics in the decreasing order. The packets will then be transferred in the sorted order. Note that if the node has already been transferring a packet, the transmission of this packet is not interrupted. After its completion, the packets shall be transmitted according to the new order. By ordering the packets, we are essentially doing the link scheduling in a distributed way. However, before a packet transfer can be started, a vehicle has to follow media access control to avoid potential collisions.

As packets are transferred between vehicles, the packet set, $\mathcal{P}_i$, is updated whenever the vehicle receives a new packet from its neighbors.

An illustrating example for explaining the distributed algorithm is shown in Figure 4. At first, the vehicles exchange their short metadata with each other. With these metadata and the metadata obtained from other vehicles in the previous rounds, vehicle a and c calculate the delivery probabilities of p and q, respectively. Then, they compete for the wireless media with each other following a media access control protocol. The winner sends the packet to the chosen next hop.

The distributed algorithm is efficient and its complexity is analyzed. The analysis can be found in the online supplementary material.

5.3 Other Design Issues

For the design of the distributed algorithm, there are some other issues, including use of historical data, division of grids, and privacy issue. Due to the page limit, we leave the handling of these design issues in the online supplementary material.

6 Performance Evaluation

In this section, we first present the methodology of performance evaluation, describe the settings of simulations, introduce the compared algorithms and finally present evaluation results.

6.1 Methodology and Settings

We evaluate our algorithms with the performance metric of delivery ratio, delay, cost and efficiency. These metrics have been defined in Section 2. We compare our algorithms with several other algorithms, which is to be introduced shortly.

The simulations are conducted with the three datasets of real GPS vehicular traces. One dataset includes the vehicular traces of more than 4,000 taxis collected in Shanghai, the second dataset is the traces of more than 2,000 buses in Shanghai and the third dataset consists of vehicular traces of more than 12,000 taxis collected in Shenzhen. The Shanghai dataset covers a duration of two years and the Shenzhen dataset a duration of one month. The whole urban area of Shanghai is 133 kilometers in length and 69 kilometers in width, and that of Shenzhen is about 27 kilometers in length and 97 kilometers in width.

The whole space is divided into grids, and the grid size is three kilometers by default and will be varied in the impact study of grid size. The communication range is 300 meters. We consider link interfaces, and each node can communicate with one neighbor at any time. We randomly select a subset of 400 taxis or buses from the complete trace for simulations.

For each packet, we randomly select its source and destination. The packets are injected at different times. Every packet has the same size and priority. The maximum hop and the maximum time-to-live (TTL) of each packet is set to 20 and 2 hours, respectively. The number of packets is varied from 100 to 800 to study different loads of the network.

The order of Markov chains K is set to two. From conditional entropy analysis, we have already found that a higher order beyond two gives little reduction in uncertainty. To build the mobility pattern for each vehicle, its vehicular trace of the past 15 days is used. Each data point is the average over five different runs of trace-driven simulations.

6.2 Compared Algorithms

Flooding [21], also known as epidemic routing, is a simple algorithm. Each node forwards all the packets it carries to any node it meets. This algorithm provides an upper bound on delivery ratio and a lower bound on delivery delay. It introduces very high cost, which is the major defect.

P-Random [18] is an opportunistic routing algorithm, which randomly decides whether to forward a packet to another node. In simulations, the probability is set to 0.4. This algorithm represents the algorithms without predictions.

MobySpace [14] is a routing algorithm for delay tolerant networks. It obtains the probability of each node residing in each possible location by analyzing the his-

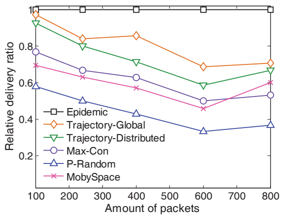


Figure 6: Delivery ratio vs. amount of packets, with Shanghai taxi dataset.

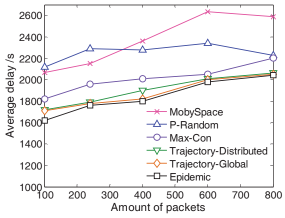


Figure 7: Average delay vs. amount of packets, with Shanghai taxi dataset.

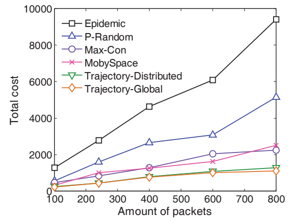


Figure 8: Total cost vs. amount of packets, with Shanghai taxi dataset.

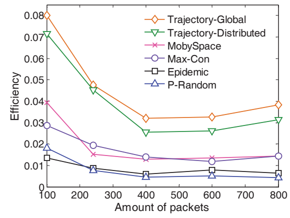


Figure 9: Efficiency vs. amount of packets, with Shanghai taxi dataset

torical trace data. It assumes that every node has the full knowledge of such probabilities of all vehicles. It then computes the contact probability of every two vehicles based on the probability distribution, and then routing decisions can be made.

Max-Contribution [16] is a routing algorithm using a simple mobility pattern that only considers the inter-meeting times of exponential distribution. This pattern makes prediction independent of the current location of a vehicle.

6.3 Comparative Results

We present performance results of our algorithms compared against other algorithms. First, we show the results using the Shanghai dataset of real taxi traces. To characterize the mobility pattern, the vehicular trace of the recent 15 days is used.

We vary the amount of packets and compare the six algorithms in terms of delivery ratio, average delay and total cost under different loads of network. For a given setting of amount of packets, the same set of packets and the set of nodes are used for all the six algorithms.

Since the delivery ratio is largely affected by the simulated time length, we use the metric of relative delivery ratio instead of bare delivery ratio, which is the delivery ratio of each algorithm normalized by that of Flooding. In Figure 6, the performance of the six algorithms in terms of relative delivery ratio is shown. We can see that our algorithms perform better than P-Random, MobySpace and Max-contribution. Among the six algorithms, Flooding performs the best, as expected. In general, the algorithms that use predictions produce better delivery ratios than the algorithms that make no predictions. Our algorithms are better than MobySpace and Max-Contribution because our algorithms use not only the historical traces information but also the current state and the previous states. Predicted trajectories of vehicles help to find better routing paths. The global algorithm is better than the distributed one since it has the global, complete network information. We can also find that when the amount of packets increases, the overall delivery ratios of all algorithms decrease. The reason is that the overall capacity of the network is limited. By injecting more packets, the packets may compete for network resources and as a result, fewer packets can be delivered in the end.

In Figure 7, average delay against amount of packets is plotted. Our algorithms have a lower delay than P-Random, MobySpace and Max-Contribution. As expected, Flooding has the smallest delay. The average delays of our algorithms are slightly larger than that of flooding. This performance gain of our algorithms is mainly due to the fact that routing paths with high delivery probabilities usually lead to shorter delays. Since our algorithms can select routing paths of high delivery probabilities, the resultant delay is low.

Figure 8 shows the costs of the six algorithms. We can find that our algorithms have lower costs, better than all the other algorithms. The main reason is that by effectively predicting the trajectories of vehicles, we only consider the paths that lead to eventual delivery with high probability. Therefore, unnecessary packet transfers are greatly reduced.

In Figure 9, we compare the efficiencies of the six algorithms. We can see that our algorithms have higher efficiency, better than all the other algorithms. Flooding and P-Random have a similar efficiency and are much worse than the rest four. This is due to the blindness of Flooding and P-Random when they are making transfer decisions. Max-Contribution and MobySpace have larger efficiency than Flooding and P-random, but have lower efficiency than our algorithms, since Max-Contribution and MobySpace merely uses a simple mobility pattern of inter-meeting time.

6.4 Effect of Trace Datasets

To validate the performance of the proposed algorithms, we also show the performance results with the Shanghai bus dataset and Shenzhen taxi dataset. It should be noted that the whole area of Shenzhen is only one third of that of Shanghai. Thus, the Shenzhen dataset provides a better network connectivity than the Shanghai taxi dataset does. The Shanghai bus dataset has the best network connectivity because buses tend to run on the same routes and the buses are more densely distributed along the routes.

In Figure 10 and Figure 14, the performance of relative delivery ratio for the six algorithms is shown. We can see the similar trend as shown in Figure 6 that the proposed global algorithm outperforms the distributed algorithm and that both of the algorithms achieve better delivery ratio performance than the rest four other algorithms. On average, the distributed algorithm achieves several times larger delivery ratio than P-Random and is about 75% of the delivery ratio achieved by the Flooding. However, we also find that the impacts of

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

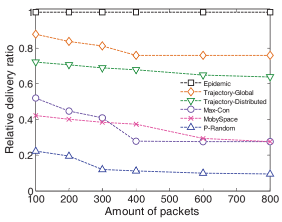


Figure 10: Delivery ratio vs. amount of packets, with Shanghai bus dataset.

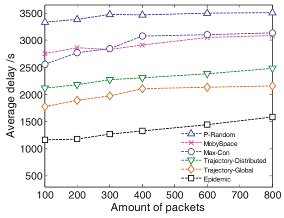


Figure 11: Average delay vs. amount of packets, with Shanghai bus dataset.

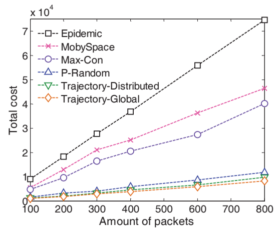


Figure 12: Total cost vs. amount of packets, with Shanghai bus dataset.

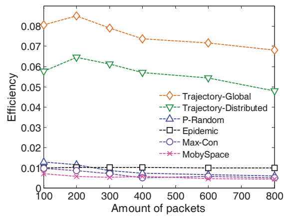


Figure 13: Efficiency vs. amount of packets, with Shanghai bus dataset.

packet amount in Shanghai bus dataset and Shenzhen taxi dataset are not as large as in Shanghai taxi dataset. This is because the better network connectivity in both Shanghai bus dataset and Shenzhen taxi dataset leads to a higher network capacity.

The performance of average delay for the six algorithms is shown in Figure 11 and Figure 15. We can see that the result is consistent with that in Shanghai taxi dataset. The proposed algorithms produce an average delay that is close to the delay of Flooding. The average delay of P-Random is much larger than the rest of algorithms. We can see the average delay for all algorithms increase slightly as the packet amount increases, since the delivery capability is limited.

In Figure 12 and Figure 16, the total cost for the six algorithms is shown. It is shown that the proposed algorithms introduce much lower cost than Epidemic, P-Random, MobySpace and Max-Con. In addition, as the packet amount increases, the costs of the proposed global and distributed algorithm increase with a lower rate than the rest algorithms.

In Figure 13 and Figure 17, we compare the efficiency performance of the six algorithms. The global algorithm achieves the highest efficiency and the distributed algorithm is the second best among all the algorithms.

In summary, the performance results using the Shanghai Bus and Shenzhen Taxi dataset are consistent with those using the Shanghai Taxi dataset. This confirms that the proposed algorithms can still achieve good performance for different types of vehicles and for different densities of vehicles.

Note that taxis and buses cannot represent all types of vehicles. Mobility patterns of taxis may be different from those of other type of vehicles. We do not have access to traces of private vehicles. Thus, we are not able to show the performance of our algorithm when traces of private cars are used. However, our algorithm should work well if there exist mobility patterns for private cars.

6.5 Additional Evaluation Results

It is clear that other important system parameters including grid size and length of historical trace make impact on the performance of our algorithms. We have also performed additional simulations to study such impact of the two system parameters. The evaluation results can be found in the online supplementary material of the paper.

7 Related Work

We classify existing routing algorithms for vehicular networks into two categories.

For first class of routing algorithms, there is no need to make estimation or prediction on vehicular traces. They take advantage of navigation systems, in which vehicular traces are priori known before they begin to travel [10]. This model and assumption work well in traditional Delay Tolerant Networks, such as interplanetary networks or satellite networks. The nodes in such networks have simple and stable mobility patterns. These algorithms can only be applicable to those vehicular networks, in which drivers must tell the navigation system the destinations before their journeys and must follow the lead of the system [12].

Some work assumes partial knowledge about the vehicular traces. In TBD [22], a routing problem is considered to deliver a packet from a mobile vehicle to a static AP. The movement path of a vehicle from the source to the destination is known by itself through a navigation system. It then proposes a link model for computing the expected delay of a road segment, based on which the end to end delay can be computed.

The second class of routing algorithms makes estimations about routing metrics, since there is no further information about node future traces. Delegate Forwarding [23] demonstrates that forwarding with a metric of good quality can reduce network cost. This is done by the strategy of only forwarding packets to those nodes which lead the packet to the highest quality. RAPID [13] treats the routing of DTNs as a resource allocation problem and proposes a routing protocol to maximize the performance of a specific routing metric. The algorithms mentioned above care about calculation of metrics by estimating delay or inter-meeting time based on the simple model of exponential inter-meeting times. Max-Contribution [16] considers the joint optimization of link scheduling and packet forwarding. It illustrates that this optimization is better than former algorithms. MobySpace [14] characterizes mobility patterns with appearance frequencies of nodes over a space. The algorithm indicates that nodes with the similar frequencies of appearances have more opportunities to connect. This brings benefits to routing in DTNs.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication.

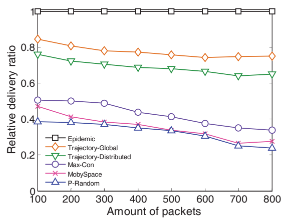


Figure 14: Delivery ratio vs. amount of packets, with Shenzhen taxi dataset.

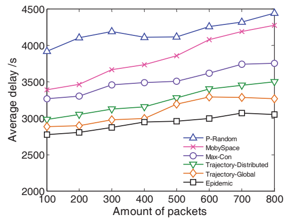


Figure 15: Average delay vs. amount of packets, with Shenzhen taxi dataset.

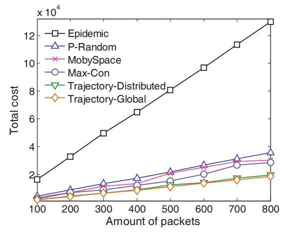


Figure 16: Total cost vs. amount of packets, with Shenzhen taxi dataset.

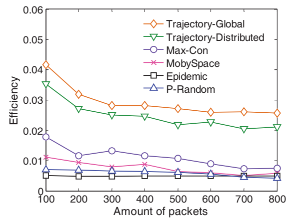


Figure 17: Efficiency vs. amount of packets, with Shenzhen taxi dataset.

8 Conclusion

In this paper, we demonstrate the strong spatiotemporal regularity with vehicle mobility by entropy analysis with the extensive datasets of real taxi and bus traces. By developing multiple order Markov chains, we predict vehicle trajectories. Based on the analytical model, we derive delivery probability with the predicted probabilistic trajectories. The proposed algorithms take full advantage of predicted probabilistic vehicle trajectories. Extensive simulations based on the large datasets of real vehicular GPS traces. Performance results verify that our algorithm outperforms other algorithms. This demonstrates that predicted trajectories do help data delivery in vehicular networks.

Acknowledgements

This research is supported by Shanghai Pu Jiang Talents Program (10PJ1405800), Shanghai Chen Guang Program (10CG11), NSFC (No. 61170238, 60903190, 61027009, 60933011, 61202375, 61170237), MIIT of China (2009ZX03006-001-01), Doctoral Fund of Ministry of Education of China (20100073120021), National 863 Program (2009AA012201 and 2011AA010500), HP IRP (CW267311), SJTU SMC Project (201120), STCSM (08dz1501600, 12ZR1414900), Singapore NRF (CREATE E2S2), NSFC/RGC (N-HKUST610/11), HKUST (RPC11EG29, SRFI11EG17-C and SBI09/10.EG01-C), and Guangdong Bureau of Science and Technology (GDST11EG06).

References

[1] F. Bai, D. D. Stancil, and H. Krishnan, "Toward understanding characteristics of dedicated short range communications (DSRC) from a perspective of vehicular network engineers," *Proc. ACM MOBICOM*, 2010.

[2] Z. Li, Y. Liu, M. Li, J. Wang, and Z. Cao, "Exploiting Ubiquitous Data Collection for Mobile Users in Wireless Sensor Networks," *IEEE TPDS*, vol. 24(2), pp, 312-326, 2013.

[3] M. Li and Y. Liu, "Rendered Path: Range-Free Localization in Anisotropic Sensor Networks with Holes," *IEEE/ACM Transactions on Networking*, vol. 18(1), pp, 320-332, 2010.

[4] L. Chisalita and N. Shahmehri, "A peer-to-peer approach to vehicular communication for the support of traffic safety applications," *Proc. The 5th IEEE Conference on Intelligent Transportation Systems*, 2002, pp. 336-341.

[5] Z. Li, Y. Zhu, H. Zhu, and M. Li, "Compressive Sensing Approach to Urban Traffic Sensing," *Proc. IEEE ICDCS*, 2011.

[6] H. Zhu, Y. Zhu, M. Li, and L. M. Ni, "SEER: Metropolitan-scale Traffic Perception Based on Lossy Sensory Data," *Proc. IEEE INFOCOM*, 2009.

[7] J. Eriksson, L. Girod, B. Hull, R. Newton, S. Madden, and H. Balakrishnan, "The Pothole Patrol: Using a Mobile Sensor Network for Road Surface Monitoring," *Proc. ACM MobiSys*, 2008.

[8] P. Gibbons, B. Karp, Y. Ke, S. Nath, and S. Seshan, "Irisnet: An architecture for a worldwide sensor web," *IEEE Pervasive Computing*, pp, 22-33, 2003.

[9] U. Lee, B. Zhou, M. Gerla, E. Magistretti, P. Bellavista, and A. Corradi, "Mobeyes: Smart Mobs for Urban Monitoring with a Vehicular Sensor Network," *IEEE Wireless Communications*, vol. 13(5), pp, 52-57, 2006.

[10] I. Leontiadis and C. Mascolo, "Geopps: Geographical opportunistic routing for vehicular networks," *Proc. IEEE WoWMoM*, 2007, pp. 1-6.

[11] G. Marfia, M. Roccetti, C. E. Palazzi, and A. Amoroso, "Efficient vehicle-to-pedestrian exchange of medical data: an empirical model with preliminary results," *Proc. ACM MobiHoc Workshop on Pervasive Wireless* 2011.

[12] I. Leontiadis, P. Costa, and C. Mascolo, "Extending access point connectivity through opportunistic routing in vehicular networks," *Proc. IEEE INFOCOM*, 2010.

[13] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN routing as a resource allocation problem," *Proc. ACM SIGCOMM*, 2007.

[14] J. Leguay, T. Friedman, and V. Conan, "DTN Routing in a Mobility Pattern Space," *Proc. ACM SIGCOMM Workshop on Delay-tolerant Networking*, 2005, pp. 276-283.

[15] J. Krumm and E. Horvitz, "Predestination: Where Do You Want to Go Today?," *Computer*, vol. 40(4), pp, 105 - 107, April 2007.

[16] K. Lee, Y. Yi, J. Jeong, H. Won, I. Rhee, and S. Chong, "Max-contribution: On optimal resource allocation in delay tolerant networks," *Proc. IEEE INFOCOM*, 2010.

[17] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "Maxprop: Routing for vehicle-based disruption-tolerant networks," *Proc. IEEE INFOCOM*, 2006.

[18] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," *Proc. ACM SIGCOMM* 2004.

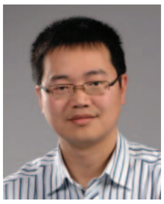
[19] Z. Chen, H. Kung, and D. Vlah, "Ad hoc relay wireless networks over moving vehicles on highways," *Proc. ACM Mobihoc*, 2001, p. 250.

[20] Y. Yi, A. Proutière, and M. Chiang, "Complexity in wireless scheduling: Impact and tradeoffs," *Proc. ACM MobiHoc*, 2008, pp. 33-42.

[21] R. Ramanathan, R. Hansen, P. Basu, R. Rosales-Hain, and R. Krishnan, "Prioritized epidemic routing for opportunistic networks," *Proc. ACM MobiSys workshop on Mobile Opportunistic Networks (MobiOpp)*, 2007, p. 66.

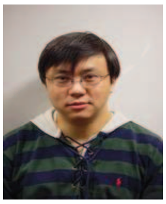
[22] J. Jeong, S. Guo, Y. Gu, T. He, and D. Du, "TBD: Trajectory-based data forwarding for light-traffic vehicular networks," *Proc. IEEE ICDCS*, 2009.

[23] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation forwarding," *Proc. ACM MobiHoc*, 2008, pp. 251-260.



Yanmin Zhu (M'02) is an Associate Professor in the Department of Computer Science and Engineering at Shanghai Jiao Tong University. He obtained his PhD in computer science from the Department of Computer Science and Engineering at the Hong Kong University of Science and Technology in 2007. His research interests include vehicular networks, sensor networks and mobile computing. Prior to joining Shanghai Jiao Tong University, he was a research associate with the Department of Computing at the Imperial College London. He is a member of the IEEE and the IEEE Communications Society.

Yuchen Wu obtained his bachelor degree from the Department of Computer Science and Engineering at Shanghai Jiao Tong University in 2010. He is a master student in the Department of Computer Science and Engineering at Shanghai Jiao Tong University in 2010. His research interests include vehicular networks and mobile computing.



Bo Li (F'11) is a professor in the Department of Computer Science and Engineering, Hong Kong University of Science and Technology. He is a Cheung Kong Chair Professor in Shanghai Jiao Tong University, China, and the Chief Technical Advisor for China Cache Corp. (NASDAQ:CCIH). He was previously with IBM Networking System Division, Research Triangle Park, and an adjunct researcher at Microsoft Research Asia. His recent interests include: large-scale content distribution in the Internet, Peer-to-Peer media streaming, the Internet topology, cloud computing, green computing and communications. He received his B. Eng. Degree in the Computer Science from Tsinghua University, Beijing, and his Ph.D. degree in the Electrical and Computer Engineering from University of Massachusetts at Amherst.