

# Exploring the Hidden Connectivity in Urban Vehicular Networks

Kebin Liu<sup>1,2</sup>, Mo Li<sup>3</sup>, Yunhao Liu<sup>1,2</sup>, Xiang-Yang Li<sup>1</sup>, Minglu Li<sup>4</sup> and Huadong Ma<sup>5</sup>

<sup>1</sup> TNLIST, School of Software, Tsinghua University

<sup>2</sup> Department of Computer Science, Hong Kong University of Science and Technology

<sup>3</sup> School of Computer Engineering, Nanyang Technological University, Singapore

<sup>4</sup> Department of Computer Science, Shanghai Jiaotong University

<sup>5</sup> Department of Computer Science, Beijing University of Posts and Telecommunications

**Abstract**—The high mobility of VANET makes information exchange across the network excessively difficult. Traditional approaches designed for stationary networks are not applicable due to the high dynamics among the nodes. Applying the routing techniques tailored for general mobile networks inevitably brings huge traffic burden to the crowded urban VANET and leads to low efficiency. To make the information exchange fluent and efficient, we explore the unique features of the urban VANET. By exploring the invariants in the mobile network topology, we are able to efficiently manage the information on top of the “*intersection graph*” transformed from the underlying network of road segments in the urban area. Our approach can thus achieve efficient query dissemination and data retrieval on this information organization. We intensively investigate and analyze a trace that records the movement of more than 4000 taxis in the urban area of Shanghai City over several months. We grasp the key impact of the fundamental factors that affect the VANET behaviors and accordingly develop tailored techniques to maximize the performance of this design. Experimental results validate the effectiveness and efficiency of our design.

## I. INTRODUCTION

Recent advances in various Dedicated Short Range Communication (DSRC) techniques provide easy and effective communication between vehicles with high mobility. A Vehicular Ad Hoc Network (VANET) is a type of Mobile Ad Hoc Network (MANET) where intermediate nodes are moving vehicles connected through DSRC to form a multi-hop network. Drivers are provided critical information from VANET, which helps their safe and comfortable driving, such as emergency warning of approaching vehicles, intersection collision avoidance, news group broadcast, opportunistic access into internet, etc. One specific application we envision is the information exchange over VANET in the urban area. Urban vehicular networks are quite different from those with inter-city environment in many aspects, such as the traffic density, the complexity of road network structure, and the like. Drivers on different vehicles exchange information, such as witness of accidents, emergency warnings, traffic conditions, among the citywide network to assist driving. Each vehicle acts as an information provider that observes surrounding environment and

contributes useful information to the VANET. Meanwhile each vehicle is also an information consumer that queries the items of particular interest and retrieves them from the VANET.

There have been intensive studies that investigate the problem of information exchange within a wireless multi-hop network. Nevertheless, the high mobility of VANET makes it excessively difficult to perform efficient information exchange in the highly dynamic network. The traditional routing approaches put heavy efforts to maintain an end-to-end connection between communicating nodes or construct an in-network structure to organize the information storage and dissemination. They suffer from the frequent network alternations in the urban VANET environment. Even those frameworks specifically designed for general mobile networks inevitably bring huge network burden for maintaining the ordered structures on top of the unordered network, leading to poor efficiency. Those approaches, like Dynamic Source Routing (DSR) [14], Ad Hoc On-Demand Distance Vector (AODV) routing [19], or recently proposed Weak State Routing (WSR) [2], largely rely on flooded status information across the entire network for establishing the routes or discovering the resources. The cost of end-to-end communication significantly increases due to the node mobility and path dynamics.

When vehicles in the network are sparse, the VANET falls into a disruptive network. While there have been tremendous studies for the Disruption Tolerant Network (DTN), the intrinsic requirement of efficient information exchange makes it difficult to directly borrow from them. On one hand, users are quite sensitive to the efficiency of information search and the latency of data retrieval, and on the other hand different from that in inter-city vehicular networks, plenty of network connectivity in most of the urban area allows us much more optimistic ways of utilizing the network resources instead of physically carry the information with the vehicles.

In this work, we explore the unique features of the urban VANET and design a tailored approach such that we make the information exchange fluent and efficient with largely reduced network overhead. Our design is based on two observations on the urban VANET characteristics. (1) Although each node in the network is highly mobile, its mobility pattern is restricted



Figure 1: An instant record of taxis in the urban area.

by the street map of the urban area, as vehicles can only move along the road segments. This allows us to design corresponding networking paradigm against such an invariant of the mobility model. (2) The communicational signals of vehicles are highly directional subject to the road orientations. As recently reported by J. Robinson et al. [21], wireless signals are substantially obstructed from the building blocks along the road segments. This allows us to manage the information network more efficiently on top of the network of road segments. Above two observations suggest us to utilize the inner connectivity of the underlying road network that regulates the vehicles flowing on.

The main idea of our design is that we embed the mobile VANET flows into the static road map which is pre-loaded in all internal vehicles. We label the road segments and organize the storage of useful information through the static connectivity of road segments in the map instead of directly maintaining it on the mobile network above. By exploring the possibility of information exchange at the intersections of road segments where the vehicle flows on different roads join with each other, we achieve efficient query dissemination as well as data retrieval.

Nevertheless, enabling an applicable design that works in the real circumstance needs to deal with many practical conditions. What is the connectivity of the VANET in the urban environment and how long the connected links between moving vehicles can persist? What is the node density and how such a parameter varies across different parts of the network? Is there a constant distribution of node density over time?

To better understand those practical conditions and how they react on the urban VANET, we intensively investigate and analyze a trace that records the movement of more than 4000 taxis in the urban area of Shanghai City over several months (Fig. 1 exhibits a part of the records which illustrates the traffic distribution on a road map. Each blue dot depicts a taxi on the road). We conduct a trace based study on those fundamental factors that affect the VANET behaviors. We grasp the key impact of them and accordingly develop tailored techniques to maximize the performance of our design. Our key experimental findings are as follows:

- The network connectivity is a direct factor that affects the performance of the information delivery. The average node degree reflects the actual network connectivity and thus

should be considered in designing the information exchange scheme.

- The vehicle density intensively impacts on the efficiency of information delivery and it varies across areas. However, the global distribution of the vehicle density is stable along with time, which enables us to design an adaptive approach that diversifies over different areas of different vehicle densities.
- The vehicle mobility is highly restricted by the underlying road map. The chances that vehicles on different road segments meet at the road crosses are substantial and can be used to improve the efficiency of information delivery.

The rest of this paper is organized as follows. We briefly introduce the related work in Section 2. In Section 3 we describe our design principles and in Section 4 we present our trace based study. We conclude this work in Section 5.

## II. RELATED WORK

VANET has emerged as a promising networking style that facilitates the information share among the moving vehicles. The 75 MHz of Dedicated Short Range Communications (DSRC) [1] spectrum at 5.9 GHz has been allocated for vehicular communications in many countries including the U.S. In the DieselNet project researchers have deployed a test bed consisting of 40 buses in Amherst. They have studied both the routing protocols [5] and connectivity models [28] in such a bus-to-bus network. The recently proposed Cabernet [7] aims to provide efficient WiFi-based data delivery for vehicles. There are also studies that explore the vehicle mobility for data aggregation within VANET [27].

VANET is by nature a special type of mobile ad hoc networks. There have been many schemes proposed to facilitate the information exchange and data routing [30] in MANET [4]. To deal with the network dynamics, reactive approaches such as DSR [14] and AODV [19] work in an on-demand manner, avoiding constant updates on topology information. While route discovery is highly flooding based, those approaches are inefficient for vehicular networks where the end-to-end connections are vulnerable and route discovery for new information or new vehicles is frequently launched. Instead of blindly flooding, Weak State Routing [2] exploits a dissemination and search framework. Each node periodically disseminates its state information (location and ID) to the network. Packets are then forwarded with the random directional walk routes guided occasionally by the weak state information. However, the usage of random walks leads to long path lengths (high delay) and the packet delivery ratio cannot be guaranteed. When applied to VANET environment, existing routing or information exchange approaches designed for general mobiles networks put heavy efforts on maintaining a stable structure and suffer from the high disorder of the vehicular network.

There are also some research efforts devoted to general routing protocols design in VANET. Lochert et. al. [18] analyze a position-based routing approach for VANET in city environments. Harsch et. al. [11] integrate security mechanisms to protect the routing functionality and services for VANET. Taleb et. al. [23] propose a stable routing protocol in VANET which takes the vehicles' movement information into account

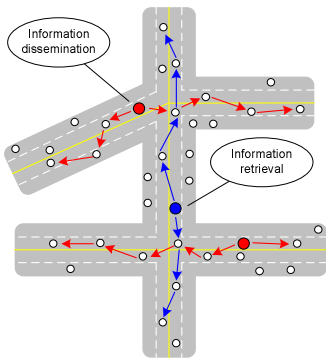


Figure 2: Information dissemination and retrieval paths.

(e.g., direction, speed, and the like) to predict a possible link-breakage event prior to its occurrence. Li et. al. present a survey in [16] about the research challenges of routing in VANET and recent routing protocols for VANET. Nevertheless, those studies for general routing problems do not focus on the query dissemination and information search, which we particularly consider in this paper.

The studies on Disruption Tolerant Network (DTN) opportunistically exchange information within the vehicular network when two vehicles meet [8, 15, 29]. Many efforts have been made on various aspects of DTNs. Epidemic routing protocols [25] explore all possible chances of message exchange between nodes when they meet each other such that a packet eventually reaches the destination. Improvements have been made to optimize the data replications and resource usage in the network [5, 6]. Some other approaches [3] treat routing as a resource allocation problem and intentionally optimize a specific routing metric with per-packet utilities that determine how packets should be replicated in the network. Nevertheless, in the urban VANET environment, users are quite sensitive to the efficiency of information search and the latency of data retrieval, which makes it difficult to directly borrow from the DTN studies.

Efficient information exchange has been studied in traditional multi-hop networks of stationary nodes. The approach of GHTs [20] hashes different types of data into multiple rendezvous nodes, enabling fast retrieval of user interested data. Double Rulings [22] deals with the problem of information brokerage in the sensor network context [17, 26]. Data storage and retrieval are well organized along specific curves in the network such that the delivery ratio is guaranteed. In vehicular networks, however, vehicle nodes are mobile. Those static structures built on stationary networks become extremely difficult to maintain in the highly dynamic VANET environment.

### III. DESIGN

Our design goal mainly focuses on efficient information dissemination and retrieval in urban VANETs. According to our two observations, the node mobility and signal transmission in the urban area are largely restricted by road segments. We build a network structure on top of the road map to support information exchange such that as a pre-knowledge for all vehicles the road map could be fully utilized to maximize the design efficiency. Each vehicle node obtains its location from the carried GPS device and maps it to the road map. At this

stage, we assume that the vehicles on the road are fully connected and information can be fluently delivered along the road. We will later reveal from the trace records how this assumption is away from practice and how we adapt our design to the practical conditions.

#### A. Design Framework

In this approach, we divide the road map into road segments. Each road segment is a section of road that spans one or several road intersections. We embed the VANET into the road map of the urban area and divide it into multiple components according to the road segments. A network component is a portion of VANET corresponding to a certain road segment and each component is assigned a unique identifier. Note that if a vehicle drives to a new road segment, it leaves its prior network component and join a new component. When a node has information to publish, it disseminates the information within its component along the road segment and each intermediate node stores a copy of that piece of information. Thus any node in the road segment stores the published information of all nodes within the network component of that road segment. When a node needs to query a certain type of information, it delivers its query along a set of road segments to search for desired information. The query carries the location of current road segment as the return address. When the query travels along the road segment, it checks all pieces of information stored in that component and when it comes across an intersection of two road segments it is able to access all pieces of information stored in the other network component. Figure 2 plots an example case including three network components along three road segments. The red arrows depict the paths of information dissemination from the source nodes (denoted as red) within their network components. The blue arrows depict the path of information retrieval from a query node (denoted as blue) where the query is delivered. At the intersections where two road segments meet with each other, the query hits the local storage of published information within the other road segment and returns the satisfied information pieces to the query node.

As Fig. 2 depicts, each road segment can be represented as a curve which contains all information pieces along the road segment. The most challenging issue is how we design the search path of the query to guarantee that a query hits all information pieces of arbitrary sources. In other words, we need to figure out a search path that intersects all curves of potential information, and we hope this path is optimized. In those approaches for stationary networks like Double Rulings [22], the dissemination paths as well as the retrieval paths are mathematically calculated and steadily embedded across the entire region of deployment. In this context, however, we need to adapt our structure for information organization to the road map regulations as our network structure is restricted by such regulations. We then optimize the information search over this structure of information organization.

We propose a novel solution to organize the information disseminated on network components of different road segments. We transform the original road map to a new graph  $G = \{V, E\}$  which we denote as *intersection graph*. We abstract

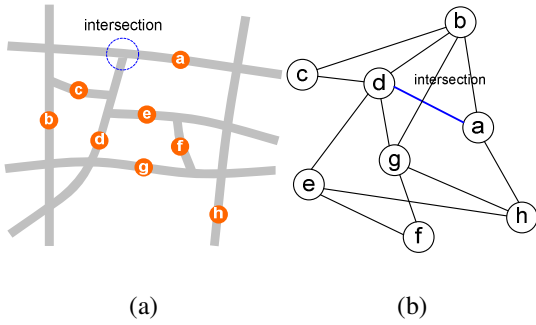


Figure 3 Map transformation: (a) the road map block, and (b) the corresponding *intersection graph*.

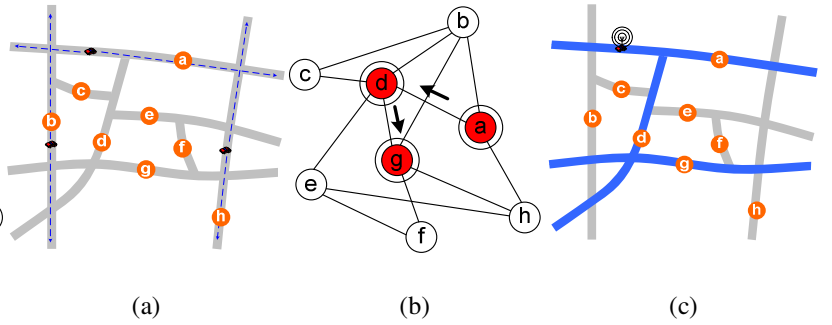


Figure 4 Design framework: (a) Information dissemination on local road segments, (b) CDS calculation on the *intersection graph*, and (c) the query delivery on the road map block.

each road segment as a vertex  $v_i$ ,  $v_i \in V$ . If two road segments intersect with each other, we add an edge  $e_{ij}$  in  $E$  connecting the corresponding vertices  $v_i$  and  $v_j$ . Figure 3 (a) depicts a sample block of road map in the urban area. It contains eight road segments labeled from  $a$  to  $h$ . Figure 3 (b) depicts the corresponding *intersection graph* transformed from Fig. 3 (a), where each road segment is mapped to a vertex as illustrated in Fig. 3 (b). As shown in the figures, the intersection points in the road map are represented by edges connecting the vertices of the two roads in the *intersection graph*. For example, roads  $a$  and  $d$  intersect with each other in Fig. 3 (a), so there is an edge connecting their corresponding vertices in Fig. 3 (b).

With the help of *intersection graph*, we can achieve information retrieval by building a connected dominating set (CDS) on the *intersection graph*. The CDS figures out a group of vertices and such that all vertices of the *intersection graph* are either in the CDS or directly connected to the vertices in the CDS. We then regard the CDS as a search area of queries. When we map this search area on the *intersection graph* back to the road map, accessing such a search area corresponds to delivering the query within those road segments labeled in the CDS. According to the definition of CDS and *intersection graph*, it is guaranteed that all road segments are either in the CDS or intersect with at least one road segment within the CDS. Thus with the search area we are able to retrieve all information pieces stored along the road segments. As the induced sub-graph of the CDS is a connected component, the query will be delivered across the corresponding road segments without connecting through any other road segments.

Figure 4 illustrates this process with an example. Figure 4 (a) shows that vehicles disseminate information along local road segments. Then when a vehicle node on road  $a$  is willing to query the network, it builds a CDS from node  $a$  in the *intersection graph*, i.e., set  $\{a, d, g\}$  as illustrated in Fig. 4 (b). By delivering its query along the three road segments, the node is able to access the stored information pieces on all roads through the road intersections as shown in Fig.4 (c).

### B. Efficient Query Zone

By building a connected dominating set, we guarantee that the query path hits all information pieces. To further make such a search path more efficient, we limit its size and explore a minimal dominating set. As different road segments corre-

spond to different sizes of network components, we weight their corresponding vertices in the *intersection graph* according to their sizes. We aim to find a Minimal Weight Connected Dominating Set (MWCDS) [9] on the weighted *intersection graph*. At current stage, we approximate the size of each road segment by its length  $L_i$  ( $i = 1, 2, \dots, n$ ) and thus the weight is calculated as  $W = L_i/L_{max}$ , where  $L_{max}$  denotes the maximum road length. In such a way, we normalize the road weight to range  $[0, 1]$ . The total path length for query delivering is bounded by the size of the constructed MWCDS. Later we will show by our trace based study that the weight could be more accurately assigned according to the vehicle density on the road segment.

The problem of constructing the CDS has been widely studied by literatures and it is proved to be NP-hard [9]. With the urban road map pre-loaded in each vehicle, it can be locally computed at each node. In this work, we apply the method proposed by Guha and khuller [9] for calculating our MWCDS. Their algorithm achieves an approximation factor of  $(C_n+1)\ln n$  to the optimal solution where  $C_n$  is a constant ( $C_n = 1.6103$ ). We use the graph shown in Fig. 5 as an example to describe the MWCDS construction process. In Fig.5 each circle denotes a vertex and each line denotes an edge (note that we omit some edges in Fig.5 for concise). The calculation for MWCDS contains three stages. In the first stage, a minimal weight dominating set is calculated by using an approximation algorithm for weighted set cover problem [12], i.e., if we regard each vertex in the graph as an element, and each vertex corresponds to a set including the vertex itself as well as its neighbors, thus all the vertices associated to a set cover forms a dominating set in the graph. A greedy algorithm iteratively picks sets based on the ratio of their weight to the number of new elements they cover until all elements are involved. After this stage, the selected vertices form a minimal weight dominating set. As illustrated in Fig. 5, all blue vertices are selected to the result set in this stage. However, this dominating set may not be a connected dominating set. For example, in Fig.5 the blue vertices are partitioned into several clusters. Then in the second stage, we connect all vertices in the dominating set by building a Steiner tree [10]. In this stage, some additional vertices are picked into the final result set to achieve connectivity. As shown in Fig. 5, by carefully selecting some red nodes into the result set, we connect all the separated parts (constructed by blue vertices) into one cluster. All vertices in the final result set

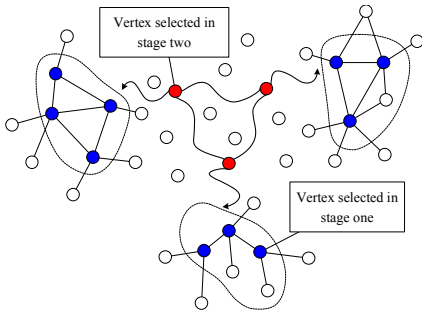


Figure 5: Constructing the MWCDS.

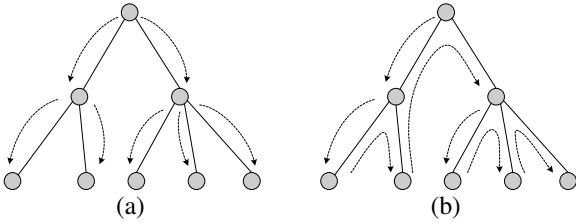


Figure 6 Two search schemes: (a) parallel search with multiple query copies in the zone, and (b) sequential search with a single query in the zone.

are included in the MWCDS. After stage two, the MWCDS is constructed. In this work, our final MWCDS should include the road segment that the query node is located in. In the third stage, we find the road segment that contains the query node and add its corresponding vertex in the *intersection graph* to the MWCDS. Thus the query zone for an arbitrary vehicle node consists of those road segments corresponding to the final MWCDS.

### C. Search Scheme

For delivering the query inside a query zone that corresponds to the MWCDS in the *intersection graph*, we build a spanning tree within the MWCDS rooted at the vertex of the road containing the query node. The path on the spanning tree is mapped to the delivery path on the road map. As previously defined, each vertex in the *intersection graph* denotes a road segment in the physical world, so accessing a vertex in the MWCDS means that a packet is forwarded along the corresponding road. Each edge in the *intersection graph* represents an intersection between the two road segments that its two end vertices represent, so there are definitely road crosses to forward the query from the road segment labeled by one vertex to the road segments represented by its child vertices. Thus the consecutive search on the MWCDS can always be performed on a connected path in the real VANET on the roads.

We can use two different ways to search on top of the spanning tree. If we let the query node generate multiple copies of a query and distribute them simultaneously in the network, the search is performed in a flooding manner on the spanning tree. As Fig. 6 (a) depicts, the query is delivered from the root to its successors. At each intermediate vertex on the spanning tree the query is copied and forwarded to all child vertices. Mapping this process to the real road map, the query node generates the query and delivers it along the road seg-

ment it resides in. Every time the query meets a road intersection between two roads labeled in the MWCDS, it is copied and forwarded to the new road segments. Since the spanning tree is a shortest path tree on the MWCDS, the search latency is minimized and bounded by the path from the root to the leaf that has the maximal summation of the weights on the passed vertices. We can save the burst traffic that the query introduces by maintaining only a single copy of the query in the network. We can figure out a sequential search path by sequentially traversing all vertices on the spanning tree. For example, as depicted in Fig. 6 (b), a DFS search on the spanning tree corresponds to forwarding the query along one road segment for at most twice. We mainly use the parallel search scheme to shorten query latency. However, as our later study shows, the latter scheme could be applied to prevent burst traffic burden when the network is busy.

### D. Properties of the Design

The advantages of our design lie in several aspects:

1. **Guaranteed delivery.** By leveraging the global map information, as long as the query message finishes traveling on all roads associated with the CDS, it is guaranteed to meet all information required in spite of the dynamics in the network.
2. **Low traffic generation.** Due to the sparse property of the dominating set, we only select a small percentage of roads over the entire network for query dissemination. Compared with the epidemic-like or random search schemes, the transmission cost of our approach is much lower.
3. **Bounded Search latency.** Different from random search schemes, the search latency of this scheme is bounded by the time to reach the farthest road which is close to that of flooding the network.
4. **Minimized control overhead.** We do not need to maintain any internal routing structures or other state information to guide the query. Meanwhile, the information dissemination is strictly limited on the local road segment. Thus the control overhead is minimized compared with traditional approaches.

## IV. DEVELOPING PROTOCOLS FROM THE TRACE

We focus on the design principles in the previous section. In this section, we present details of our information dissemination and retrieval protocol. Each node disseminates its information along the local road segment and all intermediate nodes on the same road segment store a copy of that piece of information. When a node needs to query a certain type of information, it delivers its query inside a query zone that corresponds to the MWCDS in the *intersection graph* to search for desired information as discussed in section 3. When applying this design to the real environment, we need to develop a complete protocol that adapt to various practical circumstances, such as the discrete network, mobility of vehicles, insufficient communication bandwidth, low network density, and etc. In this section, we intensively study the 4000 taxi trace data and accordingly develop out protocol to best suit the urban VANET characteristics.

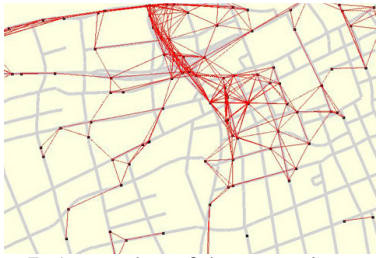


Figure 7: A snapshot of the network topology.

#### A. The Experimental Settings

The trace we investigate records the instant locations of more than 4000 taxis in the urban area of Shanghai City. It collects the GPS data of the taxis over several months. All taxis periodically report their real time locations through the GPS devices carried on the car. The vehicle velocities vary from 0m/s to 30m/s with an average of 9.75m/s. The report intervals vary from 20 seconds to a few minutes with an average value of 60 seconds. The location is represented by a pair of longitude and latitude coordinates. With the help of a digital map of Shanghai urban area, we reconstruct the mobile network of vehicle nodes in laboratory. The rough GPS data recorded contains measurement errors such that some taxi locations deviate from the driveways. We refine the coordinates of different taxis and associate them to the road segments according to their locations and headings. We can also interpolate to figure out their trajectories on the map from the samplings of their instant locations.

Our experiments are conducted in a selected 9km by 9km square region in southwest of Shanghai which spans both crowded areas of busy traffic and suburb areas of less density. According to the general specifications in current Vehicular Communication Systems, the DSRC-based devices on vehicles can achieve a maximum communication range of 1000m [1]. However, due to the collisions and environmental interferences in the urban area, the real DSRC radius is considered as 300-500m [13] for stable signal strength. Besides, different terrain features such as road segments, building blocks or trees, often bring different shadowing effects on the received signal. According to a recent study in urban wireless network assessment [21], we assign two different signal attenuation rates respectively for road segments and building blocks. In this setting, when the global maximum communication range is set to 500m, signals can be received by vehicles within 500m distance along the road. This distance, however, will be reduced to 300m if there are buildings between the sender and receiver. Under this setting, in different urban areas the average node degrees vary from 1 to 31. According to the regulations of most DSRC standards, there are two signal channels for each vehicle, a control channel for exchanging control messages as well as critical information, and a data channel for delivering data blocks. In our experiment, we assume that each vehicle node is able to discover and update its neighbor nodes through periodical beaconing within its neighborhood through the control channel (1 second interval in our experiment). Figure 7 shows a network snapshot when the maximum communication range is set to 500m. Each black spot on the road map denotes a vehicle and a red line connecting two spots indicates that the two vehicles can communicate with each other.

In the experiments, each taxi is an information provider that generates pieces of new information following a Poisson distribution with independent arrival rate. Each taxi disseminates information along the local road segment it resides in. Within each time interval, we randomly assign a set of vehicle nodes as information consumer that issue queries for specific types of information.

#### B. Organizing the Information

As previously described, when a vehicle generates a new piece of information, it disseminates the information along the road it resides in. While the road segment that contains the information is static, the information carriers namely the vehicles that ride along the road are mobile. We need to bind the information on the vehicle flow on the road instead of simply restoring it on some dedicated vehicles that might soon drive to other road segments. To achieve this goal, we let each vehicle node restore the information and exchange with other nodes at any possible chances when meeting with new neighbor nodes on the road. When one vehicle drives into a new road segment, it discards all information copies in its local storage and stores information copies of the new road segment. As such, information can be clung to the static road map although the actual carriers are mobile vehicles that run on top of the map.

We compare the overhead of information storage in each vehicle of our method and the flooding-based dissemination in which information is flooded and stored across the entire network. In this test, each taxi is assumed to generate a new piece of message every 120 seconds and the life time of each message is set to 300 seconds. Figure 8 shows the average number of messages stored in each vehicle during 120 minutes. Both methods achieve a stable state after a few minutes. By only replicating the information along local road segments our scheme results in much less storage cost. In the figure, we depict the 80% confidence intervals of the test results.

Our scheme has a higher variance of the message distribution. This is because the number of vehicles vary among different road segments so compared with flooding, our scheme stores different amount of messages on vehicles residing on different roads. In the second test, we count the average number of messages in each taxi while varying the size of the test region from 3 square km to 9 square km. From the results in Fig. 9, we find that the storage cost in the flooding approach increases nearly linearly with the network size, while our approach consumes only constant amount of storage space.

#### C. Query Forwarding

According to our design, by generating the *MWCDS* on the *intersection graph*, we obtain a group of road segments that are associated with the vertices in the *MWCDS*. We denote such a group the *CDS group* and the road segments in *CDS group* as *CDS roads*. The queries delivered along those road segments are guaranteed to hit all information if the underlying network is connected.

A baseline forwarding strategy delivers queries on all road segments in the *CDS group* and visits the information of the intersecting roads at the road crosses. Once a vehicle node receives a query, it firstly checks whether or not there are corresponding data within its local storage. It then forwards this

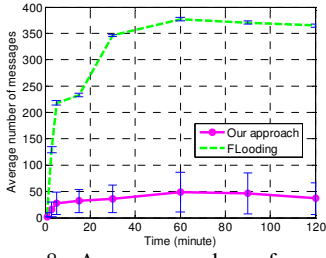


Figure 8: Average number of messages stored in each vehicle.

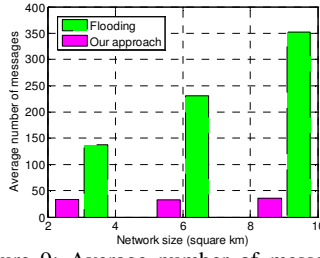


Figure 9: Average number of messages in vehicles under varied network size.

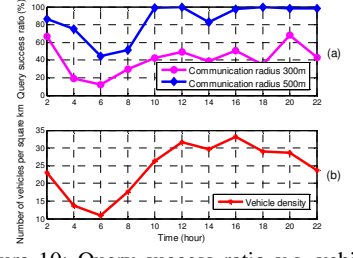


Figure 10: Query success ratio v.s. vehicle density.

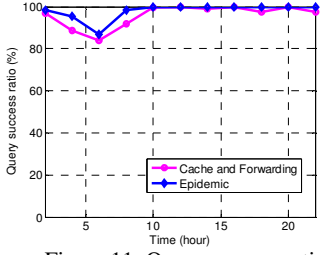


Figure 11: Query success ratios.

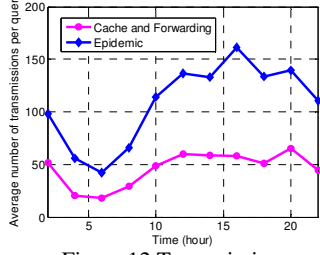


Figure 12: Transmission cost.

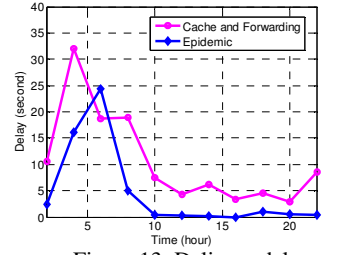


Figure 13: Delivery delay.

query to its neighbor nodes. If the vehicle receives the query from the same road, it forwards the query in the same direction as the query goes. If the vehicle receives the query from another road, it forwards the query to all its neighbors in both directions along the road. When the vehicle is to the road intersection, it also forwards the query to vehicles on the other road. Algorithm 1 illustrates the details of such a scheme.

---

#### Algorithm 1 Direct Forwarding

---

- 1: Receive query  $q$ ;
  - 2: Visit information on local road for  $q$ ;
  - 3: Visit information on neighbor vehicle's road;
  - 4: **If**  $q$  is received from another road **do**
  - 5: Forward  $q$  to two random neighbors on both directions;
  - 6: **end if**
  - 7: **If**  $q$  is received from the same road **do**
  - 8: Forward  $q$  to one random neighbor on the same direction as when receiving it;
  - 9: **end if**
  - 10: Forward  $q$  to all neighbors on other *CDS roads*
- 

We evaluate the performance of this scheme in our experiment field over a one day period from 2:00am to 10:00pm. We test two different settings of the network with 300 meter communication radius and 500 meter communication radius. We randomly issue 100 queries from different vehicles every hour and examine the success ratio of those queries. The query success ratio is plotted in Fig. 10 (a). Obviously, a larger communication radius results in better query success ratio. That is because the network has better connectivity in such a setting. Another interesting observation is that the performance of our scheme is relatively stable in a whole day except for a significant decrease in the period around 6:00am. In Fig. 10 (b), we further plot the variation of the density of vehicles in this area. We can find that the vehicle density has a sudden drop during

the period around 6:00am, as there are fewer taxis on the streets at the dawn time.

By comparing the two figures, we observe a perfect match between the vehicle density and the performance of our scheme. Indeed, the network connectivity severely affects the query delivery over the network. When the network becomes highly sparse, the VANET degrades into a disruptive network. In such a case the network might be separated into different parts where the queries cannot be guaranteed to deliver to all vehicles by instant forwarding. In order to deal with this problem, we propose a cache and forwarding scheme. The vehicles cache the received queries and rebroadcast the query as they have chances to meet new neighbors. Algorithm 2 illustrates the details of this scheme.

---

#### Algorithm 2 Cache and Forwarding

---

- 1: Receive query  $q$ ;
  - 2: **while** new neighbors are discovered **do**
  - 3: Visit information on local road for  $q$ ;
  - 4: **for all** neighbor vehicles  $v_i$  **do**
  - 5: **if**  $v_i$  has not received  $q$  **do**
  - 6: Visit information on  $v_i$ 's road;
  - 7: **if** road of  $v_i$  belongs to *CDS group* **do**
  - 8: Forward  $q$  to  $v_i$ ;
  - 9: **end if**
  - 10: **end if**
  - 11: **end for**
  - 12: **end while**
- 

By enabling the intermediate vehicle node to cache the queries, such a scheme explores much more opportunities to disseminate the queries while still limiting the transmission scale within the *CDS group*. We compare this scheme with the well-known Epidemic routing approach which is designed to deal with the intermittent connectivity in DTN [8]. In Epi-

demetic routing approach, each vehicle caches and disseminates the packets at any chances within the entire network. Through the results from Fig. 11 to Fig. 13, we conclude that even under low network connectivity, our scheme achieves high query success ratio similar to that of Epidemic. Our approach results in much less network traffic than the Epidemic method with a slightly higher delay. That might be caused by the fact that CDS-based forwarding strategy misses some chances to achieve the destination through some shortest paths on the map.

#### D. Making the Protocol Adaptive

Now we have a cache and forwarding scheme that achieves high delivery ratio with the sparse and even disruptive network. However, it will generate much extra overhead in a neatly connected network, leading to unnecessary consumption of network bandwidth resources. In such a case, the direct forwarding scheme appears to be a more feasible way with light weight. In practice, road diversities in the urban area such as one way streets, road networks with a combination of fast highways, or slow side streets near each other may have impact on the traffic densities and distribution. We hope to design an adaptive approach that alternates between the two strategies so as to achieve high query success ratio while efficiently accessing the network resources. When the network connectivity is low, our protocol applies the cache and forwarding scheme to improve the query success ratio, and when there is rich connectivity, our protocol chooses the direct forwarding scheme which is more economical in terms of traffic overhead and bandwidth consumptions.

Both the density and communication radius of vehicle nodes affect the network connectivity. A universal measure of the network connectivity is the average node degree [24]. We evaluate the query success ratios of both two forwarding strategies on our traces of varied node degrees (from 1 to 31). Figure 14 shows the results. While the Cache and Forwarding scheme has a persistent performance, the Direct Forwarding scheme experiences a nearly linear performance gain as the node degree of the network increases. It achieves a satisfactory query success ratio when the average node degree is above 18. This observation suggests us to set a cut-off threshold on the node degree for the vehicle to determine the forwarding scheme it uses. In practical usage, each vehicle measures the average node degree within its two hop neighborhood. It adaptively selects the forwarding scheme according to whether or not the estimated node degree is above 18.

A natural question is whether such an adaptive approach is stable. We want to prevent that the vehicles frequently switch their forwarding schemes due to the frequent dynamics of the network connectivity. Fortunately, according to the statistics shown in Fig. 15, it is not the case. Fig. 15 plots the variations of the vehicle densities over a 24 hour period in four different regions. Each region we examine is a randomly selected square region of 1km by 1km area. We find that although the node densities in different regions deviate much from each other they are relatively persistent along with the time. Even in the dawn time around 6:00am, the node densities change smoothly over hours. Such an observation enables us to apply our adaptive approach without much worry about vehicles frequently switching their schemes. Indeed, as the network connectivity is

bound to the concrete map regions, we can use a more convenient method. We can preload the statistical vehicle densities of different regions into the digital map and each vehicle is able to estimate the network connectivity locally according to its current time and location.

#### E. Optimizing the CDS group

The key operation of our design is that we select a small set of road segments into the *CDS group* and by delivering queries across the roads in the *CDS group* we have guaranteed access to all information disseminated over the field. We weight each road segment by its length. By searching within the MWCDS our approach delivers the queries along shorter paths so as to achieve shorter query latency with fewer transmissions.

Different road segments have varying vehicle densities due to many facts such as different geographical positions, different road layouts, and etc. Indeed, if we consider that different road segments might have different vehicle densities, the length of the road segment may not be the best approximation. To minimize the query latency, we shall choose to search along those road segments of dense vehicles that provide us good network connectivity. On the other hand, to reduce the waste of network bandwidth, we shall prevent to choose those road segments of excessively high vehicle densities that might lead to heavy transmission collisions. Towards this aim, we improve the way we weight each road segment and adaptively build the MWCDS that optimizes network utility. We use both the length  $L$  and the vehicle density  $D$  as the parameters to weight each road segment. While the previous parameter is an invariant in the field and can be directly obtained from the digital map, the latter one is relatively stable across the field as shown in the previous section and can be easily obtained from the historical statistics. We use a function  $f$  described by the two parameters to weight a road segment:  $W = f(L, D) = L \times e^{D-d}$ . In function  $f$ ,  $d$  is a constant threshold denoting the optimal node density for data transmission which is determined by the communication radius. When  $D$  is larger than  $d$ , the increase of  $D$  leads to a larger  $W$ , which reflects over-dense road segments, and the minimum weight selection procedure will select the less crowded road segments into the MWCDS. When  $D$  is smaller than  $d$ , it is opposite. An increasing  $D$  leads to the decrease of the weight, as in such a low density case, we prefer to choose the road segments with higher density to achieve better network connectivity. In function  $f$ , the term  $|D - d|$  is introduced to provide such an adaptive selection of road segments. The length  $L$  of the road segment is always preferred to be shorter for faster data delivery.

We conduct experiments to validate our adaptive design for optimizing the CDS groups. Figure 16 and 17 show the performance gain of introducing the new weight scheme. CDS1 denotes the weighting scheme with road length and CDS2 denotes the proposed adaptive weighting scheme with function  $f$ . Figure 16 compares the query delivery delay within the CDS groups selected by the two schemes. As indicated by the figure, the adaptive weighting scheme achieves lower delay across the entire time span. Figure 17 compares the transmission cost with the two different weighing schemes. Similarly the adaptive weighting scheme results in smaller overhead for all time instances.



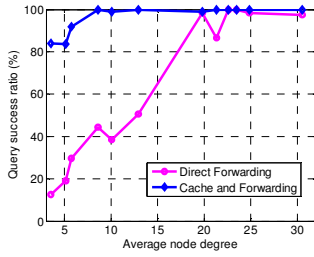


Figure 14: Query success ratios with varied node degree.

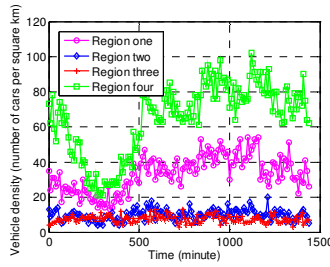


Figure 15: Vehicle densities in different urban areas.

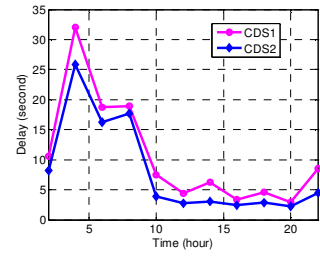


Figure 16: Delivery delay of different weighting schemes in CDS selection.

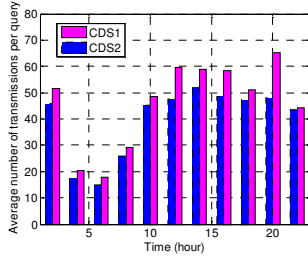


Figure 17: Transmission cost of different weighting schemes in CDS selection.

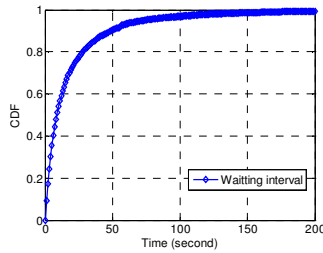


Figure 18: Waiting interval for coming vehicles at a road cross.

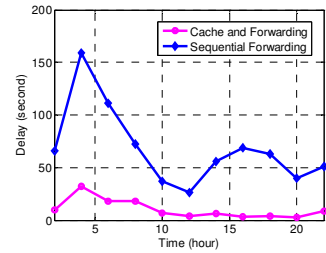


Figure 19: Delivery delay over time.

#### F. Trading Bandwidth Usage with Delay

Throughout our discussion, we assume a parallel query search over the *CDS group*. Such a scheme maximizes the query efficiency but inevitably introduces a burst traffic to the network when many queries are generated simultaneously, leading to heavy congestion within busy regions. In Section 3.3 we have discussed the scheme of sequentially forwarding one copy of the query on the MWCDS tree of the *CDS group*. Indeed, such a scheme can be utilized as a supplementary method for delivering those delay tolerant queries. In this scheme, each query specifies a path calculated from the MWCDS and its current position on the path. Each intermediate vehicle node only forwards the query towards the next road segment along the path.

As there is only one copy of the query maintained in the network, we need to carefully deal with the cases where the network is disruptive. For example, the query is supposed to be delivered from one road segment to the other at the road cross. However, due to the network dynamics or low connectivity, it is possible that there is no vehicle within the communication radius on the other road. There is a natural contradiction that the query message should have stuck at the cross waiting for the coming carriers while the current vehicle is still going ahead over the cross. In this approach, we let the current vehicle always relay the query to the neighboring vehicle that runs oppositely. Such a method will not work with some extraordinary cases, say, one way streets. To address such an issue, we let the current vehicle also pass its query to a vehicle behind it while leaving the cross. By such a means, we can stick the query message at the cross until it reaches a vehicle on the other road. Figure 18 plots a statistical measure that summarizes the waiting interval for coming vehicles at the road cross. The figure shows the cumulative distribution function of the waiting time and the result indicates that in the urban area we

can meet a coming vehicle from the other road within 40 seconds with high probability. If we take into account the node communication radius of 300m and average speed of 9.75m/s, almost all the time the query can be relayed with less than two times before it is delivered to the vehicles on the other road.

We can further utilize the mobility of vehicles to assist query delivery. When the network on the road segment is sparse, we always let the vehicle relay the query by chance to a neighbor that runs faster and overtakes it such that the query would be delivered faster along its path. Also, when the vehicle runs towards the direction other than the query, it will always find the chance to relay the query to another vehicle that runs oppositely. Indeed, it is easy for two vehicles to get the headings of each other through the DSRC control channel [1].

We compare the sequential query delivery scheme with the parallel scheme discussed in prior sections. In this test, both methods work with 300m communication range for 20 hours. The average delay of query delivery is shown in Fig. 19. Compared with the parallel query delivery the sequential scheme has a nearly four times higher delay at all times. As a reward, as Fig. 20 depicts, the sequential delivery scheme consumes much less network bandwidth resources, especially during the peak hours where there are excessively many vehicles on the roads. As validated by the experiment results, the sequential scheme helps to prevent burst bandwidth consumption with a cost of increased delivery delay. It is suitable for those delay tolerant applications, where instant information collection is not a necessity.

#### G. Robustness of the Protocol

In previous sections, we assume perfect road map updating at all vehicles and the road map is regarded as a pre-knowledge in each vehicle. In practice, inconsistent road maps across different vehicles may affect the efficiency of the query process and lead to query failures to access information from

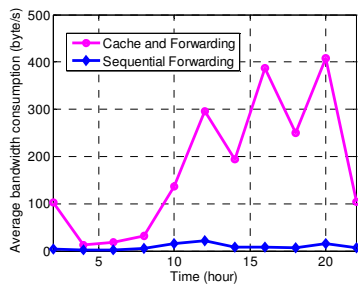


Figure 20: Bandwidth usage per query.

some road segments. Therefore, we need additional communication cost to keep the updated road map in each vehicle. Since the road networks are relatively stable, the road map updating process is not frequent and thus the additional communication overhead brought by road map updating is negligible. In the future work, we will try to find efficient approaches for road map updating and thus to reduce the communication cost.

Our protocol may also face security issues such as fake information, malicious packet dropping, and etc., Nevertheless, existing security mechanisms including information encryption, message authentication, and other strategies can be directly applied on top of our solution to enhance the robustness of the protocol against malicious behaviors.

## V. CONCLUSION

Vehicular network is one instance of mobile ad hoc network but has its own characteristics. In this work, we explore the hidden connectivity of the underlying road segments that contain the vehicle flows and design efficient information exchange approach by using the *intersection graph* of road segments. We conduct a trace based study with our 4000 taxi trace in Shanghai City and according to the vast experimental findings we develop tailored techniques that adapt our design to the practical urban VANET environment.

## ACKNOWLEDGMENT

This work is supported in part by NSFC/RGC Joint Research Scheme N\_HKUST602/08, National Basic Research Program of China (973 Program) under Grants No. 2011CB302705, and NSF China grants No.60833009 and No.60925010.

## REFERENCES

- [1] "Dedicated Short Range Communications (DSRC) Home " <http://www.leearmstrong.com/DSRC/DSRCHomeset.htm>
- [2] U. G. Acer, S. Kalyanaraman, and A. A. Abouzeid, "Weak State Routing for Large Scale Dynamic Networks," In Proc. of ACM/SIGMOBILE MOBICOM, 2007.
- [3] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN Routing as a Resource Allocation Problem," In Proc. of ACM SIGCOMM, 2007.
- [4] N. Banerjee, M. D. Corner, D. Towsley, and B. N. Levine, "Relays, Base Stations, and Meshes: Enhancing Mobile Networks with Infrastructure," In Proc. of ACM/SIGMOBILE MOBICOM, 2008.
- [5] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, "MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks," In Proc. of IEEE INFOCOM, 2006.
- [6] B. Burns, O. Brock, and B. N. Levine, "MV Routing and Capacity Building in Disruption Tolerant Networks," In Proc. of IEEE INFOCOM, 2005.

- [7] J. Eriksson, H. Balakrishnan, and S. Madden, "Cabernet: Vehicular Content Delivery Using WiFi," In Proc. of ACM/SIGMOBILE MOBICOM, 2008.
- [8] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," In Proc. of ACM SIGCOMM, 2003.
- [9] S. Guha and S. Khuller, "Approximation Algorithms for Connected Dominating Sets," *Algorithmica*, vol. 20, pp. 374-387, 1998.
- [10] S. Guha and S. Khuller, "Improved Methods for Approximating Node Weighted Steiner Trees and Connected Dominating Sets," in *Information and Computation*, vol. 150: Springer Berlin / Heidelberg, 1999, pp. 57-74.
- [11] C. Harsch, A. Festag, and P. Papadimitratos, "Secure Position-Based Routing for VANETs," In Proc. of IEEE VTC, 2007.
- [12] D. S. Hochbaum, *Approximation Algorithms for NP-Hard Problems*: PWS Publishing Company, 1996.
- [13] D. Jiang, Q. Chen, and L. Delgrossi, "Optimal Data Rate Selection for Vehicle Safety Communications," In Proc. of The Fifth ACM International Workshop on Vehicular Ad Hoc Networks, 2008.
- [14] D. B. Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," in *Mobile Computing*, vol. 353: Kluwer Academic Publishers, 1996, pp. 153-181.
- [15] P. Juang, H. Oki, Y. Wang, M. Martonosi, L.-S. Peh, and D. Rubenstein, "Energy-Efficient Computing for Wildlife Tracking: Design Tradeoffs and Early Experiences with ZebraNet," In Proc. of ACM ASPLOS, 2002.
- [16] F. Li and Y. Wang, "Routing in Vehicular Ad Hoc Networks: A Survey," *IEEE Vehicular Technology Magazine*, vol. 2, pp. 12 - 22, 2007.
- [17] M. Li and Y. Liu, "Rendered Path: Range-Free Localization in Anisotropic Sensor Networks with Holes," *IEEE/ACM Transactions on Networking (TON)*, vol. 18, pp. 320-332, 2010.
- [18] C. Lochert, H. Hartenstein, J. Tian, H. Fussler, D. Hermann, and M. Mauve, "A Routing Strategy for Vehicular Ad Hoc Networks in City Environments," In Proc. of IEEE Intelligent Vehicles Symposium, 2003.
- [19] C. E. Perkins and E. M. Royer, "Ad-Hoc On-Demand Distance Vector Routing," In Proc. of IEEE Workshop on Mobile Computing Systems and Applications, 1999.
- [20] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker, "GHT: A Geographic Hash Table for Data-Centric Storage," In Proc. of 1st ACM International Workshop on Wireless Sensor Networks and Applications 2002.
- [21] J. Robinson, R. Swaminathan, and E. W. Knightly, "Assessment of Urban-Scale Wireless Networks with a Small Number of Measurements," In Proc. of ACM/SIGMOBILE MOBICOM, 2008.
- [22] R. Sarkar, X. Zhu, and J. Gao, "Double Rulings for Information Brokerage in Sensor Networks," In Proc. of ACM/SIGMOBILE MOBICOM, 2006.
- [23] T. Taleb, E. Sakhaee, A. Jamalipour, K. Hashimoto, N. Kato, and Y. Nemoto, "A Stable Routing Protocol to Support ITS Services in VANET Networks," *IEEE Transactions on Vehicular Technology*, vol. 56, pp. 3337 - 3347, 2007.
- [24] S. Ukkusuri, L. Du, and S. Kalyanaraman, "Geometric Connectivity of Vehicular Ad Hoc Networks: Analytical Characterization," In Proc. of The Fifth ACM International Workshop on Vehicular Ad Hoc Networks, 2007.
- [25] A. Vahdat and D. Becker, "Epidemic Routing for Partially-Connected Ad Hoc Networks," Duke University, Technical Report 2000.
- [26] Z. Yang and Y. Liu, "Quality of Trilateration: Confidence based Iterative Localization," *IEEE Transactions on Parallel and Distributed Systems (TPDS)*, vol. 21, pp. 631-640, 2010.
- [27] B. Yu, J. Gong, and C.-Z. Xu, "Catch-Up: A Data Aggregation Scheme for VANETs," In Proc. of The Fifth ACM International Workshop on Vehicular Ad Hoc Networks, 2008.
- [28] X. Zhang, J. Kurose, B. N. Levine, D. Towsley, and H. Zhang, "Study of a Bus-based Disruption-Tolerant Network: Mobility Modeling and Impact on Routing," In Proc. of ACM/SIGMOBILE MOBICOM, 2007.
- [29] W. Zhao, M. Ammar, and E. Zegura, "A Message Ferrying Approach for Data Delivery in Sparse Mobile Ad Hoc Networks," In Proc. of ACM MobiHoc, 2004.
- [30] S. Zhong and F. Wu, "On Designing Collusion-resistant Routing Schemes for Non-cooperative Wireless Ad Hoc Networks," In Proc. of ACM/SIGMOBILE MOBICOM, 2007.