On Optimal Service Directory Selection in Urban Vehicular Networks

Fenggang Wu Hongzi Zhu Jia-Liang Lu Min-You Wu
fenggang.wu@sjtu.edu.cn, {hongzi,lu-jl,wu-my}@cs.sjtu.edu.cn
Department of Computer Science and Engineering
Shanghai Jiao Tong University, Shanghai, China

ABSTRACT

This paper studies the Delay Bounded Service Discovery Protocol (DB-SDP) problem in Urban Vehicular Ad-hoc Networks (VANETs). Services and resources on the vehicles must be discovered before they become available to the whole network. In distributed Service Discovery Protocols (SDPs), Service Directories (SDs) are selected to store the service description for other vehicles. Selecting SDs is difficult because VANETs has the disruptive nature which incurs a significant delay, whereas the users may have QoS requirements that the query for some certain service must be answered within some delay bound. We study the problem of optimal SD selection that minimizes the number of the SDs under such delay bound requirement. We theoretically prove that such SD selection problem is NP-complete even when the future positions of the vehicles are known as a priori. To solve the DB-SDP without the prior knowledge of future traces, we theoretically and empirically analyze the number of vehicles covered by the set of SDs within a delay bound. We find the number of covered vehicles exhibits some strong regularity. Regarding this regularity, we develop a heuristic iterative algorithm for the optimal SD selection. We conduct extensive trace driven simulations based on real vehicular GPS data and the results show that with high probability, our algorithm can select SD sets 50% smaller than those selected by alternative algorithms.

Categories and Subject Descriptors

C.2.1 [**Computer-Communication Networks**]: Network Architecture and Design—*Wireless Communication*

General Terms

Algorithm, Design, Performance

Keywords

Delay Bounded, Service Directory, Service Discovery, Vehicular Ad-hoc Networks (VANETs)

UrbaNE'12, December 10, 2012, Nice, France.

1. INTRODUCTION

Recently, as an important component of the future digital/smart city, urban Vehicular Ad Hoc Networks (VANETs) is gaining more attention in the academical field. In VANETs, vehicles are equipped with wireless communication modules and can exchange data when they are within the communication range of each other. Such network of interconnected vehicles can provide various kinds of services to the citizens.

In VANETs, vehicles can share the resources and services it possesses (sensory data, p3 files, camera, printer, etc.) with the rest of the network. We refer to such vehicles the *Service Providers, (SPs)*. These services and resources must be discovered by the *Service Requesters, (SRs)* before they become available to others and be fully utilized.

Designing an efficient and scalable Service Discovery Protocol (SDP) in VANETs is a non-easy job because of the highly dynamic mobility of the vehicles and the fast-changing topology of the network. Also due to the infrastructure-less nature of VANETs, data delivery often suffers from significant delay, which is in contrast with the users' QoS requirement that a query for a service may have to be answered in a bounded or controllable delay.

There are many existing works about service discovery in Mobile Ad Hoc Networks (MANETs). Many works adopting the *Directory-less Architecture* such as [10][4][6] and[3] are known to have significant discover overhead caused by flooding the service queries. Other approaches in [1] and [5] follow the *Centralized Directory Architecture*, and they will suffer from single point failure and scalability problems.

Some works resort to a *Distributed Directory Architecture* for better scalability, as well as efficiency[9][12][2]. In such architecture, a subset of the nodes, namely, the *Service Directories* (*SD*s) are selected to form a dominant set of the network. This architecture (Fig. 1) is well suited to the mobile ad hoc network scenario due to its better scalability, shorter response time and less communication overhead. However, existing works addressing SDP in MANETs do not take the significant delay in VANETs into consideration, so they will not work well in the VANETs environment. In VANETs, such SDs must be selected carefully in that i) the SD set size cannot be unlimited because of the cost of deployment, and that ii) the queries from the network must be addressed properly. Optimally selecting the SDs from all the nodes is challenging because:

- The infrastructure-less and disruptive nature of the VANETs will incur significant delay whereas users may have QoS requirement for a bounded and controllable delay.

- The vehicles exhibit highly dynamic mobility and the network topology is volatile and fast-changing, which will make existing SDPs to fail.

In this paper, we consider the problem of selecting the optimal set of SDs so as to minimize the number of the SDs under the constraint of a bounded response delay, which is a fundamental issue for the SDP design in VANETs. We prove that the problem is NP-Complete even when the future vehicle positions are given as a priori. We propose a greedy algorithm to find a nearly optimal solution when the future locations are given.

And in more realistic settings, however, the future positions of the vehicles cannot be known in advance. To deal with this problem, we perform extensive data mining on a data set of Shanghai taxi GPS of over 4000 vehicles during a time span of more than one year. We find the number of vehicles covered by the set of SDs during a given delay bound exhibits some strong characteristics. Regarding those characteristics, we develop a heuristic algorithm for the optimal SD selection. Our contribution of the paper can be highlighted as follows:

- We first formulate the Delay Bounded SDP (DB-SDP) problem in VANETs and prove that it is NP-Complete even with the future traces given as priori.

- We reveal the stochastic characteristics of the number of vehicles covered by a set of *SDs* in a given delay bound D. And based on this finding we develop heuristic iterative algorithms to deal with the *DB-SDP* problem.

- We conduct comprehensive trace driven simulation based on real GPS traces of Shanghai taxis to illustrate the effectiveness and efficiency of our approach.

The paper is organized as follows. The next Section discusses Related Work. In Section 3 we present the system model and formally define the problem. The analysis of the delay bounded cover set is presented in Section 4. Algorithm design is presented in Section 5. In Section 6 we present the evaluation methodology as well as the comparison results. We conclude the paper in Section 7.

2. RELATED WORK

According to whether a *directory* exists or not, the service discovery architecture can be divided into two categories: *directory-based architecture* and *directory-less architecture*. And the *directory-based architecture* approaches can be further divided into two categories, namely, *centralized directory-based architecture* and *decentralized directory-based architecture*.

In the directory-less architecture, the SPs do not spread the service description to other nodes. The service discovery is performed by the SPs broadcasting their service descriptions and/or the SRs broadcasting their service requests [4][10][7][3]. In [10][7] service servers periodically push service advertisements to their 1-hop neighbor by broadcasting. And [3] follows a similar approach while taking into account the power constraint and the computing capacity limit.

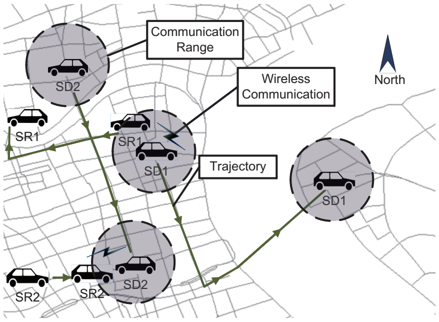

Figure 1: An illustration of the distributed directory-based architecture in VANETs. When SRs meets SDs, they will send the query for a certain service. Then the SDs can reply the query by the service owner's id. If there are no SDs in the vicinity, SR will wait for some time delay until it encounters the first SD and gets its request answered.

In the centralized directory architecture, one or a few service directory nodes are selected to store the service descriptions of all the available services in the whole network. In [1], service providers register their services onto the lookup servers and clients will receive the service interface after requesting the servers. SLP [5] is a service discovery protocol and has enterprise scalability by using service scope, which is a set of services that can be assigned to certain domain.

In the decentralized directory architecture, the directories are distributed to cope with the availability and scalability problem. [8] proposes a distributed service discovery architecture which relies on a virtual backbone for locating and registering available services. In [12], directories are deployed so as to ensure at least one directory is reachable within a fixed number of hops. DSDM [2] selects the nodes with the largest residing time, battery life, available bandwidth and available memory for caching, and uses the Minimum Distance Packet Forwarding (MDPF) algorithm to find the nearest service directory in terms of hop count. [11] creates quorums of directory nodes that exhibit higher resource and lower mobility. Also, it divides the network into topological domains to reduce service discovery cost.

In summary, some of the existing protocols are not initially designed for the Ad - hoc environment. Others proposed protocols in Ad-hoc networks either do not consider the delay of the service discovery request or assume the network is always connected without partition and estimate the delay by the hop counts. Our work differs from the existing work in that we consider the SD selection problem in disruptive environments such as VANETs.

3. SYSTEM MODEL

In this section, firstly we will describe the architecture of decentralized directory based service discovery. Then we will give the formal definition of the Delay Bounded Service Discovery Protocol (DB-SDP) Problem.

3.1 System Overview

The scenario is under VANETs, where vehicles that are equipped with wireless communication modules can communicate with each other within a Communication Range R. When two vehicles are within R, one communication link will form. The links are assumed to have the same capacity that is sufficient to support the exchange of the service querying and service description data. And we refer to such encounter as a *contact event*. All the n vehicles in the VANETs are denoted by the set $\mathcal{U}$. In this paper, we will use vehicles and nodes interchangeably.

Every vehicle v_i is assumed to have some kinds of services or resources Γ_i. And each vehicle has a service description file F_i that describes the detail of Γ_i. On the other hand, each of the vehicles may want to discover some kinds of service it does not have. We follow the decentralized directory based architecture as is described in [9], [12], and [2]. In such architecture, there are 3 roles of nodes, i.e. the SPs, SRs, and the SDs, as is described in Section 1. The set of the SDs is denoted by S.

Once an SP has a new service to share, it has to register it to the nearest SD. Then this SD will be in charge of spreading the description of the service to all the other SDs. When an SR wants to access to a certain type of service, it will query the nearest SD it encounters. We assume the users have QoS requirement that a query not answered within a time bound D is considered to fail. At the same time, the number of vehicles in the set S cannot be unlimited in that such directory vehicles will have a cost for deployment. This assumption is realistic because such vehicles may have to be equipped with better communication modules, larger memories and storage disks or more powerful processors. Therefore the SDs must be selected carefully to use a small number of directories to cover more nodes within the time bound D. Here for simplicity, we only consider the direct contact between the SRs and the SDs.

We assume that the services maintained by the SPs can keep unchanged in a relatively long time than the inter-query time, and can be seen as stable. Thus the updating cost can be neglect compared to the querying cost.

3.2 Problem Formulation

In this section, we give the problem definition the *Delay Bounded Service Discovery Protocol* Problem *(DB-SDP)*.

Definition 1. Delay Bounded Cover by Node Node v is said to be in the delay bounded cover set of node u with a delay D or covered by node u with a delay D iff. v can meet u within delay D. It is denoted as

$$v \in \mathcal{C}(u, D), \tag{1}$$

Definition 2. Delay Bounded Cover by Node Set Node v is said to be in the delay bounded cover set of set S with a delay D or covered by set (S) with a delay D iff. $\exists$u$\in S$ s.t. $v \in \mathcal{C}(u, D)$. It is denoted as

$$v \in \mathcal{C}(S, D). \tag{2}$$

Definition 3. Delay Bounded Coverage Ratio Delay bounded coverage ratio is the fraction of the nodes that satisfies $v \in \mathcal{C}(S, D)$ among all of the nodes. It is denoted as

$$\eta(S, D) \triangleq \frac{\parallel \mathcal{C}(S, D) \parallel}{\parallel \mathcal{U} \parallel}, \tag{3}$$

where $\parallel . \parallel$ means the size of a set. Here we arrive at the formal definition of the *DB-SDP* problem.

Definition 4. Delay Bounded Service Discovery Protocol Problem (DB-SDP)

$$\begin{aligned} \min_{S} \quad & \parallel S \parallel \\ \text{s.t.} \quad & \eta(S, D) > \gamma \quad , \end{aligned} \tag{4}$$

where γ is a required *bounded coverage ratio*.

Further, according to whether the future trace is given, we can have two versions of *DB-SDP*, i.e. the *Determined DB-SDP* and *Non-determined DB-SDP*

Definition 5. Determined DB-SDP: The Determined DB-SDP problem is the problem with the future contact information of the vehicles given as priori.

THEOREM 1. *Determined* DB-SDP *problem is NP-Complete.*

PROOF. Here we first prove that the Determined DB-SDP is in the NP class. We can choose one subset S from $\mathcal{U}$ that has k elements. Then we check whether all the other vehicle will meet them by scanning the contact events. This testing phase can be executed in Polynomial time. Therefore, Determined DB-SDP is in the NP class. We prove the Determined DB-SDP problem is NP-hard by reducing the *Dominating Set Problem* to it. In graph theory, the dominating set for a graph $G = (V, E)$ is a subset T of V such that every vertex in V that is not in T will share some edge with at least one vertex in T. And the dominating set problem is defined as whether $\parallel T \parallel \leq K$ for a given Graph and input integer K.

Here we reduce *Dominating Set Problem* to Determined DB-SDP problem. Given the input of the dominating set problem G and K, we can construct the input of Determined DB-SDP as follows: Each vehicle v_i corresponds to one vertex V_i of G. For each edge e_{ij} in G, we create a contact event between v_i and v_j in the future at a random time $t_{ij} < D$. Finally we set $\gamma = 100\%$. We can see the reduction can be performed in Polynomial time, and the answer to the newly constructed Determined DB-SDP is also the answer of the dominating set problem. Thus the Determined DB-SDP is NP-Hard. Since Determined DB-SDP is in the NP class and is NP-hard, DB-SDP is NP-Complete. ☐

In realistic settings, future positions of the vehicles are very hard or impossible to determine. And given a set of directories S, the number of covered vehicles $\parallel \mathcal{C}(S, D) \parallel$ is no longer a determined value. We denoted the size of the covered vehicles as a random variable $\xi(S, D)$. Also the coverage ratio η becomes a random variable subsequently. We cannot guarantee η to be always above the required coverage ratio γ in all cases. Thus we change the constraint into a stochastic one as follows.

Definition 6. Delay Bounded Guarantee Ratio The probability that the Delay Bounded Coverage requirement $\eta(S, D) > \gamma$ is satisfied.

$$\varrho(S, D, \gamma) \triangleq \Pr\{\eta(S, D) > \gamma\}. \tag{5}$$

Definition 7. Non-determined Delay Bounded Service Discovery Protocol Problem

$$\begin{aligned} \min_{S} \quad & \parallel S \parallel \\ \text{s.t.} \quad & \varrho(S, D, \gamma) > \varpi \end{aligned} \tag{6}$$

where ϖ is a required *guarantee ratio*.

4. DELAY BOUNDED COVER SET ANALYSIS

In this section, in order to design the algorithm that satisfies the constraint eq. 6, we study the statistical characteristics of the variable $\xi(S, D)$. First we prove that $\xi(S, D)$ follows a normal distribution, then we verify this by empirical study. This empirical study result also shows the parameters of the normal distribution is nearly stable over time.

4.1 Analytical Study for $\xi(S)$

In this section, we prove that $\xi(S)$ follows a normal distribution under some assumptions. It has been recognized that the inter-contact time between two vehicles exhibits exponential distribution. Let τ_{ij} be the inter-contact time between v_i and v_j. We assume $\tau_{ij} \sim$Exponential(λ_{ij}). And we assume each taxi moves independently. Therefore the contact events are independent of each other. We define an indicator $I_{ij}(D)$ to indicate whether two vehicles v_i and v_j will meet in the future D time.

$$I_{ij}(D) \triangleq \begin{cases} 1 & , \text{ if } v_i \text{ and } v_j \text{ meet in the future } D \text{ time} \\ 0 & , \text{ otherwise} \end{cases} \tag{7}$$

Theorem 2. *Random variable $\xi(S, D)$ follows a normal distribution when the number of vehicles goes to infinity.*

Proof. Since the inter-contact time follows exponential distribution, given the delay bound D, we can calculate the probability that two vehicles v_i and v_j have contact in the future D time by $p_{ij} = 1 - e^{-\lambda_{ij} D}$.

Here for consistency, p_{ii} is defined to be 1. We use a matrix M to include all the probability entries.

$$M(S, D) \triangleq (p_{ij}(S, D))_{n\times n}. \tag{8}$$

Because the vehicle moves independently, the contact events between each pair of the vehicles are independent. For any vehicle v_i, its probability q_i of being covered by any of the directories in S can be computed as

$$q_i = 1 - \prod_{v_j \in S} (1 - p_{ij}). \tag{9}$$

Now we know that $I_i \sim$ Bernoulli (q_i), with each I_i independent. $\xi(S, D)$ can be viewed as the sum of all the indicators I_i: $\xi(S, D) = \sum I_i$. Here let $Y_i = I_i - q_i$. For any $\delta > 0$,

$$1 > q_i(1 - q_i) = \text{ E } Y_i^2 > \text{ E } |Y_i|^{2+\delta}. \tag{10}$$

Let $T_n = \sum_{i=1}^n Y_i$ and $s_n = $ Var $T_n = \sum_{i=1}^n \sigma_i^2$. We have

$$\frac{1}{s_n^{2+\delta}} \sum_{i=i}^{n} \text{ E } |Y_i|^{2+\delta} \leq \frac{1}{s_n^{2+\delta}} \sum_{i=1}^{n} \text{ Var } Y_i = \frac{1}{s_n^{\delta}}. \tag{11}$$

Therefore, if $n \to \infty$, $s_n \to \infty$ (which is true given q_i is not all 0 or 1), the Lyapunov condition is satisfied. According to the Lyapunov Central Limit Theory, and we have $\sum_{i=1}^n \frac{Y_i}{s_n} \sim \mathcal{N}(0, 1)$, i.e.

$$\xi(S, D) \sim \mathcal{N}\left(\sum_{i=1}^{n} q_i, \sum_{i=1}^{n} q_i(1 - q_i)\right). \tag{12}$$

□

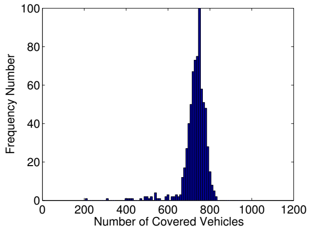

Figure 2: Histogram of $\xi(S, D)$. $|S| = 200$, $D = 30$min.

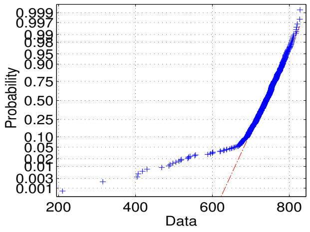

Figure 3: Normal probability plot of $\xi(S, D)$. $|S| = 200$, $D = 30$min.

4.2 Empirical Study

By extensive statistical analysis on a large data set of the real GPS traces of Shanghai taxis, we find that $\xi(S, D)$ follows a normal distribution in most cases, which verifies the conclusion of Section 4.1.

We collect $\xi(S, D)$ at a wide range of different S and different D. Then we draw the histogram and normal probability plot for the samples. The result shows that $\xi(S, D)$ exhibits significant characteristics of normal distributions. One example is given by Fig. 2 and Fig. 3. We also find that this normal distribution of $\xi(S, D)$ is stable. We choose two consecutive time periods (two weeks each) and make a hypothesis that its distribution is stable and then we conduct the second hypothesis test. Results show that the parameter is stable in a high confidence interval.

Since $\xi(S, D)$ is normally distributed. Given the parameter of the distribution, i.e. μ and σ^2, we can determine whether the requirement defined by eq. 6 is satisfied. Meanwhile, because the distribution is stable, we can estimate the distribution parameter of the $\xi(S, D)$ of the future D time using the distribution of the past D time. We will follow this idea to design the iterative algorithm in the next Section.

5. DELAY BOUNDED SDP ALGORITHMS

In this section, we first give a greedy algorithm for Determined DB-SDP. Then we develop it into a new heuristic iterative algorithm that deals with the Non-determined DB-SDP.

5.1 Algorithm for Determined DB-SDP

As in an urban VANETs of a metropolis such as Shanghai, the node number is huge. It is computationally infeasible to find the optimal solution. Since the objective of the algorithms is to achieve a required coverage ratio γ using as less SDs as possible, we can follow the heuristic to add the vehicle that has the most uncovered neighbors to S. This leads to a simple greedy algorithm.

The algorithm iteratively adds the node with the most neighbors into the SD set and excludes the nodes that have been covered, until the total number of the covered node exceeds the γ requirement.

The pseudo code of this algorithm is omitted here and can be found in our technical report [13].

5.2 Algorithm for Non-determined DB-SDP

We will follow a similar approach to propose a greedy algorithm to solve the Non-determined DB-SDP problem. We

Algorithm 1 Greedy Algorithm for Non-determined DB-SDP

```
Input: $M$, $D$, $\gamma$, $\varpi$
Output: $S$
$S, G \leftarrow \emptyset$ { $G$ is the graph in which the edge is the probability of meeting in the next $D$ time}
compute $G$ from $M$, $D$
set the covered probability of all vertices in $G$ to be 0
while True do
    for all nodes that are not covered do
        compute the gain of $\mu$ for adding the node
    end for
    $v_m \leftarrow$ candidates with the most gain of $\mu$
    update all the neighbor's covered probability.
    $S \leftarrow S \bigcup v_m$
    if $\mu > \gamma$ then
        if Set $S$ satisfied the test based on a history of $T_H$ then
            return $S$
        end if
    end if
end while
```

should answer two key questions before doing this algorithm migration.

- The rule of selecting best candidate SD.
- The stopping criterion for the iteration.

Our algorithm will select the next candidate with the max expected coverage ratio gain.

As is shown in eq. 12, the expected size of delay bounded cover set μ, can be estimated by $\mu = \frac{1}{n}\sum_{i=1}^{n} q_i$, which means summing up all the expected covered ratio of every node. When a new node is added to S, all q_i will increase correspondingly (eq. 9). The gain g_i for node v_i when add a new node v_j can be calculated by $g_i = q_i(k+1) - q_i(k) = p_{ij} - p_{ij}q_i(k)$, where $q_i(k)$ means the probability of being covered by S in the k-th round. Therefore, the gain of $\mu = \sum g_i$, can be used as a measure to choose the *best* candidate that has the biggest expected incremental value of μ.

We will use the historical data to determine whether to stop the iteration.

To guarantee the requirement given by eq. 6 and to mitigate the possible inaccuracy of measurement of the inter-contact time between vehicles, we design the stop condition called *split-time historical check* as follows. We choose a history with a time span of T_H. The algorithm splits the historical data by time intervals of time D. So there are $T_a = \frac{T_H}{D}$ intervals in total. The algorithm counts the times when the coverage ratio requirement γ is satisfied, say, T_s times. Then ϖ can be estimated by $\varpi^* = \frac{T_s}{T_a}$. The algorithm will stop when $\varpi^* > \varpi$. For acceleration, the algorithm does not perform *split-time historical check* until $\mu > \gamma$. The pseudo code of this iterative Algorithm is shown in Algorithm 1.

6. PERFORMANCE EVALUATION

In this section, firstly we present the methodology as well as the experimental setup of the performance evaluation of the proposed algorithm. Then we present the results.

6.1 Methodology and Experimental Setup

We conduct a real trace based simulation to show the efficiency of our proposed algorithm. The traces are collected from more than 4,000 operational taxis over a two year period from Jan. 2006 to Dec. 2007. We randomly select a subset of about 1,000 of the traces for the experiment.

The metric of this performance evaluation is the size of S. We take the variables of γ, ϖ and the delay bound D as the influencing factors. The default value of the controlling variables is as follows: communication range $R = 500$m, delay bound $D = 30$ min, $\gamma = 0.95$, $\varpi = 0.95$, The time period of traces used as the historical check lasts from Feb. 1st 2007 to Mar. 3rd, a one month period.

6.2 Compared Algorithms

Random Selection (Random) In this algorithm, for any given period of time D, we iteratively add new randomly chosen nodes. And then we test whether the encounter between the vehicles in this time period D satisfies the delay bounded coverage requirement γ. Actually, this algorithm supposes the knowledge of the future trace of time D is given. However, it does not take any knowledge of the character of the nodes or the contact properties.

Last Time Optimal (Last Opt) The heuristic behind this algorithm is that we use the near-optimal SD set of the last period of time D to apply to the future D time. It uses the our greedy algorithm for the Determined DB-SDP to sort all the nodes. Then it iteratively adds nodes to the directory set until the delay bounded coverage ratio γ for the next D time is satisfied.

Max Prob. Degree (MAX-DEG) This is a predicting algorithm. First, it uses historical information to get the inter-contact time information. Like our algorithm, it is also based on the probability graph. Instead of calculating the expect gain of μ, it computes the sum of all the meeting probabilities of the adjacent nodes. Then it sorts the nodes according to the probability sum. It iteratively adds the node with the max probability sum, until the expected coverage rate meets the requirement of γ.

6.3 Results

Firstly we present the accuracy of *split-time historical check*. Then we provide the results of our algorithm against other candidate algorithms.

In order to show the effectiveness of our proposed algorithm, we take a wide range of the condition settings of parameter γ and ϖ, and then see whether the condition is satisfied when our iterative algorithm for Non-determined DB-SDP stops. In each pair of γ and ϖ, we use the algorithm to calculate a subset of directory S of delay bound D. We use a history of $T_H = 1$ month.

Then we use this SD set and apply it to the trace of another month. We can get ϖ^* by following a similar routine as the *split-time history check* in Section 5.2 to divide this one month's time by D intervals. D is set to 60 min in this experiment. Then we plot ϖ^* against γ and ϖ. As Fig. 4 shows, in most of the cases, the ϖ^* is very close to the expected value ϖ.

Then we compare the performance against other alternative algorithms by the metric of result SD set size. We take different γ, and compare the SD set size calculated by each of the algorithms. In Fig. 5, we can see as the coverage ratio requirement increases, the size of directory set increases as

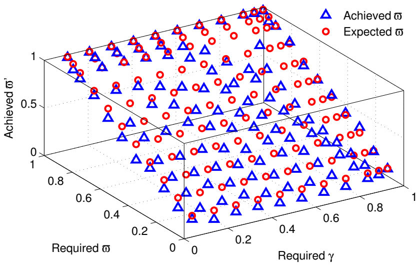

Figure 4: Compare ϖ' and ϖ in different γ and ϖ.

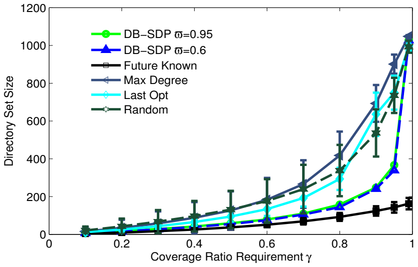

Figure 5: SD set size S vs. delay bound ratio γ.

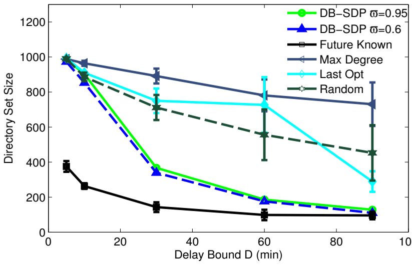

Figure 6: SD set size S vs. delay bound D.

expected. The black line at the bottom shows that knowing the future traces of the vehicles can dramatically reduce the SD set size. We plot the size of the SD set produced by our Non-determined DB-SDP algorithm when ϖ is 0.95 and 0.6. We find that the result sets are very close when ϖ is taken 0.95 and 0.6. This indicates that parameter γ have more impact on the result SD set size than parameter ϖ does. The result SD set of our iterative algorithm is more than 50% smaller than the other alternative algorithms at the expense of only a slight decrease of guarantee ratio ϖ (1 to 0.95). Furthermore, we find that the result given by the LastOpt algorithm is very close to that given by Random algorithm, which means the nearly optimal result of the previous time interval only has the performance of a randomly chosen set. This shows the highly dynamic mobility of the vehicles.

In order to study the impact of different delay bound, we compare the directory set size of each algorithm when the delay bound varies from 5 min to 90 min (Fig. 6). As expected, the SD set selected by each of algorithm decreases as the delay bound D increases. Again we find that with the knowledge of the future traces, the SD set size is greatly reduced. Also, the SD set sizes produced by our Non-determined DB-SDP algorithm are very close when ϖ is taken 0.6 and 0.95 respectively. Again, our Non-determined DB-SDP algorithm outperforms other algorithms by reducing the SD set size by more than 50% with a slight loss of the guarantee ratio (ϖ deceases form 1 to 0.95).

7. CONCLUSION

In this paper, we have studied the Delay Bounded Service Discovery Protocol (DB-SDP) problem in the Urban VANETs. The objective is to minimize the directory size while satisfying the coverage ratio requirement. Firstly we theoretically prove that this problem is NP-Complete even under the assumption of the prior knowledge of the future vehicles traces. Afterwards, to deal with the more realistic situation of unknown traces for the future, we make the key observation that the covered vehicle number follows normal distribution. Then we give a greedy algorithm to tackle the Determined DB-SDP problem and improve it into an iterative algorithm that addresses the Non-determined DB-SDP problem. Trace driven simulation shows our algorithm can select an SD set more than 50% smaller than the alternative algorithms with a slight loss of the guarantee ratio.

8. ACKNOWLEDGEMENTS

This research was supported by NSF of China under grant No.60773091, No.61100210 and Doctoral Program Foundation of Institutions of Higher Education of China under grant No.20110073120021.

9. REFERENCES

[1] K. Arnold, B. Scheifler, J. Waldo, and A. Wollrath. The Jini Specification, 1999.

[2] H. Artail, K. W. Mershad, and H. Hamze. DSDM: A Distributed Service Discovery Model for MANETs. *IEEE Transactions on Parallel and Distributed Systems*, 19:1224–1236, 2008.

[3] C. Campo, M. Munoz, J. Perea, A. Mann, and C. Garcia-Rubio. PDP and GSDL: a New Service Discovery Middleware to Support Spontaneous Interactions in Pervasive Systems. In *Proc. IEEE PERCOM Workshops*, pages 178 – 182, march 2005.

[4] Microsoft Corporation. Universal Plug and Play: Background, 1999.

[5] E. Guttman, C. Perkins, J. Veizades, and M. Day. Service Location Protocol, version 2, 1999.

[6] S. Helal, N. Desai, V. Verma, and C. Lee. Konark - a Service Discovery and Delivery Protocol for Ad-hoc Networks. In *Proc. IEEE WCNC*, volume 3, 2003.

[7] R. Hermann, D. Husemann, M. Moser, M. Nidd, C. Rohner, and A. Schade. DEAPspace - Transient Ad Hoc Networking of Pervasive Devices. *Computer Networks*, 35:411–428, 2001.

[8] U. C. Kozat and L. Tassiulas. Network Layer Support for Service Discovery in Mobile Ad Hoc Networks. In *Proc. IEEE INFOCOM*, volume 3, 2003.

[9] U. C. Kozat and L. Tassiulas. Service Discovery in Mobile Ad Hoc Networks: an Overall Perspective on Architectural Choices and Network Layer Support Issues. *Ad Hoc Networks*, 2:23–44, 2004.

[10] M. Nidd. Service Discovery in DEAPspace. *IEEE Wireless Communications*, 8:39–45, 2001.

[11] V. Raychoudhury, J. Cao, W. Wu, Y. Lai, C. Chen, and J. Ma. K-directory Community: Reliable Service Discovery in MANET. *Proc. IEEE PERCOM*, 7:140–158, 2011.

[12] F. Sailhan and V. Issarny. Scalable Service Discovery for MANET. *Proc. IEEE PERCOM*, 0:235–244, 2005.

[13] F. Wu, H. Zhu, J.-L. Lu, and M.-Y. Wu. On Optimal Service Directory Selection in Urban Vehicular Ad-hoc Networks. Technical report, Shanghai Jiao Tong University, 2012. [Online]. Availabe: http://wirelesslab.sjtu.edu.cn/~fenggang/service-discovery.pdf