

# Reliability-based server selection for heterogeneous VANETs

Seyedali Hosseinienezhad\*, Victor C. M. Leung

Department of Electrical and Computer Engineering, The University of British Columbia, Vancouver, BC, Canada

## Abstract

Heterogeneous wireless networks are capable of providing customers with better services while service providers can offer more applications to more customers with lower costs. To provide services, some applications rely on existing servers in the network. In a vehicular ad-hoc network (VANET) some mobile nodes may function as servers. Due to high mobility of nodes and short lifetime of links, server-to-client and server-to-server communications become challenging. In this paper we propose to enhance the performance of server selection by taking link reliability into consideration in the server selection mechanism, thereby avoiding extra client-to-server hand-offs and reducing the need of server-to-server synchronization. As a case study we focus on location management service in a heterogeneous VANET. We provide a routing algorithm for transactions between location servers and mobile nodes. We assume that location servers are vehicles equipped with at least one long-range and one short-range radio interfaces, whereas regular nodes (clients) are only equipped with a short-range radio interface. The primary goal of our design is to minimize hand-offs between location servers while limiting the delays of location updates. Taking advantage of vehicle mobility patterns, we propose a mobility-aware server selection scheme and show that it can reduce the number of hand-offs and yet avoid large delays during location updates. We present simulation results to show that proposed scheme significantly lowers the costs of signaling and rate of server hand-offs by increasing the connection lifetimes between clients and servers.

**Keywords:** *ad hoc routing, heterogeneous networks, service discovery, VANET*

Received on 3 January 2011

Copyright © 2011 Hosseinienezhad and Leung, licensed to ICST. This is an open access article distributed under the terms of the Creative Commons Attribution licence (<http://creativecommons.org/licenses/by/3.0/>), which permits unlimited use, distribution and reproduction in any medium so long as the original work is properly cited.

doi: 10.4108/icst.trans.mca.2011.e4

## 1. Introduction

Vehicular ad-hoc networks (VANETs) are emerging as one of the most important practical applications of mobile ad-hoc networks (MANETs). As the demand for pervasive computing is increasing, providing services to nodes in the network is also becoming critical. For instance, location management is becoming one of the most important modules in vehicular networking. Multi-media streams, news broadcasting, entertainment and other applications which require Internet connectivity, peer-to-peer applications, local advertisements, vehicle pooling and local cab services are some examples of the broad range of feasible applications when vehicles are equipped with positioning and communication capabilities [1, 2].

### 1.1. Service discovery in MANETs

In MANETs with various devices in the network, a service is a facility or capability that is provided by some of the nodes that can be used by other mobile nodes. These services are usually implemented as an application inside the nodes. Service discovery has been solved in wired networks and single hop wireless networks. DEAPSpace [3] is a service sharing mechanism produced by IBM research for short-range, single hop wireless networks. However, in multihop mobile networks, the physical proximity and mobility of nodes are issues that impact service discovery. Therefore existing approaches may not be suitable for multihop MANETs.

Service discovery protocols consist of four components:

- *Service Description* provides an abstraction of the services being provided. These information help clients

\*Corresponding author. Email: [seyedali@ece.ubc.ca](mailto:seyedali@ece.ubc.ca)

to choose between provided services and connect to the server which is providing required services. Usually a service description is composed of server identification, server characteristics and service characteristics.

- *Service Registration and Advertisement* includes the procedure of storing service descriptions in nodes and advertisement of the services in the network. Depending on the approach, advertisement and registration could be simple flooding of information or based on a hierarchical directory storage approach.
- *Service Discovery* is the method employed by requesters to find services. Service discovery can also include the decision making mechanism to choose between available services. Service discovery and service advertisements are complementary to each other in a system. Therefore, more complexity in one could be traded off for more simplicity in the other. When the services are discovered, the service selection is done based on some criteria.
- *Routing* includes the process of relaying service advertisements, service discovery, service requests and replies. Routing could be done based on different criteria as well. Since we are focusing on VANETs with very high node mobility, the performance of routing is very sensitive to decision making mechanisms used in service discovery. A combination of good advertisement and service discovery methods could decrease the number of needed packet relays and lower the rate of route changes in the network.

There are several service discovery methods designed for MANETs [4–6]. Based on literature, service discovery mechanisms are divided into directory-based vs. directory-less approaches. In networks with high mobility, less reliance on topology information would lead to a better performance because the topology is changing very fast and updating would be very costly. Therefore instead of directory-based methods like [5], directory-less methods have been proposed [4].

In a survey by Mian *et al.* [7] service discovery protocols are categorized based on their performance under various network conditions. Based on this comparison, there is a trade-off between the extent of covered area and supported mobility. There is no one-size-fits-all method that is capable of handling high mobility while the size of network is relatively large.

**Service discovery inspired by field theory.** Lenders *et al.* [4] defined an approach for efficient and robust service discovery. This concept is similar to anycast routing, which is supported in IPv6 [8]. In anycast routing, an address is associated with more than one interface belonging to distinct nodes that are similar in nature. As it is preferable for clients to get service from the nearest among several

potential servers, use of anycasting would allow the desired server to be reached easily.

From electromagnetic field theory, the point potential of a spot is related to its distance to the maxima potential charge. In wireless networks, the most commonly used definition for distance is based on hop count; nonetheless, geographical distance is also applicable. In [4], hop count is considered as the distance between nodes:

$$\varphi(n) = \sum_{i=1}^N \frac{Q_i}{\text{dist}(n, n_i)}, \quad (1)$$

where  $Q_i$  is the potential assigned to server  $i$  and  $\varphi(n)$  is the total received potential by node  $n$  from all servers. The amount of potential assigned to each server could be a factor of their capacity or quality of service (QoS) metrics.

**Reliability consideration in route selection.** Whereas in highly mobile networks, hop-based distance metrics may be unreliable and unstable, and route maintenance may cause extra signaling overhead and delay, route reliability can be a more suitable metric for route selection. Longevity of a route is introduced in [9] as a new metric for route assessment, measured by the association between two nodes. It is assumed that a link is reliable if the association between the two end nodes is higher than a certain threshold. A route-lifetime assessment based routing (RABR) algorithm is proposed in [10], in which route selection is based on hybrid criteria of route lifetime and path length. The route lifetime is measured by the link affinity, which is calculated based on the received signal strength. Since in practice the signal strength varies over time, RABR can make wrong decisions especially in an urban environment with rapid signal fading. It is desirable to utilize the concept of link lifetime as a decision factor in routing, but a new measure of link lifetime tailored for the variable conditions of VANETs is needed.

## 1.2. Location services and server selection problem

With increasing demand for pervasive computing, location management is becoming an essential function in VANETs. Location management is a service for determination of each vehicle's location and making this information available to other nodes. A location management system is generally composed of location servers and clients. Location servers are responsible to hold the location information of clients and to retrieve the information to requesters. Nowadays localization technologies have become widespread and it is a justifiable assumption that all vehicles in a VANET can determine their positions using onboard global position system (GPS) receivers. Therefore we assume that each vehicle can determine its geographic coordinate and report it to the corresponding location server(s). Since connection to stationary

resources may not be guaranteed in VANETs, vehicles should form a location management overlay on the infrastructure-less network. When mobile clients need to register, update or query about location information of other nodes, they should choose a server based on proper decision factors and send their request to it. Choosing a server for location service can be considered a specific case of server selection in service discovery. We shall next provide a short review of current location management approaches and their limitations. We shall consider location management as a specific scenario of service discovery, and propose a method to select location servers in a heterogeneous VANET.

**Location management methods in MANETs.** Several protocols have been designed to handle mobility of nodes [11–18]. Location servers are responsible for handling geographic location information of nodes in the vicinity and provide them to others when needed. Different categories of location management have been classified: flooding-based location management is the most straightforward method for passing location information. Due to high redundancy overhead, researchers have strived to decrease unnecessary packet relays. The hypothesis of methods like distance routing effect algorithm for mobility [12] is that relative locations of closer nodes are changing faster compared to nodes far away from each other. Therefore location updates are sent more frequently to location servers close by than those that are farther away. Quorum-based location management is another category that is based on assuring a rendezvous between queries and updates. A localized quorum-based location service is proposed in [13], which disperses the location of every node horizontally and vertically. As the authors have stated, this method is proper for networks without a significant relative motion. Vehicular ad-hoc networks with high relative speeds are not suitable environments for this class of location management systems.

The GLS method [14] for distributed location service management divides the area into different degrees of grids in a way that in every grid around the node there is a fixed number of servers that collect location information about that node. As grids grow larger, the probability of a server being chosen for other nodes decreases. This method is not very flexible for highly variant environments like VANETs. Hierarchical methods [17, 18] for server allocation are highly scalable because the rates of location updates are reduced for servers in higher levels. However, in VANETs, the number of servers may not be very high and forming a hierarchical structure may not be feasible.

In high mobility networks such as VANETs, keeping track of location information would in general be a huge overhead on the network if location information is saved in all location servers. Therefore having the record of every node in network while nodes are rapidly changing their locations can be performed better if we are able to save

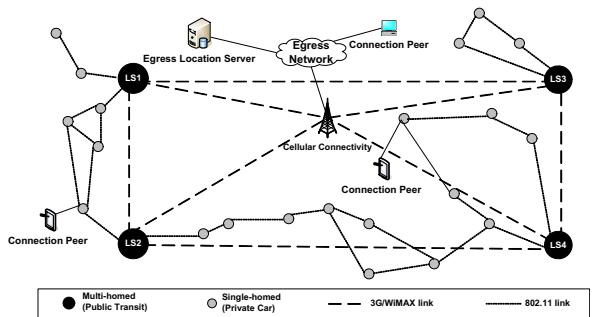
these information in a specific set of nodes. Moreover, to reduce the detrimental effect of mobility, we focus on hop-by-hop packet relaying rather than finding a deterministic route from clients to servers. We consider location management as a service to vehicles, which is offered by some specific nodes in the area. Therefore we need a service discovery mechanism to find the service providers.

Many different wireless access technologies can be employed in VANETs. Short-range technologies include wireless local area networks (WLANs) and its variant called Dedicated Short-range Radio Communications targeting specifically vehicular communications. Long-range wireless technologies include cellular networks and wireless metropolitan area networks such as WiMAX. Vehicular ad-hoc networks employing short-range radio access face problems in area coverage and fast hand-offs between nodes. Because of high mobility speed, rate of hand-offs in the network becomes a limiting factor in location registration and updates. In the presence of these difficulties, heterogeneity can come to the rescue for services like location management. Using long-range wireless access as a higher layer of communications, we can interconnect location servers together as a logical mesh network. We can assume that a connected graph of location servers can exchange signaling messages through this logical mesh network. We shall base our work on utilizing available long-range wireless connections to facilitate location management in VANETs.

In Section 2 we propose an architecture for a heterogeneous VANET to provide location service. After clarifying the problems and challenges in this server selection scenario, we present the proposed method for reliability-based server selection in Section 3. In Section 4 we give the methodology for performance evaluations, present the results and discuss their significance. Section 5 concludes the paper.

## 2. Location management over heterogeneous VANET—the architecture

Figure 1 depicts a heterogeneous VANET architecture with partial Internet connectivity. In this system, nodes



**Figure 1.** Heterogeneous VANET architecture for location management.

equipped for heterogeneous wireless access are connected to each other and edge gateways using their long-range wireless access capability.

The requirements and assumptions in aforementioned architecture are:

- All vehicles are considered to have a mechanism to extract their own geographic location, e.g. using an onboard GPS receiver.
- All nodes are equipped with at least one short-range radio (e.g. 802.11a/b/g/p).
- Some special nodes with heterogeneous wireless access capability are equipped with all valid short-range communication interfaces and one long-range communication interface (e.g. WiMAX). These nodes function as location servers that are interconnected to each other in a logical mesh network to exchange their location records, and to stationary gateways for Internet access.
- In consideration of valuable licensed spectrum used for long-range wireless access, the use of long-range radio should be minimized. Therefore it is our goal to reduce the numbers of queries between servers and server hand-offs for vehicles.
- Location queries and updates should not be propagated more than a certain number of hops.
- Server advertisements should not be rebroadcasted more than a certain limit.

In one scenario of this architecture, a public transportation system provides a wireless Internet relay service inside an urban area. Public transit vehicles are equipped with multiple radios and they are tasked to provide connectivity and related services to other vehicles. They utilize their long-range radios to relay local data network traffic to stationary gateways and to provide a location management service to vehicles in their vicinity by exchanging location information with other location servers.

For location servers to advertise location service and receive updates and queries from vehicles, we propose a service discovery mechanism to find routes to location servers in the area with the best matching mobility pattern. We shall evaluate the effect of this service discovery method with different scenarios of urban and highway mobility.

### 3. Mobility-aware service selection and packet relay

Vehicular mobility patterns (urban or highway) generally follow roadways with random change of directions at intersections. We assume that every vehicle responsibly sends its location information to a location server. This location information can be used by other vehicles or service providers to present location-based services. When

a vehicle and its location server move away from each other and the distance grows more than a certain hop distance, path delay and high link breakage probability make their interactions ineffective. Therefore the client has to hand off from the old server to another server that is better in terms of delay, robustness and lifetime. Every hand-off between two location servers is comprised of several ‘server-to-server’ and ‘server-to-client’ signaling interactions. However, server-to-server interactions are more expensive because they use licensed spectrum to communicate.

Based on expectations and assumptions in the architecture of [Figure 1](#), if we want to use the field theory approach explained before, clients should send their location management packets toward other relays or servers in the vicinity, which have the highest potential. Signaling for a location update comprises of a primary phase of registration between client and server. After the registration, the client is able to update the server periodically or when triggered by specific events. If a client is unable to send updates to the designated server, a new registration with an available server is required. Based on our proposed architecture and location management procedure, the field theoretic method reviewed in [Section 1.1.1](#) has the following deficiencies that should be addressed:

- The measure for distance between nodes is unrealistic, since mobility pattern is not considered in server selection. In our case the relative speed between a client and its server defines the connectivity lifetime and we prefer to choose a server that has a higher connectivity lifetime as long as the path delay is less than a certain limit.
- The server selection is stateless. Service discovery would lead to a set of choices for each relay to forward the packet. However there is no guarantee that a packet will be relayed to the same server to which the former packet is sent. It is desirable for a client to send location updates to a server that it has already registered in. It means that if a client selects a server with highest potential as its location server, all the relay nodes should be notified to relay the packet from that client toward the same server. Consequently a server hand-off does not happen unless the delay threshold is exceeded or disconnection occurs.

By modifying the service discovery method proposed in [\[4\]](#), we propose a location management method that minimizes hand-offs, which is applicable to geographical and topological location management.

#### 3.1. Reliability versus distance

Hop distance is a simple and effective criterion for route selection, but in cases with high mobility this measure is very unstable. To avoid this problem we propose to use

link stability and usability (also known as reliability) as the route selection criteria instead. Denote the set of links in the chosen path between  $s$  and  $d$  as  $P(s, d)$ . We want to account for reliability of each link  $l \in P(s, d)$  and choose a path with highest aggregated reliability. Reliability of a link is directly related to the estimated link lifetime. However calculation of reliability includes error and an unmeasured factor of future alternative connections. For instance, a weak link could be replaced in the future by a new relay node that is not present at the moment. Due to this factor, it is not rational to underrate a path by considering the reliability of the weakest link in path as decision factor. On the other hand, we cannot rely on arithmetic average because strong links in the path would cause overestimation of path reliability.

Since we want the factor to tend toward the most realistic reliability value of the path (to mitigate the impact of links with excellent reliability in calculation of total path reliability), we desire to have the smallest average value as the measuring factor. Instead of using arithmetic mean, we use harmonic mean to calculate the reliability factor. Harmonic means tend toward the smallest values compared to geometric and arithmetic means. Hence, with  $r_l$  being reliability of link  $l$  and the number of hops  $|P(s, d)| = n$  we have:

$$\frac{1}{\frac{1}{n} \sum_{l \in P(s, d)} \frac{1}{r_l}} \leq \sqrt[n]{\prod_{l \in P(s, d)} r_l} \leq \frac{1}{n} \sum_{l \in P(s, d)} r_l.$$

To apply the value of reliability in routing decision, we define the distance factor in (1) as:

$$D(s, d) = \frac{1}{n} \sum_{l \in P(s, d)} \frac{1}{\text{reliability}(l)}. \quad (2)$$

In every calculation period, each node will predict the locations of current neighbors and based on estimated path-loss exponent of environment and foreseen mobility patterns, calculate the link lifetime. Path prediction and lifetime estimation are two major ongoing research topics and they are explained more in Section 3.2. Intuitively, a longer link lifetime leads to a more reliable path between nodes and location servers. Notwithstanding, due to high error rates in prediction mechanisms [19], we cannot rely solely on measures of one link. Therefore we will define *reliability factor* for all  $(node, server)$  pairs to show how reliable the *node* could be to relay packets toward the *server*. Based on the assumption that a higher node density can make the route more reliable, we define the *reliability factor* as the probability of a packet being successfully relayed by a *node* to another which is closer to *server*.

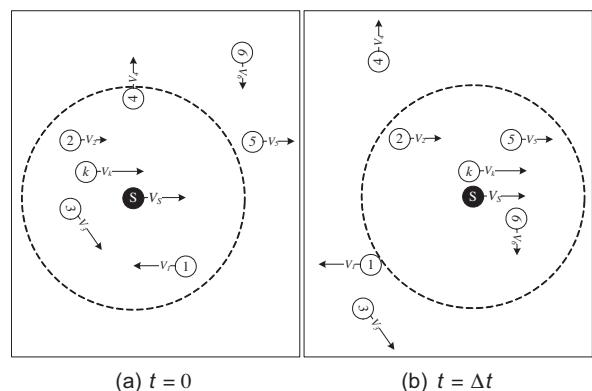
### 3.2. Reliability measurement

We define the link reliability between two neighbors as the estimated expected remaining time of connectivity

between the node pair. To calculate the link reliability we assume that each node will listen to data packets and beacons sent by its neighbors. Using sampled signal characteristics and location information, the receiver predicts how link condition will change in the future. Therefore, to calculate reliability factors, nodes need to have knowledge about future variations in link connectivity. In [20] a method for estimating link residual time and link stability has been proposed. In this method, after denoising and classification of the radio signal strength indications (RSSI) from neighbors, future lifetime is estimated. In [19], Euclidean distance information is utilized to estimate future trajectory. It seems that by using relative mobility between nodes and digital map information together, future estimations can become more precise [21]. A method to calculate the probability of turns in road intersections is proposed in [22], based on the theory that turning options that lead to more destinations in shorter times are more popular than those that lead to local areas or take more time to reach destinations. In [23], mobility behavior of nodes is used to classify their transportation mode. Moreover, using particle filter method they estimate parameters in a Bayesian network for path selection decisions. By learning these parameters, they try to estimate future velocities and turning selections.

**Figure 2** shows an example of vehicular mobilities in two time steps. The variations in links caused by their relative speeds are visible. In this example, nodes 5 and 6 will get into the range of S while 1, 3 and 4 move out of range. Therefore, the reliability of a connection with 5 from S is less than that with k or 2.

To calculate the link reliability in a path toward a server, every node will calculate the cumulative probability of connectivity to next hop for each server. The next hop is defined as any node in communication range that has a loop-free path to the server. In practice every node can put a short hash value of its unique address in forwarded advertisement to avoid considering routes which



**Figure 2.** Variation in neighbors regarding their relative speed.

are originated from itself. Hashing can reduce the length of a string to a few bits and yet avoid duplicate indexes with a high probability.

We assume that a link connectivity prediction method can provide a process consisting of connectivity probabilities during a prediction period. Suppose for a specific environment and mobility pattern, a prediction method is able to predict future motions and channel states for  $n$  time units. Moreover, we can extract the average percentage of errors in lifetime prediction (which can be empirically found) as  $E(\hat{e})$ . This error extraction could be enforced to the system as an *a priori* known measure or it could be readjusted based on observations from the past (by comparing prediction and real condition after it happened). Therefore  $P_{l(i)}(t)$  is the probability of link  $l(i)$  being connected from  $t_0$  (now) until  $t_0 + t$  and is equal to:

$$P_{l(i)}(t) = (1 - E(\hat{e}))P(\hat{L}_i, t), \quad (3)$$

where  $P(\hat{L}_i, t)$  represents the link condition (alive/dead) which is calculated using the desired mobility prediction method. As mentioned before, each mobility prediction and link classification method has a distinct estimation capability in different environments. Therefore values of  $E(\hat{e})$  and  $P(\hat{L}_i, t)$  are highly dependent on the method of prediction being used. We will evaluate our method in [Section 4](#) based on a simple linear prediction, but as long as a prediction method yields a prediction of link lifetime and an estimate of the error, it can be integrated into our approach.

Having estimates of the link lifetimes for all links in the path, the probability of having an undisrupted path from node  $k$  toward the server  $S$  for the next  $t$  time units (complement of the probability of no link being capable of relaying packets from  $k$  to  $S$ ) is:

$$C_S^k(t) = 1 - \prod_{l(i) \in H^k(S)} (1 - P_{l(i)}(t)), \quad (4)$$

where  $H^k(S)$  is the set of links between node  $k$  and its neighbors that have a loop-free path to server  $S$ . We use the cumulative distribution of  $C^k(t)$  (for  $t = 0 \dots t_{\max}$  with  $t_{\max}$  equal to the maximum duration of predictability) as a factor which shows how reliable the node  $k$  is to pass the packets toward server  $S$ :

$$\text{reliability}(k) = \text{cdf}_{t_0}^{t_{\max}}(C_S^k) = \sum_{t=0}^{t_{\max}} C_S^k(t_0 + t). \quad (5)$$

We need to extract the reliability of a node for all servers being discovered. To avoid extra calculations, we define a maximum hop threshold for acceptable potentials received from neighbors. Intuitively it is obvious that information regarding far-away servers are not of any interest because of extra relay overhead and delay.

Finally we define the distance between client  $c$  and server  $S$  as:

$$D(c, S) = \sum_{k \in P(c, S)} \frac{1}{\text{cdf}_{t_0}^{t_{\max}}(C_S^k)}. \quad (6)$$

Notice that as the hop number increases, the distance is affected and the chance of being chosen as best path decreases. This definition of distance would result in such a way that nomadic mobility patterns lead to higher potentials and connection between vehicles with opposite directions and/or sparse connectivity conditions causes less potential dispersion.

### 3.3. Potential assignment for path construction and server selection

Using [\(2\)](#) as the distance measure for [\(1\)](#), every node can receive a potential from servers based on the relative mobility and link condition of all nodes in the path from that server to the node. As described in Algorithm 1 every node advertises all valid server information received from neighbors to adjacent nodes. After receiving these advertisements, the node sets the current potential received from a server to the highest received value. These values are valid up to a certain time after the last advertisement. Whenever the node wants to send or relay a packet toward a server, it will choose the server with the highest potential. This policy leads to selection of a server that has the best mobility correlation with the transmitter.

### 3.4. Location update

To choose a new server for location updates, each node will select the accessible server with the highest potential. After choosing the best server, location updates are sent toward it using the neighbor who has the largest potential from that particular server. Using this approach instead of [\[4\]](#) we can make sure that location update packets will not face misroute to other servers which are not moving in favorable directions. Decision to hand over to another

#### Algorithm 1. Potential assignment.

---

```

1      Input servers_advertisements[]
2      for each servers_info in servers_advertisements
3          L ← servers_info.source
4          predict_link_condition(L)
5          servers[L] ← servers_info
6          for each S in servers_info
7              rel_factor ← reliability (S)
8              D(S.id) =  $\frac{S.\text{pot}.original}{S.\text{pot}.received + rel\_factor}$ 
9              pot[S.id] ← max (pot[S.id],  $\frac{S.\text{pot}.original}{D(S.id)}$ )
10             next_hop[S.id] ← arg maxi ∈ neighbors l.servers[S.id].pot
11         end for
12     end for

```

---

server is performed by a client when the hop distance to the current server has exceeded a certain threshold. Since it is assumed that location update messages are not in a high priority class and their packet size is reasonably small, packet relay in short-range wireless would still be more favorable compared to expensive long-range network. Anyhow, decision for when to hand off is still open to users and they can choose between prompt updates and lower cost. We will discuss about this trade-off in the next section. Packet relay to a chosen server is also done very easily by comparing received potential of the server from current alive links and therefore source routing is not necessary.

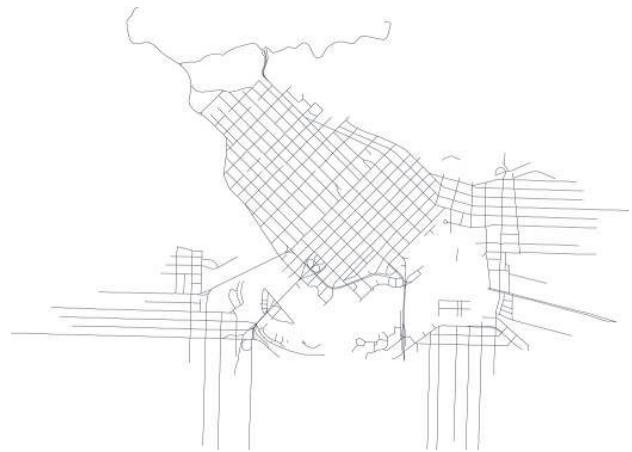
### 3.5. Location query

To find the location of another user, a requester would send a location query to the best available server at the moment (the one with the highest available potential). The packet relay mechanism would be similar to that for location updates. After receiving the query, the server looks up in its local database to see if it has up-to-date location entry. Otherwise it will send a query to its neighbors using long-range wireless. Since we assume that long-range wireless links form a connected graph topology, queries will have answer from one of the servers and this answer will reach the original server.

## 4. Performance evaluations

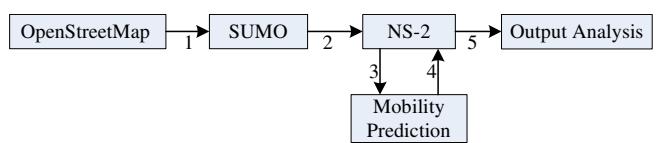
To evaluate performance of our proposed framework, we have modeled the system using the ns-2 network simulator [24]. We have added a new service discovery agent over the currently implemented network stack and added our logic as an application agent. Using application agent, we can use any routing algorithm for packet routing. We have tried our protocol on several test scenarios. These scenarios are based on realistic vehicular traffic generated by the SUMO vehicular traffic generator [25]. This microscopic vehicle traffic generator is able to create mobility patterns based on defined traffic flows, using street maps that are imported into SUMO to generate different test cases. The traces generated by SUMO yield a mobility log for vehicle movements based on road and traffic regulations. We have imported several maps with different key features for evaluations. Transmission range is set to 100 m with 95% confidence interval. This means that a transmission between two cars 100 m away would be successful with 95% probability. For short-range communication we use IEEE 802.11 with the minimum data rate (1 Mbps). We use log distance shadowing with parameters set to values provided in [27].

The first imported map is a 10 km long highway with two lanes in each direction. Two kinds of vehicles have been considered to commute on the road: private vehicles with short-range radios and public transit vehicles



**Figure 3.** Extracted map of Vancouver downtown from OpenStreetMap [26].

equipped with long-range and short-range radios. The two categories of vehicles have different characteristics in speed limit, acceleration and deceleration. The second scenario is a realistic urban area extracted from actual street maps. These maps are extracted from free maps available in OpenStreetMap [26]. Figure 3 depicts an extracted map of the downtown area in Vancouver, BC. Obviously the most significant property of a downtown area is its high number of turning options and signaled intersections. After adding traffic lights to the map, we use SUMO to generate mobility traces for 10000 s. The procedure of map extraction and simulation is shown in Figure 4. After generation of mobility traces, they are fed into ns-2 as mobility scenario and simulation is performed by ns-2. Since we need prediction in our method and it is not performed in ns-2, we do the simulation twice; the first run is done to extract exact location of every vehicle during the simulation. Then we use these data in the next run as a precise prediction of future mobility patterns in network. To make the prediction more realistic, we add noise to location information. Since prediction precision is strongly dependent on prediction mechanism, we use one of the simplest predictions: in every second, each node predicts  $x(t+1) = v + x(t)$ , where  $x(t)$  is the location of node in time  $t$  ( $0 \leq t \leq t_{\max}$ ). The term  $t_{\max}$  is the maximum time that prediction can be reasonably valid. We will set  $t_{\max}$  to a value which:



**Figure 4.** Simulation procedure.

$$E(Pr(C(t))Pr(\hat{C}(t)) + Pr(\overline{C(t)})Pr(\overline{\hat{C}(t)})) > \text{threshold}, \\ 1 < t < t_{\max}. \quad (7)$$

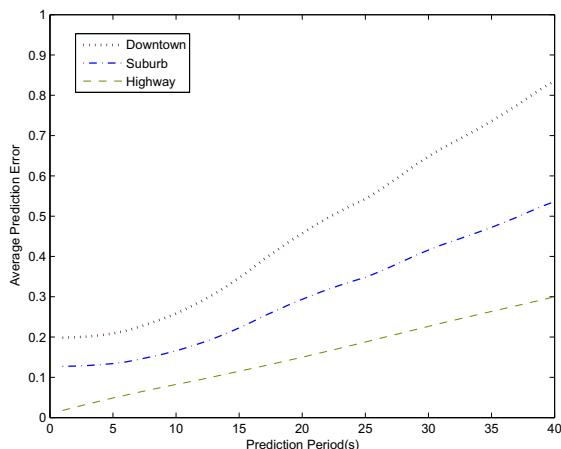
This is the sum of expected probability for having a true guess, whether positive or negative, on having a connection. This probability should be more than a certain threshold. To find this value we run the simulation and calculate the predicted and actual locations in the future. We consider a link as active if its RSSI is more than a threshold. Since measurement of RSSI is impacted by environmental clutters, it is impractical to deterministically define the link connectivity threshold. So we use the propagation model in [28]:

$$P_r(d) = P_t - PL(d) = P_t - (\overline{PL}(d) + X_\delta), \quad (8)$$

where  $\overline{PL}(d)$  is the log-distance path loss from the transmitter to receiver and  $X_\delta$  is a zero-mean Gaussian distributed random shadowing effect with standard deviation  $\delta$ . Values of path-loss exponent and  $\delta$  are usually extracted from empirical data. We have borrowed these values from the experiment done by Otto *et al.* in [27]. Finally, the probability of RSSI being more than  $\gamma$  (dBm) in distance  $d(m)$  is:

$$Pr[P_r(d) > \gamma(\text{dBm})] = Q\left(\frac{\gamma - \overline{P_r}(d)}{\sigma}\right). \quad (9)$$

**Figure 5** shows the estimated error in aforementioned prediction method. Results show that in the highway scenario prediction is close to reality and the connection condition after 40 s is predicted correctly with a 70% probability. However, in the suburban scenarios and downtown areas, nondeterministic stops and turning probabilities cause prediction errors to grow. For downtown scenario we find that predictions are 50% successful only for 20 s ahead. In suburban areas with less stops and turns compared to downtown, it is up to 35 s. We apply



**Figure 5.** Linear prediction average error for three scenarios.

**Table 1.** Variables definition in (10).

$d(i, S)$	Hop distance between $i$ and server $S_i$
$f_u$	Frequency of location update messages
$f_q$	Frequency of location query messages
$LU$	Size of location update message
$LQ$	Size of location query message
$LR$	Size of location reply message
$K$	Usage cost/Kb for long-range network
$N$	Number of location servers
$SYN$	Size of synchronization message

these errors in calculating path reliability factor for each scenario.

To avoid excessive delay caused by late hand-offs we have to set a threshold for maximum allowable hop distance between nodes and servers. The trade-off is between location update cost (which is related to the amount of relayed data and type of media used for it) and end-to-end delay.

$$X = \begin{cases} d(i, S)(f_u \cdot LU + f_q(LQ + LR)) & d(i, S) < \text{thr} \\ K*(N - 1)*SYN + \\ d(i, S_{\text{new}})(f_u \cdot LU + f_q(LQ + LR)) & \text{otherwise.} \end{cases} \quad (10)$$

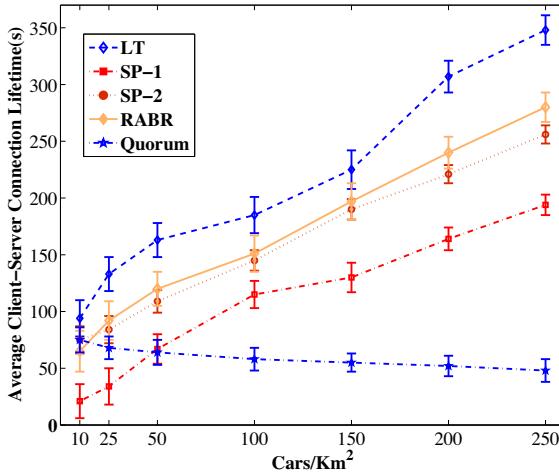
To calculate the proper threshold, we set our objective to minimize Cost \* Delay for location update packets. Using (10) as the cost function and by knowing  $d(i, S)$ , the distance between a node and its second best server with eligible hop distance, every node can calculate the threshold as follows (for definition of variables in (10) see Table 1):

$$\text{thr} = \sqrt{d^2(i, S') + \frac{K*(N - 1)*SYN}{f_u \cdot LU + f_q(LQ + LR)} \cdot d(i, S')}. \quad (11)$$

Here we assumed that delay is only based on hop distance and did not consider the delay caused by collisions.

After finding the estimation errors, we run the algorithm based on these estimation properties. We try to establish connections between servers nodes and regular nodes using the short-range wireless network.

We compare client-server path length and traffic cost with three other methods: [4] (shortest path is the metric for route selection), [10] (affinity based on signal-to-noise ratio) and [13] (quorum-based method with column-based advertisement and row-based query). **Figure 6** shows the average lifetime of connections between mobile nodes and location servers in downtown mobility pattern. Hereafter we shall refer to our method as Life Time based method (LT). SP-1 represents shortest path anycasting based on [4]. In SP-2 we use the same method as SP-1 but whenever a server is selected for a node as a location

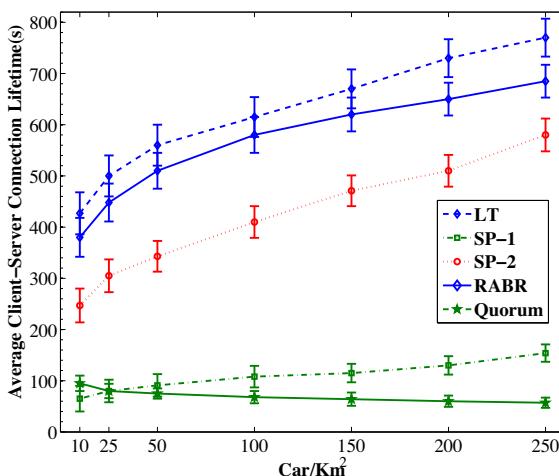


**Figure 6.** Average client-server connection lifetime (downtown).

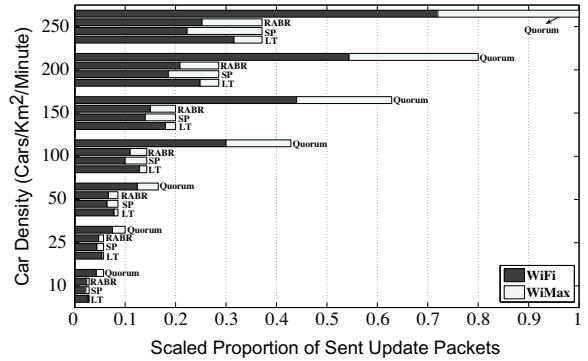
server, it will remain chosen as long as the distance is less than the maximum hop distance. Results show that using lifetime as the distance metric results in significant connection lifetime improvements specially for higher node densities.

In affinity-based method, SNR is considered as the measuring factor for decision making. Therefore for downtown areas with highly volatile SNR conditions RABR cannot perform much better than SP-2. Since in quorum-based method, chosen servers are changed rapidly after any change in topology, connection lifetimes are not comparable to other methods.

Figure 7 shows the same measures as Figure 6 but for highway scenarios. Results show a 57% overall improvement in connection lifetime compared to SP-2. Especially in lower vehicle densities, our proposed method achieves more improvements compared to the shortest path method because of the steady mobility of vehicles which



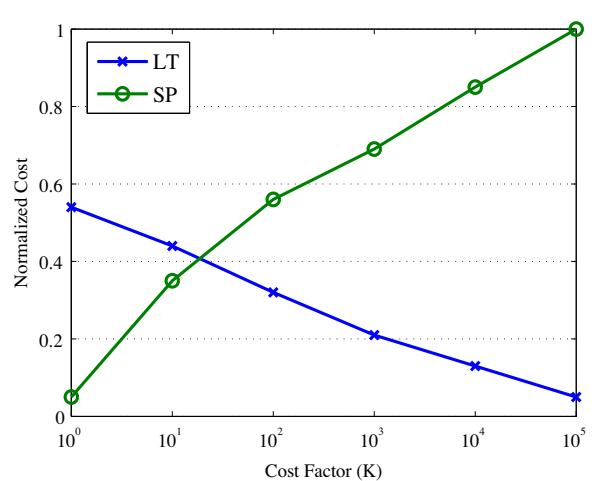
**Figure 7.** Average client-server connection lifetime (highway).



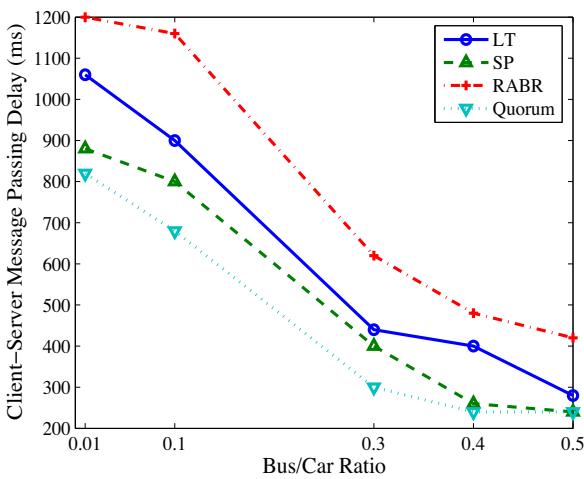
**Figure 8.** Normalized overhead of update packets being sent over WLAN and WiMax ( $K = 100$ ).

leads to higher lifetimes if paths are selected from vehicles moving along the same direction. RABR performs better in highway scenarios due to less perturbations in SNR. However, the quorum-based method exhibits the same behavior as in the downtown scenario. Figure 8 compares the overhead caused by location update packets. The quorum-based method uses several location servers, hence location updates become costly. Moreover, as mobility and interactions between nodes increase, the overhead of quorum-based method increases drastically. We have assumed that the quorum-based method uses WLAN and WiMax based on availability with no preferences. To compare the proportion of WLAN usage, we assumed that parameter  $K$  in (10) is equal to 100 (every transmission on WiMax is 100 time more expensive than WLAN). We can see that usage of WiMax in low traffic densities is significantly low and as mobility patterns grow more dynamic, the difference between LT and other three methods becomes noticeable.

Figure 9 depicts the normalized total cost of location management for different values of  $K$ . Since the cost of RABR is very close to that of SP and the cost of the



**Figure 9.** Signaling cost based on cost factor ( $K$ ) in (10).



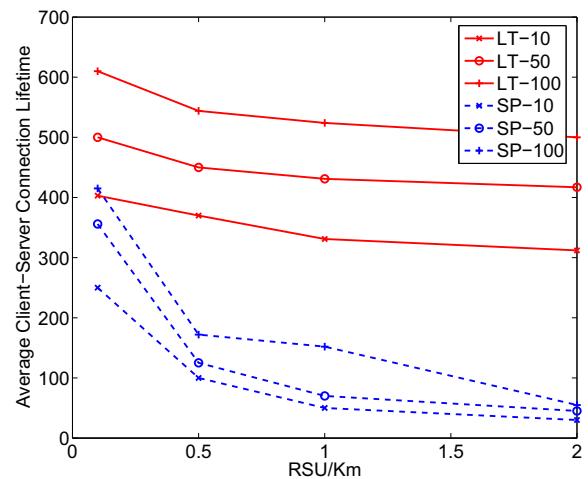
**Figure 10.** Average client–server message passing delay in downtown (only query/response messages are considered for quorum-based method).

quorum-based method is significantly higher than those of other methods, we have only compared the cost of SP vs. LT. As one can observe, for  $K \geq 100$  our method outperforms the shortest path method.  $K$  can be interpreted as a priority or preference parameter and could be tuned based on the trade-off between delay and cost efficiency. If providers prefer faster and more precise location updates, they can decrease  $K$ . In contrast, for better efficiencies (e.g. for less location-sensitive applications) higher values will lead to better spectrum conservation.

Figure 10 shows the average delay experienced by signaling packets (Location Update, Location Query and Responses) for the downtown scenario. It is important to note that the quorum-based method uses more than one location management server and always every immediate neighbor in the same row of the node is acting as a location server. As a result signaling delays for location updates and responses are very low. However, when it comes to location query, signaling delay is relatively higher than updates. Our method is always performing better than RABR in terms of delays but compared to SP, it suffers a 20% increase in delay. As the number of location servers (in this case described as buses) increases, the overall signaling delay for all methods decreases. We can see that if half of the vehicles in our system could act as location servers, delay would have become as low as SP and quorum-based methods.

#### 4.1. Hybrid network scenario

Now we are going to consider a scenario in which infrastructure is available in some positions and road-side units (RSUs) are placed in some spots to provide services of interest. We evaluate the highway scenario with this configuration. To see the effect of including the wired network infrastructure, we place RSUs with specific

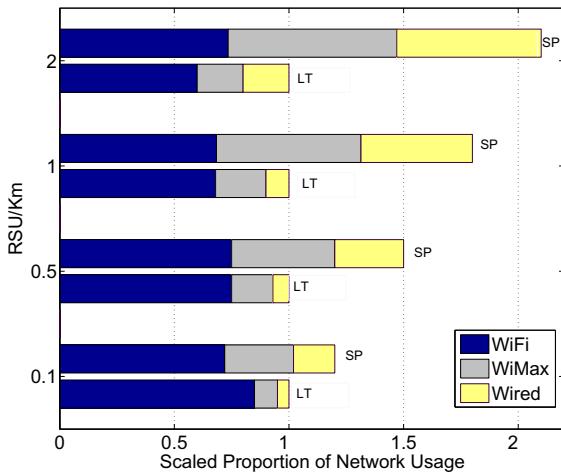


**Figure 11.** Average client–server connection lifetime with hybrid network in highway.

distances between them. RSUs are considered to provide location service to mobile nodes. The selection of location servers will be identical to fully *ad hoc* mode. However location prediction for RSUs is accurate and estimation error is set to zero. We compare the effect of different RSU densities on our lifetime-based method and shortest path method.

In Figure 11 a comparison of average connection lifetime is shown. We have compared the average connection lifetime for three different vehicle densities. Clearly, when the density increases the connection lifetime increases as well. This means in denser traffic conditions the capability to relay the packets between servers and clients helps to increase scalability. Similar to *ad hoc* only network, the connection lifetime of the shortest path method is less than that of the LT method. As the number of RSUs increases, the connection lifetime starts to decrease. In shortest path method, this change is more visible because server selection is based on physical distance. This means when vehicles get close to RSUs, they prefer to change their connection to closer RSU. In contrast, connection lifetime is not dramatically changed in LT method.

Figure 12 compares our LT method with shortest path method in terms of network usage. When the number of utilized RSUs increases, not only utilization of wired network increases, but also usage of WiMax. The reason is that every synchronization caused by selecting a RSU as location server leads to extra synchronizations with WiMax servers. This comparison shows that even if we use RSUs to collaborate with other mobile servers, still there is a chance that selection of RSU as server would result in higher overheads. However, this is the price to pay in order to gain a better reliability. RSUs are essential when the network density is low and connectivity is weak.



**Figure 12.** Scaled proportion of network usage.

## 5. Conclusion

Location management is a critical function in VANETs. In this paper we have assumed that some of the mobile nodes in a VANET are equipped with heterogeneous wireless connectivity. These vehicles are able to act as location servers for other vehicles and cooperate using their long-range radios. We have proposed a new server selection and packet relay mechanism that minimizes the rate of server hand-offs by relaying location update packets toward the server that has the lowest possibility of disconnection. This is done by proposing a new definition of distance. The proposed method has been evaluated by extensive computer simulations. The results show significant improvements in client-server connection lifetimes. Higher connection lifetimes lower the costs of server hand-offs, which require the use of long-range communications to update the record for the client at all the servers. We have provided a tuning factor which can be used for decision making based on tolerable delay and cost. The comparison has been made against three methods: associativity-based routing, shortest path selection and quorum-based location management. Results show that in scenarios with high mobility our method achieves the lowest costs and acceptable delays compared to other three methods. We have also compared our method to the shortest path method in a hybrid network scenario.

**Acknowledgements.** This work was supported by a grant from AUTO21 under the Networks of Centres of Excellence (NCE) Program of Canada.

## References

- [1] BALDESSARI, R., BÖDEKER, B., BRAKEMEIER, A., DEEGENER, M., FESTAG, A., FRANZ, W., HILLER, A. et al. (2007) *C2C-CC Manifesto*. Technical Report, Car 2 Car Communication Consortium, [http://www.car-to-car.org/fileadmin/downloads/C2C-CC\\_manifesto\\_v1.1.pdf](http://www.car-to-car.org/fileadmin/downloads/C2C-CC_manifesto_v1.1.pdf).
- [2] WILLKE, T., TIENTRAKOOL, P. and MAXEMCHUK, N. (2009) A survey of inter-vehicle communication protocols and their applications. *IEEE Commun. Surv. Tut.* **11**(2): 3–20.
- [3] NIDD, M. (2002) Service discovery in DEAPspace. *IEEE Pers. Commun.* **8**(4): 39–45.
- [4] LENDERS, V., MAY, M. and PLATTNER, B. (2005) Service discovery in mobile ad hoc networks: a field theoretic approach. *Pervasive Mob. Comput.* **1**(3): 343–370.
- [5] KOZAT, U. and TASSIULAS, L. (2003) Network layer support for service discovery in mobile ad hoc networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies* (IEEE), **3**: 1965–1975.
- [6] TYAN, J. and MAHMOUD, Q. (2004) A network layer based architecture for service discovery in mobile ad hoc networks. In *Electrical and Computer Engineering, 2004. Canadian Conference* (IEEE), **3**: 1379–1384.
- [7] MIAN, A., BALDONI, R. and BERALDI, R. (2009) A survey of service discovery protocols in multihop mobile ad hoc networks. *IEEE Pervasive Comput.* **8**: 66–74.
- [8] HINDEN, R. and DEERING, S. (1998) Ip version 6 addressing architecture, rfc 2373, <http://ietfreport.isoc.org/rfc/rfc2373.txt>.
- [9] TOH, C. (1997) Associativity-based routing for ad hoc mobile networks. *Wireless Pers. Commun.* **4**(2): 103–139.
- [10] AGARWAL, S., AHUJA, A., SINGH, J. and SHOREY, R. (2000) Route-lifetime assessment based routing (RABR) protocol for mobile ad-hoc networks. In *IEEE International Conference on Communications* (Citeseer) **3**: 1697–1701.
- [11] FRIEDMAN, R. and KLIOT, G. (2006) *Location Services in Wireless Ad Hoc and Hybrid Networks: A Survey*. Technical Report, Department of Computer Science, Technion.
- [12] BASAGNI, S., CHLAMTAC, I., SYROTIUK, V. and WOODWARD, B. (1998) A distance routing effect algorithm for mobility (DREAM). In *Proceedings of the 4th Annual ACM/IEEE International Conference on Mobile Computing and Networking* (ACM), 76–84.
- [13] LIU, D., STOJEMENOVIC, I. and JIA, X. (2006) A scalable quorum based location service in ad hoc and sensor networks. In *2006 IEEE International Conference on Mobile Ad hoc and Sensor Systems (MASS)* (IEEE), 489–492.
- [14] LI, J., JANNOTTI, J., DE COUTO, D.S., KARGER, D.R. and MORRIS, R. (2000) Scalable location service for geographic ad hoc routing. In *Proceedings of the Annual International Conference on Mobile Computing and Networking(MOBICOM)* (ACM), 120–130.
- [15] SALEET, H., LANGAR, R., BASIR, O. and BOUTABA, R. (2008) Proposal and analysis of region-based location service management protocol for VANETs. In *IEEE Global Telecommunications Conference (GLOBECOM)* (IEEE), 491–496.
- [16] DIKAIAKOS, M.D., FLORIDES, A., NADEEM, T. andIFTODE, L. (2007) Location-aware services over vehicular ad-hoc networks using car-to-car communication. *IEEE J. Sel. Areas Commun.* **25**(8): 1590–1602.
- [17] KIEß, W., FÜSSLER, H., WIDMER, J. and MAUVE, M. (2004) Hierarchical location service for mobile adhoc networks. *ACM SIGMOBILE Mob. Comput. Commun. Rev.* (*MC2R*) **8**(4): 47–58.

- [18] AHMED, S., KARMAKAR, G. and KAMRUZZAMAN, J. (2009) Hierarchical adaptive location service protocol for mobile ad hoc network. In *Proceedings of the 2009 IEEE Conference on Wireless Communications & Networking Conference* (Institute of Electrical and Electronics Engineers Inc.), 2932–2937.
- [19] HAAS, Z.J. and HUA, E.Y. (2008) Residual link lifetime prediction with limited information input in mobile ad hoc networks. In *Proceedings of IEEE INFOCOM* (IEEE), 26–30.
- [20] SOFRA, N. and LEUNG, K.K. (2009) Link classification and residual time estimation through adaptive modeling for vanets. In *IEEE VTC* (IEEE), 1–5.
- [21] MENOUARAND, H., LENARDI, M. and FILALI, F. (2007) Movement prediction-based routing (MOPR) concept for position-based routing in vehicular networks. In *IEEE Vehicular Technology Conference* (IEEE), 2101–2105.
- [22] KRUMM, J. (2009) Where will they turn: predicting turn proportions at intersections. *Personal and Ubiquitous Computing*, Springer, <http://www.springerlink.com/content/n41v0m2q08g84554/>.
- [23] PATTERSON, D., LIAO, L., FOX, D. and KAUTZ, H. (2003) Inferring high-level behavior from low-level sensors. *Lect. Notes Comput. Sci.* **2864**: 73–89.
- [24] The network simulator ns-2, <http://www.isi.edu/nsnam/ns/>.
- [25] Simulation of urban mobility, <http://sourceforge.net/apps/mediawiki/sumo/>.
- [26] OpenStreetMap, <http://www.openstreetmap.org/>.
- [27] OTTO, J.S., BUSTAMANTE, F.E. and BERRY, R.A. (2009) Down the block and around the corner: the impact of radio propagation on inter-vehicle wireless communication. In *Proceedings—International Conference on Distributed Computing Systems* (Montreal, QC, Canada: IEEE), 605–614.
- [28] RAPPAPORT, T. (2001) *Wireless Communications: Principles and Practice* (Upper Saddle River, NJ, USA: Prentice Hall PTR).