Location-aware service discovery on IPv6 GeoNetworking for VANET

Noguchi Satoru, Manabu Tsukada, Thierry Ernst, Atsuo Inomata, Kazutoshi Fujikawa

Location-aware service discovery on IPv6 GeoNetworking for VANET

Satoru Noguchi[‡], Manabu Tsukada[*†], Thierry Ernst[*†], Atsuo Inomata[‡], Kazutoshi Fujikawa[‡]
[*]INRIA Rocquencourt - IMARA Project-Team - Domaine de Voluceau, BP 105 78153, Le Chesnay, FRANCE
Email: {manabu.tsukada, thierry.ernst}@inria.fr
[†]Mines ParisTech - CAOR Lab - 60, Boulevard Saint-Michel 75272 Paris, FRANCE
[‡]Graduate school of Information Science, Nara Institute of Science and Technology, Japan
Email: satoru-n@is.naist.jp, {atsuo, fujikawa}@itc.naist.jp

***Abstract*—Service discovery is an essential component for applications in vehicular communication systems. While there have been numerous service discovery protocols dedicated to a local network, mobile ad-hoc networks and the Internet, in vehicular communication systems, applications pose additional requirements; They need to discover services according to geographical position. In this paper, we propose a location-aware service discovery mechanism for Vehicular Ad-hoc NETwork (VANET). The proposed mechanism exploits IPv6 multicast on top of IPv6 GeoNetworking specified by the GeoNet project. Thanks to the GeoBroadcast mechanism, it efficiently propagates service discovery messages to a subset of nodes inside a relevant geographical area with encapsulating IPv6 multicast packets. We implemented the mechanism using CarGeo6, an open source implementation of IPv6 GeoNetworking. Our real field evaluation shows the system can discover services with low latency and low bandwidth usage in VANETs.**

I. Introduction

Applications for Intelligent Transportation System (ITS) aim at providing road users with improved traffic safety, traffic efficiency, and additional values in vehicular communication systems [1]. Recently various ITS stakeholders have been working on specifying ITS applications [2], [3]. Such ITS applications are distributed applications composed of a number of distinct *services*; software or hardware entities integrated into wide variety of nodes in vehicular ad-hoc networks (VANET), in which most participants are mobile nodes embedded in vehicles.

As applications need to orchestrate necessary services remotely, a Service Discovery Protocol (SDP) that dynamically discovers available services is essential. Although we can consider to introduce a static configuration, a centralized directory server that provides information of available services, or traditional broadcast to discover services, such solutions are not applicable because of the characteristics of ITS services: (i) services are mostly nomadic in VANETs, thus available services in a VANET are time-varying, (ii) a VANET may not be capable of introducing a stable node (i.e. centralized entity). On the other hand, SDPs enable applications to dynamically find the existence, characteristics and communication endpoints of services by querying service's attributes (possibly in cooperation with centralized entities). Even though the physical and logical location of services change frequently, SDPs can efficiently discover a set of actually available services.

Although A number of SDPs have been proposed to find nomadic services within mobile ad hoc networks (MANETs), discovering ITS services in VANETs raises further requirements: (i) applications need to discover services according to geographical location because ITS-related services are highly dependent on geographical location, such as cameras embedded in a particular intersection, vehicles within a certain radius of a corner, etc. (ii) Services should be discovered as quickly as possible due to stringent latency requirements of ITS applications (i.e. from 10ms to 1000ms [4]). (iii) ITS applications need to avoid consuming unnecessary bandwidth.

A potential solution is to use multicast in cooperation with ad-hoc routing protocols so that it can efficiently react to the change of the network topology due to the mobility of the vehicles and also avoid unnecessary broadcasting (i.e. flooding) to all vehicles in the area. IPv6 multicast-based SDPs, specifically Service Location Protocol version 2 (SLPv2) [5] and multicast DNS (mDNS) with DNS-based service discovery (DNS-SD) [6], [7], may therefore be possible foundation for service discovery in VANET.

However, at this moment such solutions do not support the ”geographic service discovery” requirement (i) mentioned above. In addition, they can manage geographical location as a service's attribute within the SDP mechanism in the application layer, it means that service discovery messages are delivered to unnecessarily large number of nodes since IPv6 multicast itself sends packets to all nodes which belong to a corresponding multicast group that represents a particular type of service. Consequently the requirement ”efficient bandwidth consumption” (iii) mentioned above is not met either.

In this paper, we propose a service discovery mechanism that locates services inside a particular geographical area in VANETs. The proposed mechanism is composed of IPv6 multicast-based service discovery in combination with geographical addressing and routing; SLPv2 with RFC 3111 [8] modification by IETF, and IPv6 GeoNetworking defined by the GeoNet project[1]. Modifications of SLP for IPv6 specified in RFC 3111 enable SLP to use multiple IPv6 multicast groups which allows one-multicast-address-for-one-service usage. IPv6 GeoNetworking furthermore enables

[1]The FP7 European project GeoNet. http://geonet-project.eu/

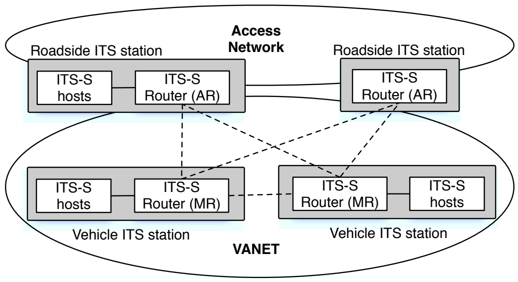

Fig. 1. VANET in vehicular communication systems

to deliver regular IPv6 packets according to geographical location. As a result, upper layer entities can transparently use the geographical routing functionality as legacy IPv6.

The rest of this paper is organized as follows. Section II overviews vehicular communication systems, service discovery, existing SDPs, and IPv6 GeoNetworking. Section III describes functional requirements of SDPs for ITS applications and then proposes our location-aware service discovery mechanism harmonized with SLPv2 and IPv6 GeoNetworking. We then present field evaluation results in Section IV. Section V finally concludes the paper.

II. Service discovery in vehicular communication systems

In vehicular communication systems, VANETs are composed of vehicles and the roadside.

In our study, the ITS equipment deployed in the vehicles and roadside infrastructure comply with the *ITS station reference architecture* from ISO/ETSI [9], [10], [1]. Each ITS station is assumed to be equipped with at least a router i.e. mobile router (MR for the vehicle ITS station) or access router (AR for the roadside ITS station). Other nodes (e.g. hosts running applications, cameras, gateways to the CAN bus, ...) are possibly connected to the routers through an ITS station internal network. Routers have at least (i) one wireless egress interface to communicate with other routers, and (ii) one wired/wireless ingress to connect to the ITS station internal network. IPv6 works as a mandatory network layer communication protocol and the routers provides certain network prefixes to their attached nodes. The AR provides Internet access to MRs. Fig. 1 shows the network architecture of the vehicular communication system.

ITS applications are distributed applications composed of several different *services*: software or hardware entities integrated into attached nodes developed by several ITS stakeholders. As applications may consume multiple services through an identical application process, services shall be self-contained, modular, and application independent entities so that service consumers can share and reuse existing services. Possible functions of services in the vehicular communication systems are (i) to provide vehicle's/roadside's characteristics (e.g. vehicle's mechanical condition, colors of traffic light, etc.), or sensor data acquired with embedded camera, radar, etc. (ii) to process consumers' request (e.g. manipulate electronic gates, perform payment, notify drivers with road traffic information) and (iii) to aggregate road traffic information from other vehicles and the roadside.

Prior to communicate with necessary services, applications obviously need to know the existence and communication endpoints using SDPs that dynamically discover actually available services. Thanks to SDPs, applications only need to specify a name and specific attributes of the service, then SDPs return a list of actually available services containing communication endpoints of attached nodes that operate the corresponding service.

SDPs are composed of service consumer (SC), service provider (SP), and service directory (SD). SPs operate services for SCs and possibly register available services to SDs, while SCs try to discover services by asking SPs and SDs. SDs are centralized cache entities that act as services' directory. SDs may not be introduced to a small network (i.e. local network), or mobile network (i.e. infrastructure-less VANET) in which such a centralized entity does not fit the characteristics of the network. From the architectural point of view, SDPs are considered as a set of mechanisms such as service description language, discovery function, registration function, and routing for each function [11]. In this paper we particularly concentrate on the functions and the routing mechanism.

A. Service discovery protocols

SLPv2 standardized by IETF introduces three system components: User Agent (UA), Service Agent (SA), and optional Directory Agent (DA), which behave as the above-mentioned SC, SP, and SD respectively in the context of this paper. In SLPv2, a SC issues a Service Request (*SrvRqst*), which contains a type and attributes of the requested service, to SPs or SDs. If SPs and/or SDs satisfy the request, they return a Service Reply (*SrvRply*), which contains URL representation of all available services in the considered network, to the SC. SPs join a particular IP multicast group, then the SC can either directly send *SrvRqst* to SPs via IP multicast or to SDs via unicast. On the other hand, SPs always return *SrvRply* to SCs via unicast.

RFC 3111 provides a modification of SLPv2 and enables SLPv2 to operate service discovery over IPv6. While the original SLPv2 is using only one IPv4 multicast address, the modification allows to use multiple IPv6 multicast addresses assigned for each service (available address range is FF0X::1:1000-FF0X::1:13FF). SPs join the multicast groups that correspond to the service type of their services. The multicast address is calculated according to a hash algorithm, which generates a numerical value (0-1023: corresponds to the multicast address range) from a service type's string representation. From the communication point of view, the benefit of this modification is to send the service discovery messages to a specific subset of nodes that join a corresponding IPv6 multicast group in the considered network. If there is a large number of SPs that operate several different services in a network, such a modification can significantly reduce bandwidth usage.

mDNS with DNS-SD also provides a service discovery mechanism by introducing a special DNS domain '.local.' It discovers services using the regular DNS message format via IP multicast without DNS servers, thus SCs directly communicate with SPs. Unlike SLPv2 with IPv6 modification, mDNS uses one IP multicast address for all discovery messaging.

In static and/or local networks, each SDP mechanism efficiently works thanks to IPv6 multicast, however, as we mentioned above, ITS applications need to discover services according to geographical position in VANET as quickly as possible without unnecessary bandwidth consumption.

B. IPv6 GeoNetworking

The GeoNet project has developed a reference specification of IPv6 operated over GeoNetworking, which conforms to the C2CNet specification [2], [12]. C2CNet, specified by the CAR 2 CAR Communication Consortium[2], is a communication layer dedicated to car-to-car communications and is located between the network layer and the link layer. It supports geographical addressing and routing by means of encapsulation of IPv6 packet within a C2CNet packet containing GeoDestination (coordinates of the geographic area where the packet should be distributed).

Although the C2CNet layer can exchange packets without IP, the GeoNet project has defined how to transmit IPv6 packet over C2CNet ("IPv6 over C2CNet") so that IPv6 and GeoNetworking can be combined. This combination allows to distribute IPv6 multicast packets to a given GeoDestination. This is performed transparently to the upper layers; In IPv6 GeoNetworking-enabled VANETs, each MR is assigned a C2CNet identifier. When a node sends out an IPv6 packet with destination node's IPv6 address, the C2CNet layer encapsulates the IPv6 packet within a C2C packet, which includes the C2CNet identifier of the IP next hop node. The C2CNet layer thereby makes the routing decision with the C2CNet identifier and nodes' geographical location. IPv6 GeoNetworking has four types of geographical routing mechanisms: GeoUnicast, GeoBroadcast/TopoBroadCast, and GeoAnycast. Depending on the mechanisms, several types of the geographical destinations (GeoDestination) can be specified with geographical coordinates and descriptions of a shape, such as a circle area with a particular radius of a geographical position. These geographical routing mechanisms are mapped to IPv6 unicast, multicast, and anycast so that upper layer entities i.e. applications/services can transparently use these mechanisms without any interaction with the C2CNet layer. It means that the users only need to support the regular IPv6 stack. In the GeoNet project, IPv6 GeoNetworking has been implemented in Linux using TUN virtual interface. Routers at first receive regular IPv6 packets on their ingress (egress) interface and pass the packets to the userland C2CNet module via the virtual interface. Then subsequent communication is performed in the C2CNet layer [13], [14].

[2]The CAR 2 CAR Communication Consortium. http://www.car-to-car.org/

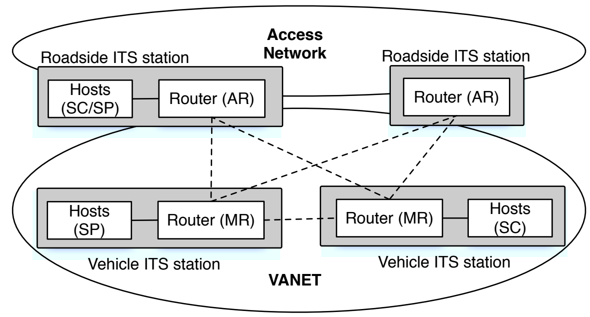

Fig. 2. System overview

III. Location-aware service discovery on IPv6 GeoNetworking

In this section, we propose a location-aware service discovery mechanism. Our solution consists of application layer service discovery components and the IPv6 GeoNetworking components in the network layer. We use SLPv2's service discovery components because of its ability to handle multiple IPv6 multicast groups, as mentioned in Section II. The IPv6 GeoNetworking components constructs and maintains VANETs. The descriptions of these components are described below.

A. Assumptions and system requirements

The proposed mechanism aims at discovering services for ITS applications in VANETs described in Section II. We assume that each attached host has a global IPv6 address configured from a network prefix assigned by its router. VANETs are made of vehicle and roadside ITS stations routers with IPv6 GeoNetworking capabilities. Multi-hop routing is performed at the C2CNet layer transparently from the IP layer. The routers are therefore one IP hop neighbors from the IPv6 point of view. Each router can get its current geographical position via embedded GPS. A service is basically identified with a service type and optional service specific attributes. It means that there may be multiple services that operate an identical service type with different service attributes.

The design principles are as follows:

- Discover services according to geographical position in addition to the services type and attributes: when SCs try to discover SPs, each SC shall specify a GeoDestination within which SPs can be discovered. The size of the GeoDestination is not dependent on the wireless communication range of the sender but just dependent on application/services' requirements.
- Specify GeoDestination for each service: each service shall be able to use a service-specific geographical range separately.
- Avoid unnecessary bandwidth usage: the service discovery mechanism should not send discovery messages to unnecessarily large number of nodes.

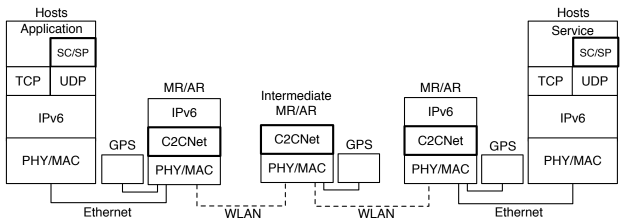

Fig. 3. Proposed Service Discovery stack

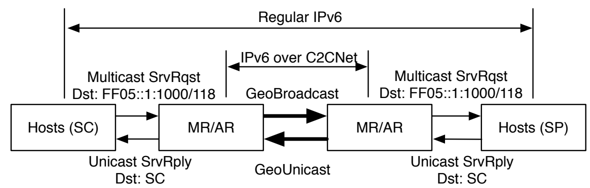

Fig. 4. Encapsulation of SLP messages

B. Directory-less SLP

SLPv2 components are used with the IPv6 modification, as mentioned in Section II. SDs are not introduced since we focus on mobile services located in VANETs where such a static centralized entity is hard to be installed. Whenever a SC tries to discover a service, it issues the multicast *SrvRqst* message to a corresponding IPv6 multicast group calculated by the service type of the requested service. When SPs, which join the corresponding IPv6 multicast group, receive the *SrvRqst* message and they operate services that meet the request, they return the unicast *SrvRply* message to the SC. We use the site-local scope IPv6 multicast (i.e. FF05::1:1000/118) instead of the link-local scope because SCs and SPs are mostly out of link-local scope; they are located behind different ITS station (ITS-S) routers.

Contrary to the original SLPv2, in the proposed mechanism SCs can specify an arbitrary GeoDestination for each service discovery process as described in the subsequent section.

C. Multicast service request over GeoBroadcast

In order to deliver the multicast *SrvRqst* message to a subset of SPs inside a particular GeoDestination in a VANET, we extend SCs to exploit the GeoBroadcast mechanism by IPv6 GeoNetworking, which disseminates IPv6 multicast packets to all nodes located inside a GeoDestination represented by its coordinates and radius [15].

The GeoNet specification proposes several solutions to determine how to map the GeoDestination to a group of destinations. For instance, an IPv6 multicast address can be statically mapped to a corresponding GeoDestination using a configuration file that assigns a GeoDestination with a radius around the centre of the area where the packet shall be propagated (i.e. FF0E::1 corresponds to a 500m radius). It means that application/service users need to specify all possible pairs of IPv6 multicast address and radius.

However, the solution relying on the static configuration does not meet the above-mentioned requirements, because the corresponding IPv6 multicast address for each service is not statically configured but calculated using the hash function. Therefore we introduce a mechanism that specifies GeoDestination by adding an *<IPv6 multicast addresses, radius>* mapping into IPv6 GeoNetworking. SCs send the mapping information to their ITS-S routers.

Regarding multi-hop IPv6 multicast routing from an attached node to others in *SrvRqst*, the multicast packets are propagated using simple IP multicast forwarding without any dedicated routing mechanism, because ITS-S routers can communicate with each other as one-hop IP neighbor thanks to IPv6 GeoNetworking. It means that while the maintenance of multicast groups in VANETs traditionally incurs some costs, the proposed mechanism does not need such overhead. Fig. 2, 3, and 4 shows the system overview, protocol stack in each system entity, and the overview of the encapsulation of the SLP messages respectively.

The operation processes of the proposed mechanism are identified with three phases: (i) service activation, (ii) service discovery, and (iii) service operation. Overall processes are shown as follows:

1) In the service activation phase, when a service is activated, its SP joins an IPv6 multicast group determined with the hash function by means of MLD report to its ITS-S router.
2) In the service discovery phase, when an application requests its SC to discover services inside a particular GeoDestination, the SC calculates the corresponding IPv6 multicast address and sends the mapping information, which contains a pair of *<IPv6 multicast address, radius>*, to the IPv6 GeoNetworking mechanism in its ITS-S router.
3) The SC issues the multicast *SrvRqst* designated to the corresponding IPv6 multicast address to its ITS-S router. The ITS-S router receives the IPv6 multicast packet through its ingress interface and forwards it to the internal C2CNet virtual interface.
4) The IPv6 GeoNetworking mechanism determines the GeoDestination by looking up the registered mapping information. Then the IPv6 multicast packets are encapsulated into the GeoBroadcast packets and sent out on the egress interface.
5) ITS-S routers located inside the GeoDestination receive the GeoBroadcast packets through egress interface. The packet is decapsulated and then the IPv6 layer checks if there are attached nodes belonging to the corresponding multicast group (i.e. a SP that operates the requested service) on their ingress interface. If there are corresponding SPs, the routers send the IPv6 multicast packets to its SPs through their ingress interface.
6) In the end of service discovery phase, SPs reply unicast *SrvRply* to the SC.
7) Finally, the application starts to consume the service in their communications.

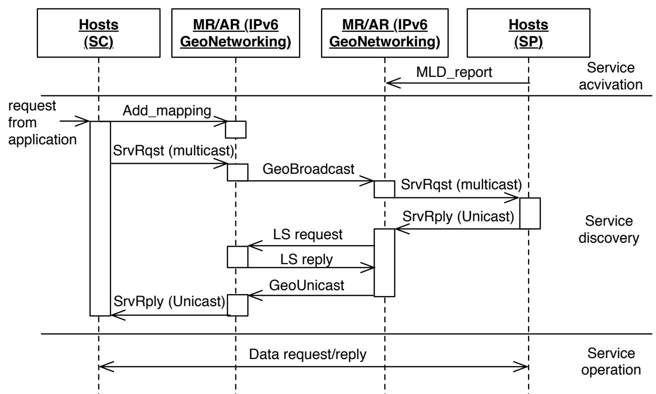

Fig. 5. Messaging sequence. *LS* is the abbreviation of *Location Service*, control messages of IPv6 GeoNetworking.

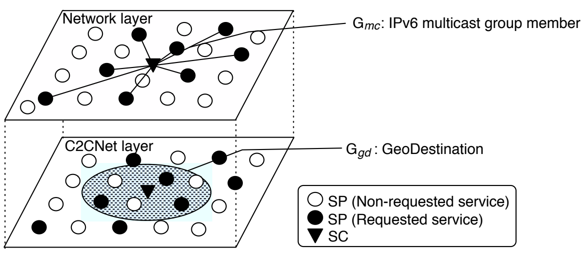

Fig. 6. Propagation of multicast SrvRqst packets.

Fig. 5 shows the overall messaging sequence.

Suppose a set of all nodes in the considered network N, a group of all nodes that join the corresponding IPv6 multicast group G_{mc}, a group of all nodes being inside the corresponding GeoDestination G_{gd}. Only if a node $N_i \in N$ fulfills $(N_i \in G_{mc}) \cap (N_i \in G_{gd})$ can receive *SrvRqst* packets. Fig. 6 shows the selective propagation of multicast *SrvRqst* packets. Thanks to this mechanism and the SLPv2's per-service IPv6 multicast address assignment, the proposed mechanism can avoid propagating service discovery messages unnecessarily large GeoDestination.

IV. Experiments

In order to observe the cost and performance of the proposed mechanism, we have implemented a prototype system and integrated it into our real field testbed.

A. Field evaluation setup

We implemented the system by extending OpenSLP 2.0 Beta 1[3] and CarGeo6[4]on Linux operating system. OpenSLP is an open-source implementation of SLP including the modification for IPv6, whereas CarGeo6 is an open-source implementation of IPv6 GeoNetworking in complying with the reference specification of the GeoNet project. Regarding the IPv6 multicast forwarding function in ITS-S router, we implemented our own multicast forwarding daemon.

[3]The OpenSLP project. http://www.openslp.org/

[4]The CarGeo6 project. http://www.cargeo6.org/

TABLE I
System configuration

Entity	Parameter	Configuration
all Hosts	Number of services	100
Host1	Service discovery frequency	1Hz
	Radius of GeoBroadcast	Random (50m-200m)
all MRs	GPS position update frequency	1Hz
MR1, Host1	Driving speed	0-20km/h
	Distance from SP1(Single hop)	10m - 100m
	Distance from SP2(Multi hop)	120m - 200m

We integrated the system into our field testbed facilities including three sets of MRs and attached hosts, which can be capable of testing multi-hop communication in a VANET. MRs are equipped with one Ethernet port as an ingress interface, and one wireless 802.11 b/g card used as an egress interface. Ubuntu 10.10 (kernel 2.6.35.11) is installed on all nodes. MRs can get current geographical position through GPS receiver connected via their USB port. In order to get coordinates, gpsd-2.96[5] is installed as a local TCP server. While routers run CarGeo6 and the IPv6 multicast forwarding daemon so that they operate IPv6 GeoNetworking and IPv6 multicast forwarding, attached hosts are conventional PCs that only support OpenSLP and legacy IPv6 multicast with MLD multicast capabilities. Applications and services only need to manipulate a set of OpenSLP functions.

We performed a field evaluation at NAIST campus in Japan. As we use three sets of MRs and attached hosts, one of them (MR1, Host1) runs a SC in Host1, which moves around the other nodes (MR2, Host2, and MR3, Host3) and periodically tries to discover services with randomly selected GeoBroadcast radius. On the contrary, the others (MR2, Host2, and MR3, Host3) are stationary nodes located along a road. Host2 and Host3 operate a SP for each (SP1, SP2). The position of MR2 and Host2 is inside the direct communication range of MR1 and Host1. MR3 and Host3 are not inside the direct communication range of MR1 and Host1 but of MR2 and Host2. It means that the SC can reach SP1 via single hop whereas it can SP2 via multiple hops.

Each SP runs the same pairs of multiple services. Therefore when both SPs are inside a GeoDestination of a particular *SrvRqst*, the SC can receive two service information for one request. The system configuration is shown in Table I.

B. End-to-end latency

Table II shows the discovery latency between (i) SC and SP1, and (ii) SC and SP2 in the succeeded service discovery phase. The average latency is 7.86ms in the single hop case and 48.6ms in the multi hop case. It shows our proposed mechanism can discover services with fairly low latency. As several ITS related research mentioned that the acceptable latency is around 10-1000ms, our proposed service discovery mechanism meets this requirement even in the two-hop case. The evaluation shows preliminary results in which maximum hop is the two-hop case, therefore more experiments will

[5]gpsd - a GPS service daemon. http://gpsd.berlios.de/

TABLE II
END-TO-END LATENCY AND DISCOVERY SUCCESS RATE

	End-to-End latency (ms)			Success rate
	Min	Max	Average	
SP1 (1hop)	3.59	23.3	7.86	75%
SP2 (2hop)	27.9	170	48.6	52%

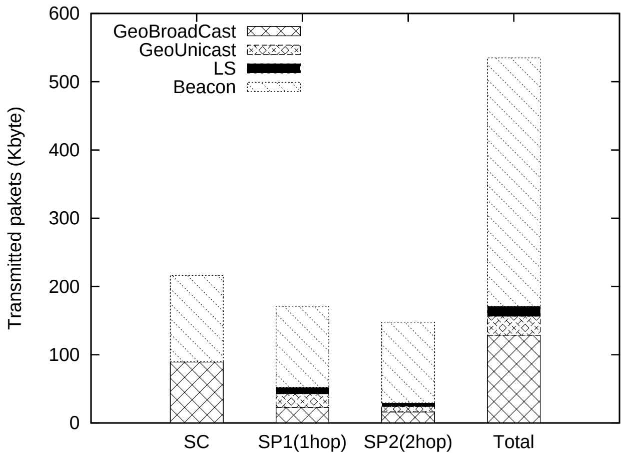

Fig. 7. Message overhead

be conducted in future work. Although the system discover services rapidly, in the multi hop case, the success rate drops sharply. We need to improve it by configuring the retransmit mechanism and discovery interval.

C. Messaging overhead

Fig. 7 shows the bandwidth usage on the egress interface at each MR. In our evaluation, we do not take into account ITS station internal communications between host and MR since they are performed with Ethernet which provides enough bandwidth and stability. In the evaluation, the size of each packet is as follows: (i) GeoBroadcast: from 189 to 264 bytes, (ii) GeoUnicast: 199 bytes, (iii) LS: from 86 to 94 bytes, and (iv) Beacon: 78bytes. The size of the GeoBroadcast packet, which contains a *SrvRqst* message, is variable because of the SLP's retransmission algorithm.

The total size of transmitted packets during the evaluation is 525Kbytes. Considering the evaluation is performed during 720 seconds, our proposed mechanism shows it does not consume much high bandwidth.

V. CONCLUSION AND FUTURE WORK

In this paper, we presented a location-aware service discovery mechanism for ITS applications in VANETs. The proposed mechanism is a harmonization of SLPv2 and IPv6 GeoNetworking developed in the GeoNet project. We showed that the IPv6 multicast-based service discovery using GeoBroadcast efficiently discovers services offered in the VANET without unnecessarily propagating IPv6 multicast packets into the entire VANET. We also implemented the proposed mechanism into Linux using the OpenSLP and the CarGeo6 implementations. The evaluation was performed in the field testbed in our campus.

As a next step we are conducting further field evaluations with more realistic scenarios. Although the proposed mechanism discovered services via not only single hop but also multiple hops with fairly low latency and low bandwidth usage, further improvements are necessary since the discovery success rate, specifically in the multi hop case, is not enough. It is also necessary to investigate how to discover services from/to the Internet in combination with IPv6 mobility support protocols.

ACKNOWLEDGMENT

The authors would like to thank the Tunisian developers from ESPRIT and all the people involved in the CarGeo6 implementation for their effort in the development and support of the GeoNetworking layer and for their collaboration with the IMARA project-team (INRIA).

REFERENCES

[1] P. Papadimitratos, A. La Fortelle, K. Evenssen, R. Brignolo, and S. Cosenza, "Vehicular communication systems: Enabling technologies, applications, and future outlook on intelligent transportation," *Communications Magazine, IEEE*, vol. 47, no. 11, pp. 84 –95, november 2009.

[2] CAR 2 CAR Communication Consortium, "CAR 2 CAR communication consortium manifesto, overview of the C2C-CC system, version 1.1," August 2007.

[3] ETSI, *ETSI TR 102 638 Technical Report, V1.1: Intelligent Transport Systems (ITS), Vehicular Communications (VC), Basic Set of Applications, Definitions*, June 2009.

[4] G. Karagiannis, R. Wakikawa, J. Kenney, C. J. Bernardos, and F. Kargl, "Traffic safety applications requirements," Internet Engineering Task Force, Internet-Draft draft-karagiannis-traffic-safety-requirements-02.txt, February 2010, (Work in progress).

[5] E. Guttman, C. Perkins, J. Veizades, and M. Day, "Service Location Protocol, Version 2," RFC 2608 (Proposed Standard), Internet Engineering Task Force, Jun. 1999, updated by RFC 3224. [Online]. Available: http://www.ietf.org/rfc/rfc2608.txt

[6] S. Cheshire and M. Krochmal, "Multicast dns," Internet Engineering Task Force, Internet-Draft draft-cheshire-dnsext-multicastdns-14, February 2011, (Work in progress).

[7] S. Cheshire and M. Krochmal, "Dns-based service discovery," Internet Engineering Task Force, Internet-Draft draft-cheshire-dnsext-dns-sd-10, Feburuary 2011, (Work in progress).

[8] E. Guttman, "Service Location Protocol Modifications for IPv6," RFC 3111 (Proposed Standard), Internet Engineering Task Force, May 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3111.txt

[9] ETSI, *ETSI EN 302 665 V1.1.1: Intelligent Transport Systems (ITS), Communications Architecture*, September 2010.

[10] ISO/IEC, *ISO21217:2010:Intelligent Transport Systems - Communications access for land mobiles (CALM) - Architecture*. International Organization for Standardization, April 2010.

[11] A. N. Mian, R. Baldoni, and R. Beraldi, "A survey of service discovery protocols in multihop mobile ad hoc networks," *Pervasive Computing, IEEE*, vol. 8, no. 1, pp. 66 –74, Jan.-March 2009.

[12] GeoNet Project, *D1.2 Final GeoNet Architecture Design*, June 2010.

[13] M. Tsukada, I. B. Jemaa, H. Menouar, W. Zhang, M. Goleva, and T. Ernst, "Experimental evaluation for IPv6 over VANET geographic routing," in *IWCMC '10: Proceedings of the 6th International Wireless Communications and Mobile Computing Conference*, 2010, pp. 736–741.

[14] S. Noguchi, M. Tsukada, I. B. Jemaa, and T. Ernst, "Real-vehicle integration of driver support application with IPv6 GeoNetworking," in *IEEE 73rd Vehicular Technology Conference (VTC 2011-Spring)*, May 2011, pp. 1–5.

[15] GeoNet Project, *D2.2 Final GeoNet Specification*, Jan 2010.