GeoSAC - Scalable Address Autoconfiguration for VANET Using Geographic Networking Concepts

Roberto Baldessari[1], Carlos J. Bernardos[2], Maria Calderon[2]

[1]NEC Europe Ltd., Network Laboratories, *baldessari@nw.neclab.eu*
[2]Universidad Carlos III de Madrid, *cjbc@it.uc3m.es*, *maria@it.uc3m.es*

(Invited Paper)

***Abstract*—In this paper, we propose a novel mechanism for application of an IPv6 automatic address configuration technique to vehicular ad-hoc networks. The solution consists of combining standardized IPv6 schemes with geographic routing functionalities, which enables the matching of geographically-scoped network partitions to single IPv6 multicast-capable links. Unlike existing solutions described in this paper and mostly derived from MANET approaches, our proposal explicitly targets automotive requirements, which we identify and analyze based on a real system architecture and its target applications. Furthermore, we examine the solution's timely aspects both analytically and experimentally with laboratory tests and compare the results. Finally, we outline intended future work on extending the proposed scheme.**

I. Introduction

Worldwide interest is currently given to vehicular ad-hoc networks (VANETs) as a promising technology used in increasing road safety and driving comfort. Although many applications of vehicular communications were already identified in the 80s, large-scale deployment of such systems has finally become possible due to the availability of new technologies, such as devices based on the IEEE 802.11 standard family, which seem to offer an affordable compromise between performance and system complexity.

The primary advantage to deploying this kind of self-organized network is the fact that timely critical applications, such as safety-of-life applications, can be implemented by letting vehicles directly communicate to each other, instead of relying on a centralized entity. Intersection collision avoidance, emergency brake notification and post-crash warning are only a few of the envisaged applications based on this technology. Additionally, by connecting the VANET to an infrastructure network, other less critical safety applications and infotainment services can be implemented. For example, a dangerous situation can be reported to a service center and then notified to a much wider area, while centralized traffic management can also benefit extremely from this technology.

With respect to infrastructure-based applications, the deployment effort required to equip most strategic road segments with access points connected to a network infrastructure is often regarded as a major obstacle. A considerable aid in reducing this difficulty could come from relying on the Internet's existing and widespread core network infrastructure, instead of deploying a dedicated network. Connecting a VANET to the Internet enables, on the one hand, vehicles to rely on protocols that have constantly been enhanced and have proved effective on a global scale. On the other hand, in order to become part of this interconnected network, the VANET is required to comply with those standard protocols and mechanisms that have allowed the Internet to interconnect heterogeneous networks. At the current VANET state of the art, this conformance is not commonly achieved, as many research activities regard a VANET as a generic spontaneous network. In particular, one aspect that strongly affects a VANET capability to correctly connect to the Internet is the IPv6 address configuration scheme and how the IPv6 requirement for link-local multicast support [1] is achieved. The lack of a standard solution for generic ad-hoc networks[1] is reflected and amplified in VANET, where the potentially huge number of nodes and their high mobility represent further challenges.

The Internet Protocol (IP) version 4 has been the robust spine on which the Internet has been able to reach its global deployment. The new version, IPv6, went through a long design phase which resulted in its excellent versatility but partially also caused its still limited application. Emerging technologies like VANETs can benefit from IPv6 functionalities and are therefore expected to generate new momentum for IPv6 deployment. In particular, all of the international committees defining architectures for vehicular communication have included a native IPv6 stack in their protocol stacks, namely IEEE 1609 [2], ISO TC 204 (CALM) [3], the Car2Car Communication Consortium (C2C-CC) [4] and the newly formed ETSI TC ITS [5]. Some of these bodies already target multi-hop IPv6 communication with an infrastructure network, whereas others might consider it in a later stage. Thus, the need for solutions for usage of IPv6 in self-organized networks with distributed and mobile relays is expected to grow.

In this paper, we focus on the address configuration problem taking into account both automotive requirements and Internet integration issues, with particular focus on scalability and deployability. The main contributions of this paper are the identification of requirements based on a real system architecture (Section II), an extensive literature analysis (Section III), the proposal of a basic solution scheme and possible extensions (Section IV), an analysis of the solution with respect to the identified requirements and in terms of timely performance (Section V) and, finally, an experimental evaluation of the solution (Section VI).

[1] To date, the IETF AUTOCONF work group has not finalized the problem statement and is not expected to provide a standard solution in the short term.

II. System Architecture and Requirements

In this paper we assume a system based on short-range communication technologies. In particular, the IEEE 802.11 standard family has recently attracted the automotive industry's interest as it seems to offer the best compromise between costs/complexity and performance. Further, the European spectrum allocation authority is to assign a protected frequency band around 5.9 GHz for safety and non-safety ITS purposes [6]. This frequency band can be effectively exploited by adopting the IEEE 802.11p draft standard [7], which defines specific amendments to 802.11 for vehicular applications.

Among the aforementioned institutions that are including IPv6 in their protocol stacks, IEEE and ISO envisage a single-hop, infrastructure-based approach, where IPv6 applications are possible only when a vehicle is in the direct communication range of a point-of-attachment. C2C-CC and ETSI TC ITS, instead, also target multi-hop communications for both safety and non-safety, with the goal of extending the communication range of an access point by using other vehicles as relays. This extended scenario multiplies the number of possible applications of this technology, but also poses challenges to the deployment of IPv6 with respect to routing, addressing, mobility, and security/privacy. Address configuration is particularly affected by the multi-hop nature of the network and standardized solutions do not exist neither for VANET nor for more general MANET. The major issues in this respect are pointed out after introducing the C2C-CC system architecture which is the reference for this paper.

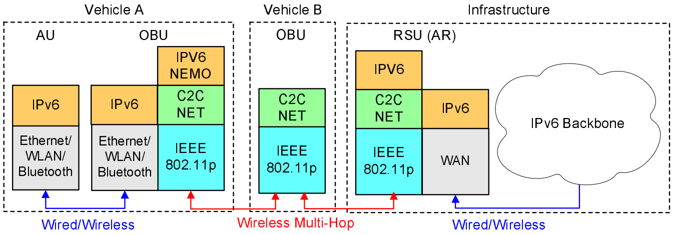

Fig. 1. IPv6 Deployment in the C2C-CC System Architecture

In the C2C-CC system architecture, vehicles are equipped with devices termed *On-Board Units* (OBU), which implement the C2C-CC protocol stack [4]. Units of different cars can communicate with each other or with fixed stations installed along roads termed *Road Side Units* (RSU). OBUs and RSUs implement the same network layer functionalities and form a self-organizing network. RSUs can be connected to a network infrastructure, most presumably an IP-based network. Also, it is reasonable to assume that RSUs will act as IPv6 Access Routers (AR) or as network bridges connected to an AR. Passenger or driver devices attached to the vehicle on-board system are called *Application Units* (AU). AUs are assumed to have a standard IPv6 protocol stack, OBUs act as gateways for the in-vehicle network optionally enhanced with the Network Mobility Basic Support protocol [8]. Figure 1 depicts the resulting set of communicating devices and their protocols with respect to IPv6.

In this system architecture based on short range communication devices, the system capability of supporting Internet-based applications over multi-hop communication strongly depends on mobility. Single-hop vehicular Internet access based on WLAN has already been investigated in highway scenarios [9], concluding that the link between OBU and RSU is stable enough to allow for several types of applications. When considering multi-hop communication, the scope of Internet-based applications is realistically reduced to lower speed scenarios (e.g. urban or semi-urban), to a proper ratio of OBUs per installed RSU and to a realistic maximum number of hops to be determined. Even within this restricted scope of applicability, the quality of address autoconfiguration procedures can determine the effectiveness of multi-hop support in use cases identified by the automotive industry [10].

Based on the C2C-CC reference system and the objective of providing Internet-based applications in the described scenario, we identified the following key requirements for IPv6 address configuration functionalities.

1) The address autoconfiguration technique must provide the capability to configure **globally valid addresses**. In fact, vehicles must be able to receive unsolicited packets originated in the infrastructure network.
2) The address autoconfiguration technique must present a **low complexity** in terms of required operations to the vehicles. This implies that the technique should require (i) the smallest possible amount of changes to already standardized mechanisms, (ii) that as little state information as possible is maintained by a node and (iii) that the technique works for both single-hop and multi-hop access with the smallest possible differences.
3) A minimum amount of **signaling** messages should be used, in order to save physical channel resources.
4) The address autoconfiguration technique must allow for usage of Network Mobility Basic Support. This implies that the technique must be suitable for **Movement Detection** [11] procedures.
5) In the case that multiple RSUs are simultaneously reachable, **gateway selection** must be provided by the address autoconfiguration technique. It is a desired functionality that the gateway selection algorithm is under the control of the infrastructure operator (e.g. road operator), which allows for enforcement of network management operations.
6) For network security reasons, the address autoconfiguration technique must not require selected nodes to carry out critical tasks on which the whole VANET operation depends. In other words, the technique must be executed in a **fully distributed** fashion.
7) For network security reasons, the technique must provide **authentication and integrity** of signalling messages.
8) The address autoconfiguration technique must protect the **privacy** of vehicles' users. This implies that the technique must not reveal information that could potentially be used to track vehicles or to link vehicles' network identifiers with real drivers' identity.

III. Related Work

The multi-hop nature of VANET and its lack of a single multicast-capable link for signalling prevent current IP address autoconfiguration related protocol specifications to be used as-is. The same problem occurs in general in any unmanaged multi-hop network. Among these, Mobile Ad-hoc Networks (MANETs) have received a remarkable attention in the research area for years. Even before the creation of the IETF AUTOCONF WG in 2005, a plethora of solutions existed for MANETs [12]. These can be classified according to many different criteria. For the purpose of applying them to the target scenario, we classify existing solutions in those for Standalone MANETs (not connected to any external network) and Connected MANETs (connected to an external infrastructure in a permanent or intermittent fashion, by means of one or more gateways). Only the latter category is considered here since Internet connectivity is required as described in the previous section.[2] In the followings, we summarize the operation of the most significant proposed approaches and explain why they do not fulfill the requirements of Section II.

Ruffino et al. [13] describe a mechanism bound to a particular ad-hoc routing protocol, OLSR, that enables nodes belonging to a Connected MANET – by means of one or more gateways – to obtain global IPv6 addresses. At bootstrap, a node configures a Primary Address (PADD) that is MANET-scoped and is used as main address to exchange OLSR messages. Each of the gateways available in the MANET has a global IPv6 prefix that is announced using a new OLSR message type, called Prefix Advertisement (PA). With the prefix information received in the PA messages, a node is able to build a set of global IPv6 addresses (called Secondary Addresses: SADDs). Among them, the node chooses the "best" prefix and starts using the address formed from this prefix (called, Designated Secondary Address: DSADD). The node introduces all (or a subset) of the SADDs (including the DSADD) in OLSR messages and starts broadcasting them, enabling these addresses to be routable and reachable within the MANET. This solution is defined for a particular routing protocol and it assumes that a MANET local IPv6 address is already configured (at the bootstrapping phase) before obtaining a global IPv6 address. However, the uniqueness of both the local and global obtained addresses should be ensured by means of a Non-unique Address Detection method, which is not covered by the solution.

Templin et al. [14] propose a mechanism based on DHCPv6 [15]. The solution connects nodes within a MANET by means of virtual Ethernet links, which are imaginary shared links that connect the MANET nodes. Nodes attach to the virtual Ethernet via an interface configured over underlying MANET interface(s). Using this virtual Ethernet, MANET nodes can configure global IPv6 addresses using DHCPv6. Two different types of "virtual Ethernets" are defined: an "enhanced" view of this virtual Ethernet sees the MANET as a fully-connected shared link that connects all MANET nodes (each node encapsulates each IP packet in an outer IP header and then sends it on an underlying MANET interface), and an "unenhanced" view sees the MANET as a multilink site (nodes send each IP packet on an underlying MANET interface without further encapsulation). The main advantages of this solution are that it re-uses a well known protocol (DHCPv6), enables prefix delegation support and ensures the uniqueness of the delegated addresses/prefixes. However, it requires certain state, it is not fully distributed and it is not easily suitable for movement detection procedures.

Fazio et al. [16] propose a solution – called Vehicular Address Autoconfiguration (VAC) – particularly designed for VANET environments, exploiting the VANETs topology and an enhanced DHCP service with dynamically elected leaders to provide a fast and reliable IP address configuration. VAC organizes leaders in a connected chain such that every node (vehicle) lies in the communication range of at least one leader. This hierarchical organization allows for limiting the signal overhead for the address management tasks. Only leaders communicate with each others to maintain updated information on configured addresses in the network. Leaders act as servers of a distributed DHCP protocol and normal nodes ask leaders for a valid IP address whenever they need to be configured. The main drawbacks of this solution are the assumption of linear topology and group movement which limits the applicability scope, the overhead due to the explicit management signaling (e.g. between leaders) and the possible security threat due to the critical tasks carried out by the leaders.

IV. GeoSAC

The solution proposed here, called GeoSAC (Geographically Scoped stateless Address Configuration), consists of adapting the existing IPv6 Stateless Address Autoconfiguration (SLAAC) mechanisms to geographic addressing and networking, where the concept of IPv6 *link* is extended to a specific geographic area associated with a point-of-attachment. In the followings we refer to the protocol architecture introduced in Section II, where the basic mechanisms of our solution are most effective.

In the considered protocol architecture (Figure 1), a sub-IP layer (C2C NET) deals with ad-hoc routing by applying geographic networking and presents to the IPv6 layer a flat network topology. Consequently, the *link* seen by the IPv6 layer includes nodes that are not directly reachable but are portrayed as such by the sub-IP layer. The IPv6 *broadcast domain* is managed by the sub-IP layer and can be configured statically or on a per-packet basis according to geographic parameters, instead of pure topological ones mostly used in MANET such as hop limit. As a result, the sub-IP presents to IPv6 a multicast link which includes a partition of the VANET made by all nodes within a certain geographical area.

Relying on the multi-hop distribution and network partitioning offered by the sub-IP geographic routing, in the proposed solution a point-of-attachment sends out standard IPv6 Router Advertisement (RA) messages which reach all

[2] It should be noted that some of the solutions defined for standalone MANETs could be extended to operate as well in connected environments, although analyzing that is out of the scope of this paper.

and only the nodes currently located within a well-defined area. In particular, the attachment point specifies as target of the sub-IP protocol header a pre-assigned geographical area which is served by this gateway. Upon reception of this packet, a node applies the geographic filtering before delivering the RA to the IPv6 layer and forwards the message according to the *geocast* forwarding procedure. As a result, if a multi-hop path exists, all the nodes within the area receive the RA and the IPv6 instance running above geo-networking processes the message as if the node was directly connected to the access router that issued the message.

According to IPv6 SLAAC, at this point a host generates an address appending its network identifier derived from the MAC address to the received IPv6 prefix and performs the IPv6 *Duplicate Address Detection* (DAD) procedure. For this purpose, we propose that the same geographic area specified by the RSU is again set as broadcast domain, which allows for uniqueness of the addresses within this area. Assuming that the IPv6 prefix announced by the RSU is exclusively assigned to this area, the address uniqueness is verified[3].

Regarding detection of duplicate addresses, we argue that the execution of DAD might be unnecessary because MAC addresses in VANET might be required to be unique, at least within macro-regions where vehicles are sold and can potentially communicate with each other (e.g. a continent). This property in fact is highly desirable for security and liability reasons, as it would allow (i) forensic teams to rely on vehicular communications to reconstruct accident scenes or other critical situations and (ii) to detect malicious nodes and considerably reduce effects of network attacks. Despite uniqueness of addresses, privacy of users can be protected by equipping vehicles with sets of unique MAC addresses to be used for limited intervals as *pseudonyms* [17]. These addresses could be assigned by authorities and, when coupled with the usage of digital certificates and cryptographic protection [18], this mechanism can accomplish support for liability as well as privacy protection.

A technique that maximizes the benefits of GeoSAC consists in shaping the geographical areas assigned to the RSU in an adjacent and non-overlapping fashion, as depicted in Figure 2. By doing so, the following key advantages are obtained: (i) **univoque gateway selection** is achieved with the infrastructure having full control on it[4], as only one RSU is assigned per geographical area; (ii) a **network partitioning** is obtained that supports Movement Detection procedures of IPv6 mobility and also allows for location-based services. In particular, a vehicle moving across regions served by different RSUs (case 2 in Figure 2) experiences a sharp sub-net change, without traversing *gray areas* where Router Advertisements are received from multiple access points.

In addition to the already mentioned benefits, the VANET partitioning obtained with GeoSAC enables a matching between geographical area and IPv6 prefix assigned to an access router. For the purpose of deploying location-based applications, this matching can be coupled with already proposed techniques for geographical routing in the infrastructure, like extended DNS [19] or geographic IPv6 prefix format [20]. In the first approach, DNS servers are extended with the capability to resolve geographical locations into IP addresses, without requiring changes in the routing behavior of today's Internet. The second approach consists in encoding the geographic position as part of the IPv6 prefix and performing the actual routing accordingly. With respect to these techniques, our solution provides the ad hoc network partitioning that is required for VANET in order to achieve fine resolution in infrastructure-to-vehicle applications, e.g. delivery of information with localized scope, like traffic and weather updates, point-of-interests notifications etc. An example usage of the proposed solution coupled with extended DNS is depicted in Figure 3, where we also suggest as option to utilize modified unicast-prefix-based IPv6 multicast addresses [21] for interdomain multicast routing[5].

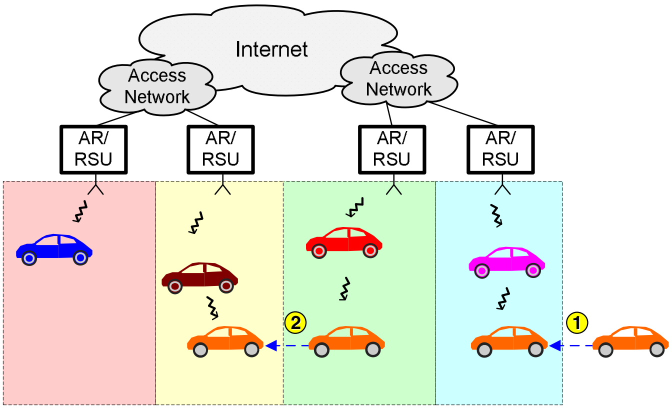

Fig. 2. VANET Partitioning Achieved with the Proposed Solution

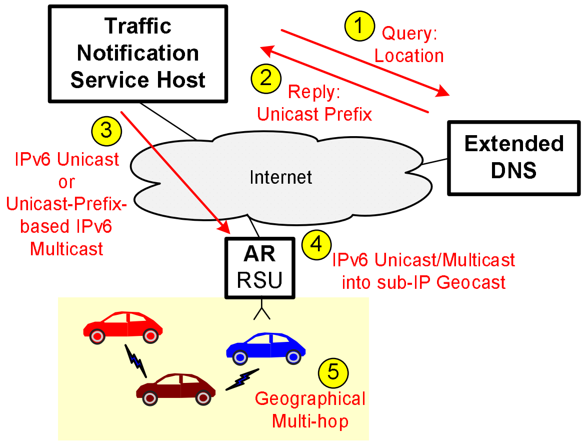
Traffic Notification Service Host
1 Query: Location
2 Reply: Unicast Prefix
Extended DNS
3 IPv6 Unicast or Unicast-Prefix-based IPv6 Multicast
Internet
AR RSU
4 IPv6 Unicast/Multicast into sub-IP Geocast
5 Geographical Multi-hop

Fig. 3. Proposed Solution Coupled with Extended DNS (Usage Example)

[3] The proposed solution could be applied to multiple RSUs acting as bridges connected to one single Access Router. In this case the DAD messages should be forwarded among the RSUs to assure uniqueness in the entire IP subnet.

[4] More precisely, in GeoSAC gateway selection is performed by the infrastructure itself and not by the nodes as in many MANET approaches.

[5] In this example, routing between source and RSU is unicast-like and the RSU maps IPv6 multicast groups into sub-IP multicast mechanisms such as geobroadcast, geoanycast or simple flooding.

V. Analysis of the Solution

This section provides an analysis of the proposed solution. It is organized in two different parts. First, GeoSAC is evaluated against each of the requirements listed in Section II Next, we analytically characterize the time required by GeoSAC to configure a new address assuming an ideal physical layer. This theoretical result is then compared to measurements results in SectionVI.

1) **Global valid address configuration.** GeoSAC fulfills this requirement, since a globally valid prefix can be included into the RA broadcast within each geographical area. Our solution is an extension of standard IPv6 Stateless Address Autoconfiguration to support geographically-scoped multi-hop domains, instead of the standard multicast-capable one-hop link assumed by classic IPv6 mechanisms.
2) **Low complexity.** The solution has low complexity in terms of per-node requested functionality. Each node only needs to perform the geographically-scoped broadcast filtering and forwarding of RA messages and to process them in the usual way [22]).
3) **Low signalling.** Due to the use of geographical routing, RA messages can reach all nodes within a given geographical area. These geographically distributed RAs are the only signalling messages required by the solution in order to work (we assume DAD is not required as explained in Section IV). However, enhancements to GeoSAC to reduce the signalling overhead (e.g. caching of RA messages) are theoretically possible and are subject of current research.
4) **NEMO-capable.** Compatibility with the NEMO Basic Support protocol, and in general with IP mobility mechanisms is guaranteed, since the solution provides link-local multicast support required by the Neighbor Discovery protocol, which ensures that IPv6 movement detection mechanisms work without any modification. For an example of usage of NEMO in VANET that relies on the present solution see [23].
5) **Gateway selection.** The geographic VANET partitioning obtained with GeoSAC provides infrastructure-based gateway selection when more RSUs are physically reachable. Further, if more RSUs are assigned to the same area, existing mechanisms like default policy table management for address selection [24] or RA extensions for router preferences [25] are supported by GeoSAC.
6) **Distributed approach.** The proposed mechanism does not rely on any particular node on the VANET playing a special role in the IP address autoconfiguration process, but only on some nodes located on the infrastructure side (i.e. the RSUs, or Access Routers attached to them).
7) **Authentication and Integrity.** GeoSAC assumes that the sub-IP layer provides these security functionalities. As an example, the scheme proposed in [18] providing both end-to-end and hop-by-hop authentication and integrity could be used for RA messages. An alternative approach, still compatible with our solution, consists of adapting Secure Neighbor Discovery (SEND) [26]. The application of Cryptographically Generated Addresses (CGA) in automotive applications is subject of ongoing research.
8) **Privacy protection.** The mechanism itself does not either protect nor compromise users' privacy. The present solution is compatible with the usage of pseudonymous MAC addresses, which has been proposed [17] with the aim of achieving better privacy protection.

Based on the above analysis, we can conclude that GeoSAC fulfills the requirements identified for a specific VANET solution. However, in order to provide a quantitative indication, in the followings we derive an analytical expression of the time required by GeoSAC to configure an address.

The address configuration time (T_{conf}) is the time elapsed since a vehicle entered a new geographical area (therefore loosing the connectivity to the old RSU) till the moment in which it can start using the newly configured global IPv6 address. This time depends on several factors, such as the shape and size of the areas, the configuration of the RSUs, etc. In this analysis we focus on the case where the RSUs of the same geographical area announce the same IPv6 prefix in periodic unsolicited RAs, being the interval between RAs a random variable uniformly distributed between a minimum value (*MinRtrAdvInterval*) and a maximum value (*MaxRtrAdvInterval*), which we refer to as $T_{RA_{min}}$ and $T_{RA_{max}}$ respectively. In such a scenario, the address configuration time is given by

$$T_{conf} = T_{RA} + T_{relay} \tag{1}$$

where T_{RA} is the time elapsed since a vehicle entered a new geographical area to the moment an RSU/AR in the new area sends an unsolicited RA message, and T_{relay} is the time required by such RA message to reach the vehicle since it was issued by the RSU/AR. T_{relay} depends, among other factors, on the number of hops between the RSU/AR and the targeted vehicle. We assume that the cars density is such that a vehicle always has connectivity in every point of the area.[6] An expression for the average of T_{RA} can be found in [27]:

$$\bar{T}_{RA} = \frac{T^2_{RA_{max}} + T_{RA_{max}}T_{RA_{min}} + T^2_{RA_{min}}}{3(T_{RA_{max}} + T_{RA_{min}})} \tag{2}$$

From equations (1) and (2) it is easy to obtain the average time required by a node to configure an IPv6 address every time it changes area:

$$\begin{aligned} \bar{T}_{conf} &= \bar{T}_{RA} + \bar{T}_{relay} = \qquad (3) \\ &= \frac{T^2_{RA_{max}} + T_{RA_{max}}T_{RA_{min}} + T^2_{RA_{min}}}{3(T_{RA_{max}} + T_{RA_{min}})} + \bar{T}_{relay} \end{aligned}$$

[6] Obviously, if a multi-hop path does not exist, global address configuration can not be provided. Enhancements including RAs caching are out of scope of this paper.

To conclude the analysis, we provide an example comparison between $\bar{T}_{conf}$ and the estimated permanency time of a vehicle within a geographical area. Assuming a rectangular area with a length of 1000m (i.e. twice as big as the expected physical communication range) and an average speed of 45 km/h, a vehicle spends 80 seconds in the area. By choosing values of $T_{RA_{max}}$ smaller than 5 seconds, GeoSAC guarantees that vehicles can run Internet-based applications for more than 70 seconds. However, it is important to note that GeoSAC parameters like size and shape of geographic areas should be chosen also taking into according the expected density of vehicles.[7]

VI. Implementation and Evaluation

GeoSAC has been entirely implemented as a software prototype for the Linux operating system. A software module integrated into the Linux operating system provides geographic routing and forwarding of packets using a new protocol header, in which a target area can be specified. Additionally, the module creates a virtual network interface that is seen by the Linux IPv6 layer as a normal interface. On the sender side, IPv6 packets given to the virtual interface are encapsulated into geographic protocol headers and sent over a real WLAN interface. On the receiver side, the geographic routing modules receives the packet, decapsulates the inner IPv6 header and re-injects it into the Linux kernel stack, so that the IPv6 layer catches and processes it.

In the followings, experimental measurements conducted with the aforementioned implementation are described. The goals of these tests are (i) to validate the proposed solution by means of a real and deployable prototype and (ii) to validate the analysis of the address configuration time presented in Section V. The scenario we emulated in the measurements[8] is illustrated in Figure 4. In this scenario, RSU1 and RSU2 issue Router Advertisement messages that are distributed to rectangular areas covering a urban road. Car A is initially inside the area managed by RSU2 and moves with a constant speed towards RSU1. For simplicity, in these tests the other vehicles (Car B, C, D and E) do not move. After entering the target area, Car A is able to configure an address after receiving an RA that is relayed hop-by-hop from the RSU to the neighboring Car B. The vehicles' topology, indeed, is chosen such that Car A can receive the RA message only from Car B, as the other cars are out of range.

The scenario shown in Figure 4 was reproduced in a single laboratory room using both commercial carPCs based on general-purpose CPUs as well as NEC embedded systems for automotive OBU based on MIPS CPU architecture. As physical and MAC layers we used Atheros-based IEEE 802.11a commercial hardware with the Madwifi driver [28] in pseudo ad hoc mode.[9] In order to emulate the physical communication range we adopted filtering of incoming packets based on the source node's position. In practice, packets sent by a node located beyond a predefined distance are filtered out at sub-IP layer. Movement of Car A is emulated by periodically feeding the geographical routing module with predefined positions, instead of using a real GPS receiver. Further testbed parameters are listed in Table I.

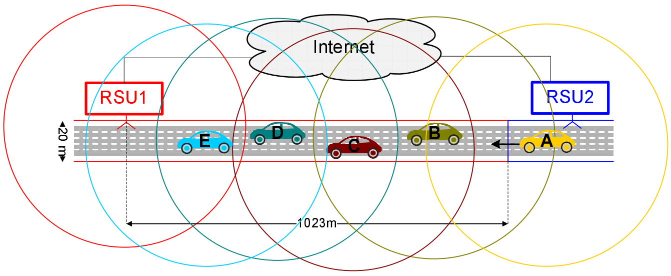

Fig. 4. Test Scenario

MAC Layer	Atheros AR5212 802.11a, pseudo ad-hoc mode
PHY Layer	5.8 GHz, 20 MHz channel, 10 dBm tx power, 6 Mb/s
Emulated Communication Range	250 m
Emulated Speed	45 km/h
Total size of RA Frames	200 Bytes

TABLE I
Tests Parameters

The conducted measurements consist of two parts: in the first, after placing Car A within the area served by RSU1, we measured the 1-way delay between RSU1 and CarA for geocasted IPv6 packets ($\bar{T}^{meas}_{relay}$). In the second part we measured the address configuration time ($\bar{T}^{meas}_{conf}$) for different values of interval between RAs. Since both time intervals strongly depend on the channel conditions (network load, packet losses and consequent retransmissions etc.), we replicated the same conditions in both experiments and we verified that the channel presented optimal propagation characteristics and low utilization, such that collisions did not occur. For the time measurements we used time inspections in the Linux kernel and frequent nodes synchronization via NTP protocol. The clock jitter between two NTP synchronizations was considered negligible by looking at the NTP logs, which showed time offsets with order of magnitude 10^{-6} s.[10] The measured 1-way delay for the scenario of Figure 4 is $\bar{T}^{meas}_{relay} = 7.026$ ms. Results for the address configuration time are summarized in Table II, where the analytical value $\bar{T}_{conf}$ obtained with Equation 3 and the measured $\bar{T}^{meas}_{relay}$ is compared with the experimental value $\bar{T}^{meas}_{conf}$.

The experimental results presented in Table II show how with an optimal communication medium and low channel

[7] For example, in sparse scenarios the area should be bigger than the physical range but in dense scenarios the opposite case is more beneficial.

[8] The scenario is intentionally simplified, as the goal of the measurements is to separate the configuration time due to protocol design from effects of due to bad channel propagation or channel congestion. To alleviate these effects, well known multi-hop broadcast enhancements can be applied to GeoSAC like contention-based suppression or caching and re-broadcasting.

[9] In this mode, no management frames are used, so all nodes share the same BSSID and no association procedure takes place.

[10] Consequently, results are deemed accurate at least in the order of milliseconds.

utilization the measured time required for a node to configure an address basically corresponds to the value foreseen with mathematical analysis.

$T_{RA_{min}}$ (s)	$T_{RA_{max}}$ (s)	$\bar{T}_{conf}$ (s)	$\bar{T}^{meas}_{conf}$ (s)
0.3	0.4	0.183	0.178
0.4	0.6	0.260	0.258
0.8	1.2	0.514	0.511
1.6	2.4	1.020	1.017
4	6	2.540	2.535

TABLE II
EXPERIMENTAL RESULTS

VII. CONCLUSIONS AND FUTURE WORK

In this paper we have tackled the problem of automatic address configuration in VANETs. Unlike many existing approaches, we have addressed both specific issues of vehicular networks as well as those of Internet integration. We have presented a solution that reuses standard mechanisms that have proved effective, like IPv6 Stateless Autoconfiguration, and also combines them with specialized distribution techniques for vehicular networks, like geographic-based routing. We have analytically evaluated the solution against the identified requirements and experimentally tested it with a real software prototype.

We would like to conclude this paper by pointing out that several aspects of the proposed solution deserve further attention. First, we adopted perfectly non-overlapping geographical areas as the target of the RA messages. Aside from the feasibility of this approach with the limited accuracy of currently available positioning systems, it is worth considering the use of overlapping areas to allow for more sophisticated handover mechanisms like *make-before-break* techniques, where a vehicle starts configuring a new IPv6 address while the old address is still valid and usable. This type of approach could, in fact, largely benefit from a position-aware sub-IP layer. For example, vehicles could be aware of the fact that they were leaving the area served by an RSU. Furthermore, vehicles could know the area served by current and next RSUs. This information is of much help in terminal-based make-before-break handover procedures, which usually only rely on indications coming from the device such as the received signal strength.

An additional item of research briefly introduced in Section IV is the design of extended DNS solutions providing resolution of geographical locations into IP addresses. A third interesting open issue is the definition of new and specific multicast groups benefiting from the availability of geographical information/knowledge. For example, sub-groups of vehicles could be formed and addressed, taking into account vehicles' type and purpose (e.g. emergency vehicles) or other characteristics (speed, direction etc.). A relatively wide area of research opens if we consider that new multicasting techniques offered by a sub-IP layer could be used by the Internet Protocol, allowing nodes in the infrastructure to address vehicles in many different and sophisticated ways.

REFERENCES

[1] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 2460 (Draft Standard), Dec. 1998.

[2] C. S. of the IEEE Intelligent Transportation Systems Council, "Draft Standard for Wireless Access in Vehicular Environments (WAVE) – Networking Services," IEEE P1609.3/D18, December 2005.

[3] "ISO TC204 WG16," http://www.tc204wg16.de.

[4] Car-to-Car Communication Consortium, "C2C-CC Manifesto," Version 1.1, August 2007, available at http://www.car-to-car.org/fileadmin/dokumente/pdf/C2C-CC_manifesto_2007_09_24_v1.1.pdf.

[5] "ETSI Techical Committee ITS," http://www.etsi.org/.

[6] L. Le, W. Zhang, A. Festag, and R. Baldessari, "Analysis of Approaches for Channel Allocation in C2X Communication," in *1st International Workshop on Interoperable Vehicles (IOV)*, Zurich, Switzerland, March 2008, pp. 33–38.

[7] IEEE, "IEEE P802.11p/D3.0 - Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 7: Wireless Access in Vehicular Environments," July 2007.

[8] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert, "Network Mobility (NEMO) Basic Support Protocol," RFC 3963 (Proposed Standard), Jan. 2005.

[9] J. Ott and D. Kutscher, "The Drive-Thru Architecture: WLAN-based Internet Access on the Road," in *Proc. VTC Fall*, May 2004.

[10] R. Baldessari, T. Ernst, A. Festag, and M. Lenardi, "Automotive Industry Requirements for NEMO Route Optimization," Internet Draft, draft-ietf-mext-nemo-ro-automotive-req-00, work in progress, February 2008.

[11] D. Johnson, C. Perkins, and J. Arkko, "Mobility Support in IPv6," RFC 3775 (Proposed Standard), June 2004.

[12] C. J. Bernardos, M. Calderon, and H. Moustafa, "Survey of IP address auto-configuration mechanisms for MANETs," IETF, draft-bernardos-manetautoconf-survey-03.txt (work-in-progress), April 2008.

[13] S. Ruffino and P. Stupar, "Automatic configuration of IPv6 addresses for MANET with multiple gateways (AMG)," IETF, draft-ruffino-manet-autoconf-multigw-03.txt (work-in-progress), June 2006.

[14] F. Templin, S. Russert, , and S. Yi, "MANET Autoconfiguration," IETF, draft-templin-autoconf-dhcp-14.txt (work-in-progress), April 2008.

[15] R. Droms, J. Bound, B. Volz, T. Lemon, C. Perkins, and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)," RFC 3315 (Proposed Standard), July 2003.

[16] M. Fazio, C. Palazzi, S. Das, and M. Gerla, "Vehicular Address Configuration," in *Proc. of the 1st IEEE Workshop on Automotive Networking and Applications (AutoNet), GLOBECOM*, 2006.

[17] E. Fonseca, A. Festag, R. Baldessari, and R. Aguiar, "Support of Anonymity in VANETs – Putting Pseudonymity into Practice," in *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC)*, Hong Kong, March 2007.

[18] C. Harsch, A. Festag, and P. Papadimitratos, "Secure Position-Based Routing for VANETs," in *VTC Fall*, Baltimore, MD, USA, October 2007, 5 pages.

[19] T. Imielinski and J. Navas, "GPS-Based Addressing and Routing," RFC 2009 (Experimental), Nov. 1996.

[20] T. Hain, "An IPv6 Provider-Independent Global Unicast Address Format," August 2006.

[21] P. Savola and B. Haberman, "Embedding the Rendezvous Point (RP) Address in an IPv6 Multicast Address," RFC 3956 (Proposed Standard), Nov. 2004.

[22] S. Thomson, T. Narten, and T. Jinmei, "IPv6 Stateless Address Autoconfiguration," RFC 4862 (Draft Stndard), Sept. 2007.

[23] R. Baldessari, A. Festag, W. Zhang, and L. Le, "A MANET-centric Solution for the Application of NEMO in VANET Using Geographic Routing," in *Proc of TridentCom*, Innsbruck, Austria, March 2008, 7 pages.

[24] R. Draves, "Default Address Selection for Internet Protocol version 6 (IPv6)," RFC 3484 (Proposed Standard), Feb. 2003.

[25] R. Draves and D. Thaler, "Default Router Preferences and More-Specific Routes," RFC 4191 (Proposed Standard), Nov. 2005.

[26] J. Arkko, J. Kempf, B. Zill, and P. Nikander, "SEcure Neighbor Discovery (SEND)," RFC 3971 (Proposed Standard), Mar. 2005.

[27] Y.-H. Han, J. Choi, and S.-H. Hwang, "Reactive handover optimization in ipv6-based mobile networks," *Selected Areas in Communications, IEEE Journal on*, vol. 24, no. 9, pp. 1758–1772, Sept. 2006.

[28] "MADWIFI: Multiband Atheros Driver for WIFI," http://www.madwifi.org.