Coherent, automatic address resolution for vehicular ad hoc networks

Luis Urquiza-Aguiar*

Department of Network Engineering,
Universitat Politècnica de Catalunya (UPC),
C. Jordi Girona 1-3, Barcelona 08034, Spain
Email: luis.urquiza@entel.upc.edu
*Corresponding author

Carolina Tripp-Barba

Faculty of Informatics,
Universidad Autónoma de Sinaloa (UAS),
De los Deportes Avenue and Leonismo Internacional,
Mazatlán 82107, Mexico
Email: ctripp@uas.edu.mx

David Rebollo-Monedero, Ahmad Mohamad Mezher, Mónica Aguilar and Jordi Forné

Department of Network Engineering,
Universitat Politècnica de Catalunya (UPC),
C. Jordi Girona 1-3, Barcelona 08034, Spain
Email: david.rebollo@entel.upc.edu
Email: ahmad.mezher@entel.upc.edu
Email: monica.aguilar@entel.upc.edu
Email: jordi.forne@entel.upc.edu

José Estrada-Jiménez

Department of Electronics, Telecommunications and Networks,
Escuela Politécnica Nacional (EPN),
Ladrón de Guevara E11 253, Quito 170413, Ecuador
Email: jose.estrada@epn.edu.ec

Abstract: The interest in vehicular communications has increased notably. In this paper, the use of the address resolution (AR) procedures is studied for vehicular ad hoc networks (VANETs). We analyse the poor performance of AR transactions in such networks and we present a new proposal called coherent, automatic address resolution (CAAR). Our approach inhibits the use of AR transactions and instead increases the usefulness of routing signalling to automatically match the IP and MAC addresses. Through extensive simulations in realistic VANET scenarios using the Estinet simulator, we compare our proposal CAAR to classical AR and to another of our proposals that enhances AR for mobile wireless networks, called AR+. In addition, we present a performance evaluation of the behaviour of CAAR, AR and AR+ with unicast traffic of a reporting service for VANETs. Results show that CAAR outperforms the other two solutions in terms of packet losses and furthermore, it does not introduce additional overhead.

Keywords: VANETs; vehicular ad hoc networks; AR; address resolution; ARP; address resolution protocol; ND; neighbour discovery; routing protocols; wireless ad hoc networks.

Reference to this paper should be made as follows: Urquiza-Aguiar, L., Tripp-Barba, C., Rebollo-Monedero, D., Mezher, A.M., Aguilar, M., Forné, J. and Estrada-Jiménez, J. (2017) 'Coherent, automatic address resolution for vehicular ad hoc networks', *Int. J. Ad Hoc and Ubiquitous Computing*, Vol. 25, No. 3, pp.163–179.

Biographical notes: Luis Urquiza-Aguiar received the MS in Telematics Engineering from the Universitat Politècnica de Catalunya (UPC) in 2012. He is a PhD candidate in Network Engineering

at the UPC in the research group SERTEL. His research activity is in vehicular ad hoc networks (VANETs), focusing on routing and privacy.

Carolina Tripp-Barba received her MS and PhD in Telecommunication Engineering from the Universitat Politècnica de Catalunya (UPC) in 2009 and 2013, respectively. Currently, she is a full-time Research Professor at the Facultad de Informática Mazatln, Universidad Autónoma de Sinaloa (UAS), Mexico and member of the research group 'Seales y Sistemas'. Her research activity is focused on VANETs and wireless sensor networks (WSN).

David Rebollo-Monedero received the MS and PhD in Electrical Engineering from Stanford University in 2003 and 2007, respectively. He is a post-doctoral researcher with the Information Security Group of the Department of Network Engineering at UPC, where he investigates the application of data compression formalisms to privacy in information systems.

Ahmad Mohamad Mezher received his MS in Signals and Systems from the Central University of Las Villas (Cuba). Currently he's a PhD student on Network Engineering at Technical University of Catalonia and his research interest is focused on vehicular ad hoc networks (VANETs).

Mónica Aguilar received her MSc and PhD in Telecommunication Engineering from the Universitat Politècnica de Catalunya (UPC) in 1995 and 2000, respectively. She is an Associate Professor and a member of the Telematic Services Group in the department of Network Engineering at the UPC. Her research interests include design, analytical modelling and performance evaluation of multimedia services over vehicular adhoc networks (VANETs).

Jordi Forné received the MS in Telecommunications Engineering from UPC in 1992, and the PhD in 1997. Currently, he is an Associate Professor at the Telecommunications Engineering School of Barcelona (ETSETB), and is with the Information Security Group.

José Estrada-Jiménez received the MS in Telematics Engineering from UPC in 2013. Currently, he is an Assistant Professor at Escuela Politécnica Nacional in Ecuador.

1 Introduction

Vehicle communications are the object of rapidly increasing attention by the research community, largely driven by interests in road safety. VANETs (Hartenstein et al., 2010) may be regarded as a special case of mobile ad hoc networks (MANETs), where nodes are vehicles. Nevertheless, VANETs face particular challenges compared to MANETs, such as faster topology changes, smaller link lifetime, or a potentially higher number of nodes in the network, among others. In VANETs, both vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications are possible.

Owing to the very dynamic topology of VANETs, their routing protocols need continuous broadcasting of signalling messages which are used to keep track changes in the network infrastructure. Next, an address resolution (AR) mechanism is traditionally required to set up a traffic flow when a new IP communication process starts. We claim that this AR task can be performed by the network-topology updating process accomplished by VANET routing protocols. In this way, we leverage the functionality of the routing signalling and avoid the AR traffic periodically interchanged among nodes.

We show the undesired effects of AR traffic in vehicular communications and highlight the advantages and drawbacks of our proposal in a multi-hop scenario for a unicast traffic service. We extend the analysis done for MANETs in Carter et al. (2003) to VANET environments by using two routing protocols. On the one hand, we tested AODV (Perkins et al., 2003), a well-known topology-based protocol, since it is the basis for many recent developments; and on the other, a proposal for a geographical routing protocol named GBSR (Tripp Barba et al., 2013), which is an improvement of GPSR (Karp and Kung, 2000).

The main contributions of this paper are:

- an in-depth study of the AR behaviour in VANETs, by showing its negative impact on the overall performance of such networks
- a proposal to suppress the common AR process in a VANET by adding L2 addressing information to the signalling routing messages.

This approach prevents collisions and facilitates the tasks of the layers, such as the addition and subtraction of headers.

The rest of the paper is organised as follows. Section 2 summarises the AR procedure and particular its work flow in a wireless communication. That section also reviews the routing protocol classification and explains briefly the operation of AODV (Perkins et al., 2003), GPSR (Karp and Kung, 2000) and GBSR (Tripp Barba et al., 2013). Next, Section 3 summarises related work to AR in MANETs. Afterwards, Section 4 describes our scenario, and shows how AR mechanisms can be enhanced to minimise the traffic necessary to set up a data communication in a multi-hop scenario. This section ends with an explanation of how the updating topology process embedded in routing mechanisms can carry out such AR. Then, Section 5 is entirely devoted to the evaluation of the different AR schemes according to the strategies we proposed. Finally, conclusions are drawn in Section 6.

2 Background

Our contribution is inspired by the AR work logic carried out by the address resolution protocol (ARP) (Plummer, 1982) in IPv4 and neighbour discovery (ND) (Narten et al., 2007) in IPv6 networks. Our proposal of an AR scheme is supported by the operation of the VANET routing protocols.

In general, the literature states that IPv6 is the most appropriate technology to support VANET communications (Mohammad et al., 2011), given the exhaustion of IPv4 addresses and the new features offered by IPv6. The IPv6's ND protocol includes an AR mechanism which is substantially similar to that of ARP (Davies, 2012). Indeed, ND is more complex owing to the fact that it combines AR and ICMP router discovery and redirect mechanisms in its operation, as it is described in the IPv6's RFC 4861 (Narten et al., 2007). Since the IPv6's AR mechanism uses multicast communication, the traffic generated when requesting this service is theoretically much lower compared to that of ARP broadcast traffic. However, its wireless shared medium prevents VANETs from taking advantage of such multicast process since all neighbouring nodes still receive the requests, even when those are not processed by all the nodes. Hence, a significant amount of traffic is generated anyway, owing to the AR process. For the purposes of our analysis, both ARP and ND AR mechanisms have the same issues in terms of the amount of traffic they generate in a VANET scenario. Hence, we will use the parameters of both ARP and ND protocols when referring to the AR processes, provided that some simulation tools are still using the IPv4 stack.

In this section, we first present a detailed study of the AR operation to then propose (Section 4.2) a modified configuration of this AR mechanism to enhance its performance for mobile and dynamic topologies. Next, we summarise a classification for routing protocols in VANET and briefly describe one of each routing category, the traditional AODV protocol and GBSR which is an improvement of GPSR.

2.1 Address resolution

The link-layer AR process performed by a neighbouring node was standardised in ARP (Plummer, 1982) for IPv4 networks in 1982, and updated in neighbour discovery (Narten et al., 2007) for IPv6 networks in 2007. The operation of the AR mechanism is really simple. When a node A needs to send information (packets) to a node B, the AR daemon looks up in a translation table whether the B's hardware address exists. If A finds B's hardware address, then AR daemon puts it in the MAC header of the packets sent to B. If A does not know B's hardware address, node A broadcasts (in ARP) or multicast (in IPv6) a message to request such information. When node B receives the request, it updates its address translation table and responds to node A with a reply message containing the hardware address of B.

Below we describe some AR operation features, which are identical in ARP an ND, to suggest improvements when applied to a VANET scenario.

- When a node uses the address information from the sender's ARP message to *update* an entry in the address translation table, it is called a *Free AR update*. This operation is done with every ARP packet whatever the destination of the message is. This operation can be performed with ND with Unsolicited Neigbour Advertisement. However, *free updates* are still possible in VANETs, if vehicles process messages in promiscuous mode owing to the shared nature of the wireless channel.
- When a node *adds* addressing information obtained from a received ARP request to its address translation table, the process is known as *ARP learning*. ARP learning is performed in ND with Neighbour Solicitation Messages, which do the same job as ARP requests.
- According to RFC 1122 (Braden, 1989) a node should keep a queue per destination to store at least one packet (to locate the last incoming packet) until its hardware address is resolved. Most of the current ARP implementations only use this really short buffer. ND (Narten et al., 2007), also keeps the use of a short queue. Nevertheless, this buffer size may not be large enough for MANETs where collisions occur frequently and, therefore, the buffer could drop packets because a request/reply was lost.
- The recommended maximum ARP request rate is 1 message per second per destination, to prevent ARP flooding in IPv4 (Braden, 1989). In ND (Narten et al., 2007), *RetransTimer*, set the wait time before sending another neighbour solicitation. In addition, the maximum number of neigbour Solicitation messages to a node can be done through `MAX_UNICAST_SOLICITS`.
- Another common implementation issue is that the AR process can only be completed with the reception of an AR reply message. In other words, node A will only try to send those packets stored in its buffer and addressed to node B when an ARP reply message (from B and addressed to A) is processed. Nevertheless, a hardware address might have been resolved by another AR message, such as an AR message (reply or request) where node B had participated, that node A could have heard from its neighbourhood

The aforementioned AR features may provoke some issues on the VANET operation. In Section 4.2 we analyse some optimisation procedures and we propose a modified configuration of AR, which we claim could be more suitable for VANETs.

2.2 Routing protocols in VANETs

A widely accepted and general classification of VANET routing protocols is presented (Lee et al., 2009) where routing protocols are divided into two big categories:

Topology-based: These routing protocols do packet forwarding by using the information of the existing links in the network. The strategy can be divided into proactive (table-driven) and reactive (on-demand) routing. these two options can build a routing table.

Geographic (position-based): In these routing mechanism the nodes take their forwarding decisions based on the position of a packets destination and the positions of the nodes one-hop neighbours. Geographic routing assumes that each node knows its own location, and the sending node is aware of the receiving node's location using a global position system (GPS) unit. Many of these routing protocols store packets while they do not have a suitable forwarding node, a process which is called carry-and-forwarding. This specific type of protocols are appropriate for delay-tolerant networks and applications (DTN) owing to the inherent delay introduced by the carrying process.

In this subsection we summarise the main features of AODV (Perkins et al., 2003) and GBSR (Tripp Barba et al., 2013). AODV is a reference protocol for MANETs and has been the foundation for many implementations of topology-based routing protocols proposed for vehicular networks (Aswathy and Tripti, 2012) (Naumov et al., 2006). GBSR (Tripp Barba et al., 2013), however, is a novel proposal based on GPSR (Karp and Kung, 2000), which improves this protocol in terms of packet losses. For the sake of clarity in the results we found, only AODV and GBSR will be used in Section 5to assess the performance of our AR proposal. Both AODV and GBSR represent the two branches (topological and geographical) of classification commonly used over routing protocols for VANETs. Hence, in this work we aim to cover both sides of the aforementioned classification.

AODV (Perkins et al., 2003) is a reactive protocol that uses the Bellman-Ford distance vector algorithm which was adapted to work in a mobile environment. It determines an end-to-end route to a destination only when a source node wants to send a packet. Routes are maintained as long as they are needed by the source while there is connectivity between nodes in the path. Sequence numbers ensure the freshness of routes and guarantee a loop-free routing. An AODV node maintains a routing table that holds information like: destination address, next-hop address, hop count, destination sequence and lifetime. AODV has been evaluated in VANET scenarios under different circumstances, showing that it can still be suitable for VANET scenarios, especially in high density areas (Ferreiro-Lage et al., 2009; Haerri et al., 2006) and with a low number of active connections (Spaho et al., 2013). These conditions were considered in the configuration settings of the evaluation scenario we used in this work, which consider a unique and fixed destination. It is important to notice that these characteristics help to build end-to-end paths and to reduce the amount of signalling messages, which are some of the details considered in AODV-inspired protocols like (Aswathy and Tripti, 2012) where a cluster is created and only the leaders of the cluster can create routes; or in Naumov et al. (2006), where nodes in promiscuous mode are able to detect better routes. Moreover, since AODV needs more signalling traffic than its enhancements. The results obtained in the AR assessment for AODV can be considered the most pessimistic case for this branch of protocols.

Greedy buffer stateless routing (GBSR) protocol (Tripp Barba et al., 2013) was designed for delay tolerant applications. GBSR is an improvement of greedy perimeter stateless routing (GPSR) (Karp and Kung, 2000), which was specially designed for VANETs. GPSR uses two different techniques to forward packets: *greedy forwarding*, which is used by default; and *perimeter forwarding*, which is used in those regions where greedy forwarding cannot be used. With greedy forwarding, the neighbour geographically closest to the packet's destination (greedy choice) is chosen as the packet's next hop. When there is not a closest neighbour to destination, GPSR uses the right-hand rule to route packets around voids. The sequence of edges traversed by the right-hand rule is called *perimeter*.

GPSR poses two important drawbacks: the first one reported (Naumov et al., 2006) is the use of outdated information; the second one is the inefficient perimeter forwarding scheme (Gerla and Kleinrock, 2011). GBSR was proposed to tackle both issues by improving GPSR through a map-aware capability and a buffer-based scheme. GBSR stores packets in a buffer (carry packets with it) instead of implementing any perimeter mode when no closest-to-destination neighbour is found. The GBSR protocol, as GPSR does, uses distance as the only metric to make forwarding decisions. Additionally, GBSR sends more information within its hello messages to get a more accurate position estimation, thus being more restrictive in selecting neighbours (e.g., avoiding nodes with low power reception signal). All these modifications help GBSR to outperform GPSR in terms of packet losses at the expense of slightly increasing the average packet delay.

3 Related work

Issues with the AR operation in ad hoc networks have been reported in other works. Authors of a study about TCP performance over MANETs (Holland and Vaidya, 1999) found that a large amount of packet losses were owing to the combination of outdated routes, unresolved MAC addresses for those next forwarding nodes and a very short ARP queue. In a measurement study of vehicular internet access (Bychkovsky et al., 2006) using IEEE 802.11b networks, authors highlighted that the 1.5 out of 5 s necessary to initialise any data transmission, were owing to ARP procedures. The AR process has been studied previously in a context of MANETs. In the performance comparison of routing protocols for multi-hop wireless ad hoc networks done (Broch et al., 1998), authors noted a serious layer-integration problem between any on-demand MANET routing protocol and ARP. The problem arises when a route for the stored packets is found. If there is a next-hop destination whose link-layer address is unknown, all the queued packets, except the last one sent from the routing protocol queue to the ARP queue, will be dropped. This is owing to the short length of the ARP buffer (1 packet). Broch and colleagues (Broch et al., 1998) solved the problem by fixing the rate at which packets are passed from the routing queue to the ARP queue. Also, they realised that the size of the ARP buffer could be increased to store additional packets. Moreover, they suggested that the routing protocol could be aware that the ARP module already has the link-layer address of the next forwarding node.

Carter et al. (2003) presents the negative effects that result from the separation of ND and link layer AR. They focused their attention in the conflict produced in the interaction between a *manycast* application and ARP. They observed the problems arose owing to the dynamic topology in MANETs, which caused issues in the routing protocols because they need to use updated neighbour information. The conflict in *manycast* occurs when the route replies are propagated back to the *manycast* source. Carter and colleagues proposed (Carter et al., 2003) an approach called automatic address resolution (AAR) that is based on the maintenance of AR coupled with the ND process carried out by the routing protocol through exchange of signalling messages needed to establish routes. They claimed that any routing protocol could easily perform automatic AR by recording the link-layer address when a ND occurs. This is specially suitable for MANETs because this operation avoids the use of ARP messages. This ensures that the neighbour's link-layer address is already known when a unicast communication with that neighbour is needed. However, this proposal assumes that the link-layer source address is automatically accessible by the routing protocol and that AR does not use additional communication resources. We claim that this fact is not so evident and easy in real scenarios as it could seem. In any case, the results (Carter et al., 2003) show that automatic AR performs better than ARP regardless the type of routing protocol (reactive or proactive).

Finally, Choi et al. (2008) a communication scheme is proposed to enable some IPv6 procedures without link-scope multicast. Since the support of link-scope multicast is difficult for VANET scenarios, the authors propose packet delivery mechanisms that take advantage of inherent location management functions. They support the proposal in the use of C2C architecture to perform IPv6 operations. The work comes up with the use of additional C2C and Layer 2 headers to encapsulate IPv6 packets in order for them to reach an IP next hop and an Layer 2 neighbour. By using beacon messages they propose to send the 802.11p MAC address (among other type of information) of the source node in a communication process. This is how the ND process is carried out without depending on multicast.

4 Address resolution proposals

This section presents the major contribution of our work, namely a process we called coherent, automatic address resolution (CAAR) for IP multi-hop vehicular ad hoc networks (VANETs). First of all, we examine in Section 4.1 the vehicular network scenario we used. Next, in Section 4.2 we review some straightforward modifications to the AR procedure to improve its performance. Finally, Section 4.3 specifies the characteristics of our novel AR approach.

4.1 Application scenario

We focus our analysis on a multi-hop vehicular ad hoc network (VANET) that performs routing operations at the network layer. This approach has the advantage of preserving the original functionality of the different layers in the stack of protocols. Afterwards we describe the communication procedure performed in our application scenario.

4.1.1 Communication procedure

For the application scenario that we considered, the following assumptions were done:

- Two vehicles that are not in communication range of each other need a forwarding node able to connect these two nodes. Different from traditional wired systems, the packet forwarding operation occurs in VANETs even when nodes are in the same network addressing (within the same 'IP site'). For ad hoc networks, the concept of forwarding is more related to the physical distance between nodes rather than a logical hierarchy of network addressing.
- A node needs to have a valid non-link local IPv6 address since nodes cannot forward packets using link-local addresses (Hinden and Deering, 2006). In particular, for this work we assume that vehicles use a unique local address. A unique local address (Hinden and Haberman, 2005) enables packet routing only within a site, so packets are not expected to be routable through the internet.

The following steps were performed to enable communication in the scenario:

- A node obtains the IP configuration parameters from the IP site it belongs to. A vehicle in a multi-hop scenario has two ways to do this:
  - If vehicles have direct communication to an access point (AP), the road side units (RSUs) provide them with IP parameters for self configuration. The RSUs periodically broadcast the IP configuration (to one-hop neighbours) in the acknowledgement frame of the wireless access vehicular environment (WAVE) service, which is defined in the IEEE 1609.3 standard (IEEE Standard for Wireless Access in Vehicular Environments (WAVE) – Networking Services, 2010).
  - If a vehicle is located several hops from a RSU, it is unable to receive a service acknowledgement frame sent by the RSU. In such case, a vehicle has to use a multihop modification of the router discovery (RD) process. The RD Process is one of the functions of the ND Protocol, and consists of

two messages: Router Solicitation and the router advertisement. The latter carries the IP parameters (network prefix, MTU, etc.) of the vehicle network configuration.
A modification of multihop RD is presented (Wakikawa et al., 2006). It proposes the use of router request and reply messages to encapsulate the Router Solicitation and Advertisement messages. The router request is sent to a predefined multicast gateway address. The gateway will respond using a unicast Router Advertisement. A unicast temporal IP address is used by the initiator to receive the router reply. This initial and temporal IP address is built from a specific prefix called `MANET_INITIAL_PREFIX`. Alternatively, this initial IP address can be the valid IP used for the previous associated IP site.

- A vehicle auto-configures a unique-valid local IP address. After a node receives its IP configuration, generates for itself a unique interface ID (IID) to be used on the stateless IP address auto configuration process. The IID is appended to the network prefix to generate a unique local IP address. This unique ID can be generated using a random generator as proposed (Wakikawa et al., 2006). This process has to do duplicate detection. Another option is using the vehicle identification number (VIN) (Imadali et al., 2013), which may avoid the duplicate detection phase.

  Other IP configuration mechanisms for MANETs have been surveyed (Bernardos et al., 2010) and all of them can be applied to our VANET scenario.
- Once a vehicle is configured with a local-unicast IPv6 address, it can establish unihop or multi-hop IP communications, either V2V or V2I. In any case, both parts of a communication process will need to perform link-layer AR to reach the next forwarding hop before sending data packets.
- When a node leaves the influence area of the IP site to which the vehicle is associated, it will receive new IP configuration parameters announced by the WAVE service acknowledgement frame of another RSU in the new IP site or through multihop procedure. A node could be aware that it is not in the range of its associated IP site, when:
  - A vehicle receives Hello messages from neighbours with different IP range.
  - Based on its position, a node is able to know whether it is in the geographical area covered by its current IP Site. This information can be sent during the IP configuration process or by other means such as an IP Site bounds layer in the preloaded maps of the vehicle.
  - Besides, during the change from one IP site to other, the node could use its neighbours from the previous IP site still reachable to send information.

  Once a node obtains the new IP configuration and begins to discover neighbours in the new site, it should erase all the obsolete associations.

These steps show possible ways that enable vehicles to keep connected with their surrounding peers and other infrastructure services. In this work, we assume that address auto-configuration has already been accomplished, and that all the nodes are in the same IP site using stable and valid configuration. Below we explain the characteristics of the traffic we used for the VANET scenario.

4.1.2 Characterisation of the scenario's traffic

Caloca and Garcia Macias (2010) provide a classification of vehicular applications and their communication requirements. These categories are: safety, vehicular traffic efficiency and infotainment applications. Traffic flow control or environmental conditions monitoring are some of the aims of such applications. They might include delivering information to traffic authority centres about road conditions, air pollution, noise level, infraction reporting, reckless drivers, etc.

All the efficiency-oriented applications require a continuous monitoring phase of the streets and city conditions. Vehicles are the most suitable collectors of this information so they can be in charge of feeding the monitoring centres. Our application scenario incorporates data traffic generated by an efficiency application during the phase of collecting data only. The characteristics of the traffic are:

- Vehicles obtain data from their sensors, they process such data and generate constant-length packets.
- The packets are sent to the closest RSU. This is a unicast and unidirectional traffic, since the information is valid only to the authority. With the purpose of facilitating the delivery process of data, RSUs can be configured with an anycast address, which does not semantically differ from unicast IPv6 addresses, and the configuration would not change the AR process regarding the ND protocol (Davies, 2012).
- This kind of applications does not have important delay constraints as the safety related ones, so the transport of its traffic is suitable for delay-tolerant protocols which deal with variable and high delay, since the data is still useful.

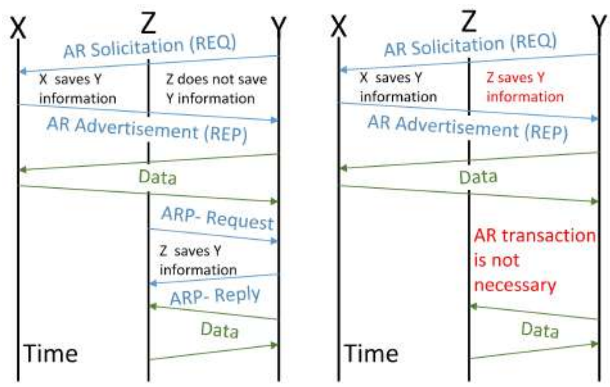
It is worth mentioning that, when the traffic authority has to send data related to traffic conditions to vehicles or drivers, this information will be interesting for the whole group of vehicles located in a specific area. Consequently, this information could be transmitted using geocast communication to a multicast group associated to a geographical area. In any case, in these kind of transmissions, AR is not necessary because the translation of multicast IP addresses to the associated Ethernet MAC address is well defined according to (Hinden and Deering, 2006) and (Haberman and Thaler, 2002).

4.2 Improvements to the address resolution procedure for ad hoc networks (AR+)

In vehicular networks, every node listens to the communications coming from the nodes within the coverage area (its neighbours). Also, the AR request messages are always received by the node regardless the recipient of the messages, so AR message could be easily processed by all nodes that anyway receive them to take advantage of their information. Based on those two facts, we propose AR+, which includes the following straightforward modifications of the AR operation to enhance the overall performance of the AR procedure.

- A node must perform the *AR learning operation*, as described in Section 2.1, using all the AR request packets that it hears, and not only from those packets addressed to the node. This change is based on the fact that all the AR packets that a node receives come from one of its neighbours, which might become next forwarding nodes for other packets in the near future. If that is the case, AR transaction might not be necessary since the nodes have already proactively saved the MAC addresses of possible next hops. In simple terms, a node adds or updates an entry of its AR table for any kind of AR message received, regardless of whether the node needs that entry for its current routes or not. Figure 1 shows the advantage of this scheme. By applying the proposed modification, the AR process would not be necessary since the address would already be resolved.
- A node will maintain an AR queue of the same length as that of the routing protocol queue. This prevents a packet in the AR buffer from being dropped when a new packet comes from the routing queue if the AR process is not completed yet. That is, we have a packet in the AR queue waiting for the AR process to be completed. If the AR queue is shorter than the routing queue,a new incoming packet from the routing queue may drop that stored packet. This problem can be avoided by just making both queues of the same length.
- Since the AR buffer is able to store more than 1 packet (its length is a design parameter), the AR process could send as many requests as stored packets in the AR module, but without exceeding the maximum AR request rate of 1 message per second per destination. In this way, a node has more than one chance to request an AR.
- A node will perform a proactive dequeuing of packets. Its AR module will not only try to send stored packets when an AR reply is received. In addition, it will try to dequeue packets when any AR message is heard. That is, given two neighbouring nodes (let us say A and B), node A might receive a request message coming from node B and going to any node (not necessarily to A) before the reply message from B arrives to A since, for example, the reply was lost. In this way, A can learn the MAC address of B even if there are losses in their own request-reply procedure.
- A final modification we propose to include in the AR procedure relies on the coherence of the services among layers. We claim that the 'validity ARP entry time' should be equal to the 'valid time of a route' in the case of topology-based protocols (e.g., AODV) and to the 'neighbour's lifetime' in geographical-based protocols (e.g., GPSR). The objective is to prevent a new AR transaction for a neighbour whose entry was previously deleted from the AR table (because its timer expired) while that node is still valid for the routing level (its entry in the routing table is still valid).

Figure 1 Extension of the AR learning operation: (a) traditional AR procedure and (b) extension of the AR learning (see online version for colours)



(a) (b)

All the previous changes were implemented in our enhanced AR process for VANETs which seeks to improve the vanilla AR in terms of losses and delay as well as to decrease the traffic necessary to perform the AR task. The improvements proposed to the AR resolution process might be interesting as strategies to enhance its performance without significantly changing the protocol.

4.3 A coherent, automatic address resolution (CAAR)

As we mentioned in the introduction of this paper, our work extends the AAR proposed (Carter et al., 2003) for MANETs. We called our proposal CAAR Coherent AAR because it agree much better with tasks assigned to each communication layer. CAAR is straightforward to be implemented, avoiding interactions between routing and MAC layers to perform the AR process. In this section we show the advantages of our approach by means a detailed description of the process flow of AAR and CAAR.

4.3.1 Automatic address resolution procedure

Carter et al. (2003) assumed that the source MAC address field of the MAC header is transparently available for the routing protocol daemon after receiving routing signalling messages. However, the implementation of a mechanism to make the MAC address available to the routing protocol is not a trivial task. The main reason lies in the extra work that the link layer should perform. We propose and show in Figure 2 two possible paths that a packet may follow when applying the approach of Carter et al. (2003). In both cases the link layer obtains the MAC header, reads the source MAC address and dispatches the frame payload to the IP layer. Figures 4 and 5 give details about the suggested implementations. A first possible implementation is:

Green path in Figure 2. The link layer pushes the Source MAC Address to the AR process which starts a timer (step 1, MAC-PROCESS algorithm in Figure 4) that waits an IP address to shape a Neighbour entry to be added in the neighbours' table. Next, the signalling packet is processed by the routing daemon (step 2, ROUTING-PROCESS in the algorithm), which extracts the IP source address and sends it to the AR daemon (step 3). Finally, a proper neighbour entry is added to the AR table (step 4, AR-PROCESS). If no IP address is provided to the AR process before the timer expires (step 5), then the MAC address is discarded, the same as TIMERHANDLER,in the algorithm.

Alternatively, the algorithm in Figure 5, differs from the previous one in the implementation of a timer, as follows:

Black path in Figure 2: The link layer saves the source MAC address and initialises a timer (MAC-PROCESS function). The signalling packet is treated by the routing daemon (step 1) which extracts the source IP address and sends it to the AR process (step 2, ROUTING-PROCESS sequence). The AR daemon (AR-PROCESS function) retrieves the source MAC address saved by the link layer (step 3) and writes the new information (an AR-ENTRY) in the neighbour table (step 4). As in the first process flow, if the timer initialised by the link layer expires (step 5), the MAC address is discarded by the TIMERHANDLER procedure.

For both alternatives, if a new packet arrives before the AR-ENTRY (corresponding to the previous packet) is created, both MAC addresses are discarded. This process can be seen in the validation done in AR-PROCESS for the first alternative (line 13), and in the MAC-PROCESS for the second alternative (line 2). We would like to point out that there are many options or extra steps that may be performed over the mechanism to do the AR, as explained before. Nevertheless, we think that all these improvements entail too much extra work to the link and routing layers and, furthermore, they are not necessary to carry out the AR.

4.3.2 Process flow of CAAR

CAAR is really straightforward. Every node copies its MAC address, available for instance from its management information base (MIB), into the signalling routing messages as a new field of them. This action will slightly increase the length of those signalling messages in 6 bytes with the purpose of not augmenting the MAC layer complexity or cross layering cooperation between MAC and routing layers. The scheme of our proposal, depicted in Figure 3 and its algorithm (in Figure 6), works only at the network layer. This is explained below.

CAAR path in Figure 3: When a routing signalling packet arrives to a node, it is processed by the routing daemon (step 1) which extracts the source IP and MAC addresses (step 2, ROUTING-PROCESS algorithm) from the IP header and the signalling message, respectively. We do not need to use timers, since the match is directly done when an IP Packet encapsulated in a routing message is received. Such IP packets, as depicted in Figure 7, contain both IP and MAC addresses at the same communication layer. Next, the pair of IP and MAC addresses is sent to the AR process (step 3), which adds or updates an entry to the AR table.

Figure 2 Possible packet flows in the automatic address resolution (see online version for colours)

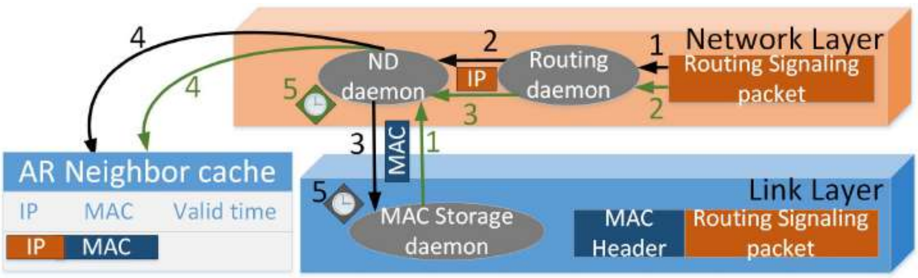
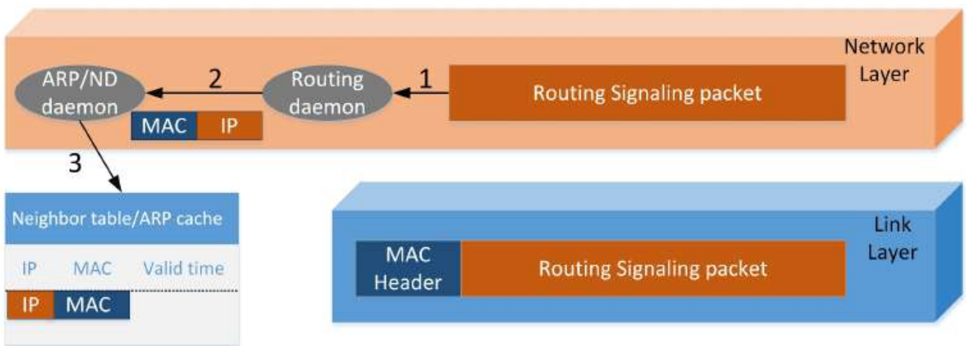


Figure 3 Scheme of our CAAR proposal (see online version for colours)



The AR-PROCESS algorithm in CAAR is extremely simple compared to the tasks assigned in the AAR alternatives. The main advantage of our approach is the minimal extra work introduced in the routing process, whose goal is to attain a proper addresses tuple (MAC, IP) to be used by the AR process. Notice that the operation of the link layer remains invariable. Moreover, the CAAR is *automatic* because it matches IP and MAC addresses without using any AR signalling.

Figure 4 Automatic address resolution algorithm. Alternative 1

```
MAC-PROCESS(P)

Require: A packet P
Ensure: MAC_src of P to AR-PROCESS.
MAC_src ← GETSOURCEMAC(P)
AR-PROCESS(MAC_src , NULL)
return

ROUTING-PROCESS(P)

Require: A packet P
Ensure: IP_src of P to AR-PROCESS.
IP_src ← GETSOURCEIP(P)
if ISROUTINGMESSAGE(IP_src)= TRUE then
    AR-PROCESS(NULL , IP_srcP)
return

AR-PROCESS(MAC_srcP , IP_srcP)

Require: MAC_src or IP_srcof P
Ensure: AR-ENTRY(MAC_src , IP_src) to TableARP.
if MAC_src ≠ NULL then
    if TEMP_MAC ≠ NULL then
        KILL(Timer)
        KILL(TIMERHANDLER)
        DELETE(TEMP_MAC)
        DELETE(MAC_src)
        return Fail
    else
        TEMP_MAC = MAC_src
        INIT(Timer)
        INIT(TIMERHANDLER)
if IP_src ≠ NULL then
    if TEMP_MAC = NULL then
        DELETE(IP_src)
        return Fail
    else
        TEMP_IP = IP_src
        Entry=AR-ENTRY(TEMP_MAC , TEMP_IP)
        ADDTABLEARP(Entry)
        KILL(Timer)
        KILL(TIMERHANDLER)
return Success

TIMERHANDLER()

Require: active Timer
Ensure: DELETE(TEMP_MAC) of AR-PROCESS
if Timer = 0 then
    AR-PROCESS.DELETE(TEMP_MAC)
else
    nothing to do
return Success
```

Figure 5 Automatic address resolution algorithm. Alternative 2

```
MAC-PROCESS(P)

Require: A packet P
Ensure: MAC_src of P to AR-PROCESS.
MAC_src ← GETSOURCEMAC(P)
if TEMP_MAC ≠ NULL then
    KILL(Timer)
    KILL(TIMERHANDLER)
    DELETE(TEMP_MAC)
    DELETE(MAC_src)
    return Fail
else
    INIT(Timer)
    INIT(TIMERHANDLER)
    TMP_MAC ← MAC_src
return

ROUTING-PROCESS(P)

Require: A packet P
Ensure: IP_src of P to AR-PROCESS.
IP_src ← GETSOURCEIP(P)
if ISROUTINGMESSAGE(IP_src)= TRUE then
    AR-PROCESS(IP_srcP)
return

AR-PROCESS(IP_srcP)

Require: IP_srcof P
Ensure: AR-ENTRY(MAC_src , IP_src) to TableARP.
if TEMP_MAC = NULL then
    DELETE(IP_src)
    return Fail
else
    TEMP_IP = IP_src
    Entry=AR-ENTRY(TEMP_MAC , TEMP_IP)
    ADDTABLEARP(Entry)
    KILL(Timer)
    KILL(TIMERHANDLER)
return Success

TIMERHANDLER()

Require: active Timer
Ensure: DELETE(TEMP_MAC)
if Timer = 0 then
    DELETE(TMP_MAC)
else
    nothing to do
return Success
```

4.4 CAAR implementation in VANET routing protocols

We have explained the advantages in the implementation of CAAR at the cost of minimal overhead in the routing signalling message. Since the AR process is triggered by the reception of a routing message, the AR procedure is done for all the neighbours, which are possible next forwarding nodes. This inhibits the exchange of neighbour solicitation and the neighbour advertisement messages of AR signalling.

Figure 6 CAAR algorithms

ROUTING-PROCESS(P)

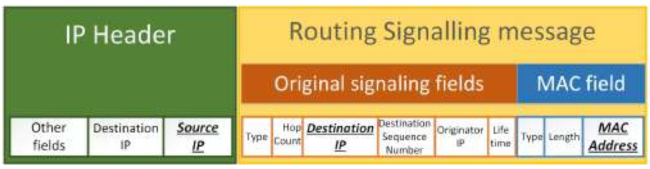
Require: A packet P
Ensure: MAC_{src} **and** IP_{src} of P to AR-PROCESS.
$\text{MAC}_{src} \leftarrow$ GETSOURCEMAC(P)
$\text{IP}_{src} \leftarrow$ GETSOURCEIP(P)
if ISROUTINGMESSAGE(IP_{src})= $TRUE$ **then**
    AR-PROCESS($\text{MAC}_{src}P$, $\text{IP}_{src}P$)
return

AR-PROCESS($\text{MAC}_{src}P$, $\text{IP}_{src}P$)

Require: MAC_{src} or $\text{IP}_{src}$of P
Ensure: AR-ENTRY(MAC_{src} , IP_{src}) to TableARP.
$\text{TEMP}_{MAC} = \text{MAC}_{src}$
$\text{TEMP}_{IP} = \text{IP}_{src}$
Entry=AR-ENTRY(TEMP_{MAC} , TEMP_{IP})
return Success

Our proposal is *coherent* in the implementation with VANET routing protocols. First, it only adds the MAC address in a new field and performs AR within the routing messages used to create or maintain routes, which are common tasks for all VANET routing protocols. Figure 7 shows the IP encapsulation for a hello message structure. Notice that the new MAC field is added at the end of the signalling message to facilitate its processing by the routing daemon. This is suggested in case other type of MAC address different than Ethernet is used. In such cases, the MAC field should include the type and length sub-fields to distinguish the MAC protocol. This approach is the same implemented by ND to deal with different Source/Target Link-layer Address (Narten et al., 2007). Nevertheless, the use of a different MAC address format seems unlikely because WAVE architecture only supports IEEE MAC 802.11p (Morgan, 2010). In this work we encapsulate only the MAC address into the MAC field of signalling messages.

Figure 7 An IP packet carrying a VANET routing signalling message (see online version for colours)



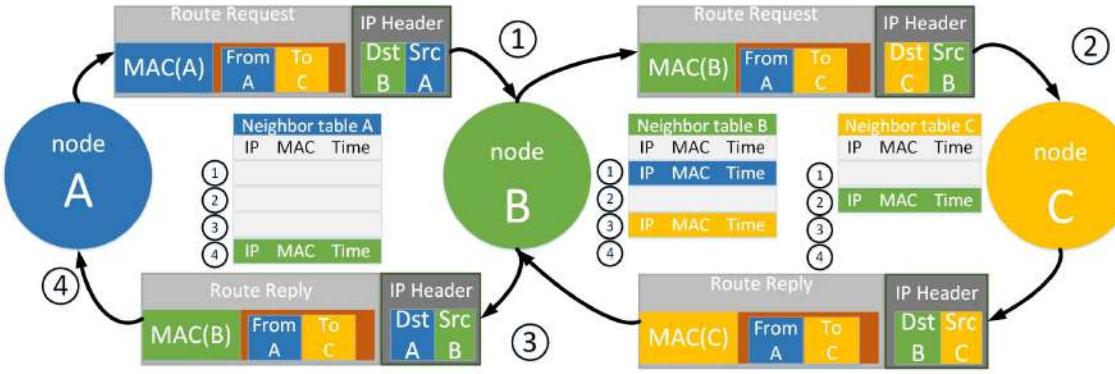
In *topology-based protocols*, the MAC field has to be added in the route request (RREQ) and route reply (RREP) during the route discovery process and in the HELLO messages part of the maintenance process. Figure 8 depicts the operation of CAAR during the creation of a route. Notice that while the MAC address is obtained from the signalling message, the source IP address is read from the IP header, since this IP address corresponds to the previous hop. The destination and originator's IP addresses from the signalling message are part of the end-to-end path (e.g., node A and C) and do not change during the path construction. Obtaining information from different parts of the IP packet (header and payload) is a task already implemented by routing protocol and it is done to create the backward path. Therefore, this does not really entail extra work. An intermediate node in the path (node B in the example) forwards a routing message containing its MAC and IP address in the IP header. These read/write operations on the signalling message should not be considered as an additional task owing to the fact that a forwarding node must anyway have to update information, like number of hops, in the signaling message. HELLO messages, for example, are one-hop keep-alive messages that only inform about the presence of a node. Thus, the work of CAAR in this situation is receiving a RREP message an not forwarding it. CAAR is not implemented through Error notification messages, which inform about invalid routes that are not going to be used any more. This might represent important savings in bandwidth, taking into account that VANETs have a very dynamic topology where routes get broken frequently.

Since the messages we use to implement CAAR are needed for every topological protocol (no matter which routing criteria is implemented) in the building process route, such messages can be leveraged to transport additional information by adding fields to them. More important CAAR is independent that the routing criteria used by node to create the path since the MAC is carried in all the signalling packets relevant to these tasks.

Geographical-based protocols are usually associated with a greedy approach because they make the forwarding decision at the arrival of each packet, based on local information only. This means that this category of routing protocols only need information coming from nodes at a distance of one hop. Therefore, most of the routing protocols implement only Hello signalling messages with a number of fields depending on the information that the protocol uses to forward packets. Since Hello messages can be considered as one-hop route replies, its destination field (see Figure 7) contains the IP address of the originating node, which is also the source IP address in the IP header. Thus, the AR process can be performed without using any information from the IP header.

Finally, CAAR is *coherent* in the management of expiration timers. We make the lifetime of ND cache entries equal to the route lifetime in topology-based protocols (like AODV) or equal to the neighbour lifetime in position-based protocols (like GBSR). It means that when the route or neighbour entry is created, the ND entry is also created with the same timer, and both timers are updated by the reception of a HELLO message. This simple action prevents unnecessary ND transactions since an ND timer with a timeout shorter than the route/neighbour timer's causes a ND entry to be deleted (its timer expires) even when the corresponding node is still a valid next forwarding node (its timer is still alive). But then, a ND timer with timeouts longer than the route/neighbour timer entails an unnecessary larger ND cache with entries useless for the routing daemon.

Figure 8 An IP packet carrying a VANET routing signalling message (see online version for colours)



5 Experimental results

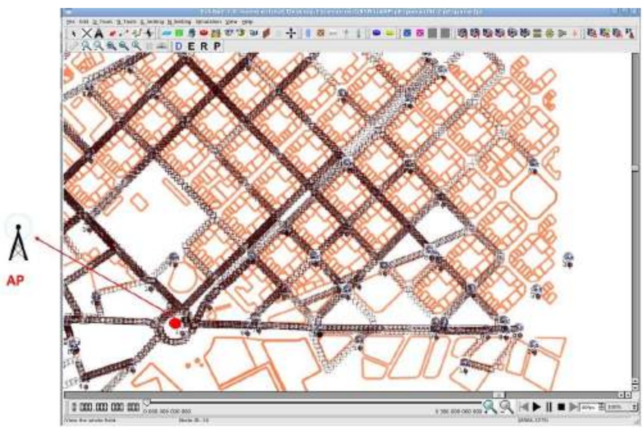
5.1 Simulation scenario

We analysed the performance of our proposed CAAR, which includes the MAC address in the routing signalling messages. We compared CAAR to the traditional AR named TAR (using the default operation) and AR+ (our improvement that modifies some parts of the traditional AR). We used AODV (Perkins et al., 2003) and GBSR (Tripp Barba et al., 2013) to evaluate the behaviour of our proposal. To do this we carried out several simulations using the Estinet Network Simulator and Emulator (Estinet-Technologies, 2014). Estinet is a simulator that includes the standard IEEE 802.11p and a simple and accurate way to design VANET realistic scenarios.

We used a real scenario of 1.5 km^2 taken from the Eixample district of Barcelona (see Figure 9) to model an urban scenario formed by streets and crossroads. Seeking to simulate a realistic scenario, the mobility model was obtained with CityMob for Roadmaps (C4R) (Fogue et al., 2012), a mobility generator that uses the SUMO engine (Krajzewicz et al., 2012). C4R is able to import maps directly from the OpenStreetMap (Open Street Maps, 2014) and to generate NS-2 compatible traces, which we transformed to Estinet traces using our own translator software. Furthermore, the scenarios have buildings information (orange lines in Figure 9) extracted from the OpenStreetMap and imported to Estinet using the SUMO tools and our own code to generate Estinet-Buildings. The buildings' walls attenuate the signal of vehicles during the simulation process.

We considered three amounts of vehicles: 60, 100 and 140 vehicles, which correspond to a density of 40, 67, and 93 vehicles per km^2, respectively. The movements of the vehicles are randomly defined in each repetition by the movement generator C4R to compute confidence intervals. There was one fixed destination, the AP(henceforth called AP), see Figure 9, that receives vehicles' traffic information. We used a single AP in the scenario because in this way we obtained a long range of route lengths, which depends on the position of the source vehicles into the scenario. All nodes sent 1000-bytes packets every T seconds to the unique destination during 300 s. T follows a uniform distribution from 2 s to 6 s. Simulations were carried out using the IEEE 802.11p standard on physical and MAC layers. We used only the Best Effort (BE) access category in the MAC layer. We used an empirical model of radio shadowing (Sommer et al., 2011) in IEEE 802.11p networks as path loss model. Also, as fading model we used Rician when vehicles are in line of sight (LOS) and Rayleigh when vehicles are not in LOS (Rappaport, 2001). We set a receiving sensing of -82 dbm according to the receiver performance requirement for 6 Mbps specified in the IEEE 802.11 standard (IEEE Standard for Information technology–Telecommunications and Information Exchange between Systems Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 2012), which corresponds to a sensing range around 400 m in LOS. All the figures are presented with confidence intervals (CI) of 95% obtained from 10 simulations per point using independent mobility seeds per simulation. Table 1 summarises the main simulation settings. Regarding AR parameters, Table 2 shows the values set in the simulations for the 3 AR schemes. The parameters used are the discussed on Section 2.1 and others match the timers of the routing protocols.

Figure 9 Simulation scenario. Eixample district of Barcelona with an access point (AP). Building from OpenStreetMap are included (see online version for colours)



5.2 Simulation results

In this section we present some representative simulation results of the general behaviour of the two evaluated routing protocols (i.e., AODV and GBSR) with the three AR schemes (i.e., TAR, AR+ and CAAR).

We have measured five results: percentage of packet losses, delay, average number of hops, average number of neighbours and incurred AR signalling. For a deep statistical analysis of the results, we have performed the Jonckheere-Terpstra test (Sheskin, 2000) (called J-T test henceforth) to check if the three AR schemes show the same behaviour (known as *null hypothesis*) or if the results indicate a certain tendency (known as *alternative hypothesis*) in relation to a given ordering among the AR schemes. The order used in this work to arrange the AR schemes and carry out the J-T test was TAR, AR+ and CAAR.

Table 1 Simulation settings

Parameter	*Value*
Number of nodes	60, 100 and 140 vehicles
Map zone and Area	Eixample district of Barcelona, 1.5×1 km
Path loss model	Empirical IEEE 802.11p radio shadowing (Sommer et al., 2011)
Fading model	Rician (LOS) and Rayleigh (not in LOS)
Power transmission	23 dbm
Receiving sensing	-82 dbm ($\sim$ 400 m in LOS)
Mobility generator	SUMO (Krajzewicz et al., 2012) / C4R (Fogue et al., 2012)
Mobility model	Krauss (Krauss et al., 1997)
Max speed	60 km/h
MAC specification	IEEE 802.11p
QoS access category	BE (Best Effort)
Bandwidth	6 Mbps
Simulation time	300 s
Maximum packet size	1000 bytes
Traffic profile	Uniform distributed from 1.3 to 4 Kbps
Routing protocol	AODV (Perkins et al., 2003), GBSR (Tripp Barba et al., 2013)
GPS precision	10 m

Table 2 Parameter settings of address resolution

	AR schemes		
Parameter	*TAR*	*AR+*	*CAAR*
AR buffer ($pkts \times dst$)	1	30 = routing buffer	1
AR request rate ($pkts/s$)	1	1	–
Max AR-Req	3	= AR queued pkts	–
Lifetime	3	6 (AODV route lifetime)	
AR entry (s)		2 (GBSR neighbour lifetime)	
Table checker		0.300	
timer (s)	1	Same as routing checker	
Update entry Process	AR pkts to node	AR pkts to anyone	routing pkts

In the cases in which the J-T test stated that the results among the AR schemes were statistically the same (and therefore the test could not reject the Null Hypothesis), we carried out the Kruskal-Wallis test (Sheskin, 2000) (called K-W test henceforth) to assess whether the three results followed the same distribution (*null hypothesis*), or if there was any difference in the results among TAR, AR+ and CAAR (*alternative hypothesis*) regardless that the difference among those results did not follow a clear tendency in the J-T test. These tests are summarised in Table 3 for AODV and in Table 4 for GBSR. In the results of J-T tests we included the value of the standardised test statistic (STS), which can be seen as a measure of the correlation existing between the results assessed and the AR scheme implemented. The sign of STS indicates if the result increases (positive sign) or decreases (negative sign) when we change the AR scheme following the evaluating order established between schemas (i.e., TAR, AR+ and CAAR). The magnitude of the STS gives a notion of how much strong is the result's change among the AR schemes. Additionally, we included for all the cases where there is a difference among the results of the AR schemes (i.e., the null hypothesis is rejected), a pairwise comparison between the AR schemes to visualise the tendency in the J-T test or the differences in the K-W test. These pairwise results are presented in Table 5. All the statistical analysis of the simulation results were done with the software SPSS (IBM, 2014).

From the packet losses point of view, drawn in Figure 10(a), in the case of GBSR (last three columns of each group of columns) we can notice that losses decrease as we included improvements in the AR operation, i.e. AR+ improves TAR, whereas CAAR further improves AR+. Losses-AR dependability decreases with the number of nodes (see how the STS negative value of the first row in Table 4 increases in magnitude as N increases). CAAR performs always significantly better than TAR in terms of losses even in the low density scenario (see Figure 10(a)). This behaviour is owing to the hop-to-hop forwarding done in GBSR for each packet that requires AR procedures in each hop. At the moment of taking the forwarding decision, the use of a buffer with AR+ and the no necessity of any AR transaction with CAAR alleviates both the packet discarding using AR+ and the collisions using CAAR.

For AODV (first three columns of each group in Figure 10(a)), none of the proposed AR schemes improves TAR when the density of vehicles is not low. We can see in the first row of Table 3 for N=100 and 140 vehicles that the test decision is *Retain* the null hypothesis, which means that there is not statistical significant variation among the three AR schemes. The reason lies on how AODV works. While a route is being created and an AR transaction (REQ/REP) of an AODV-RREP message fails, that route construction process ends. Nevertheless, another path could easily be established if there were enough vehicles. During the route reconstruction process, packets are stored in the AODV buffer, not in the AR one; therefore, the short buffer of vanilla AR is not a concern in this situation. A low density of vehicles hinders the end-to-end route construction, and if the AR transaction fails in that case, a different path can not be set quickly. This provokes a lot of packet losses. AR+ deals better than TAR in this circumstance (as it can be seen in Figure 10(a) for N=60 nodes) thanks to the *Free AR learning* and *coherent timers* features (explained in Section 4.2) rather than to the larger buffer implementation. The CAAR scheme does not present significant differences compared to AR+ since AR+ does not require many AR transactions in this low density scenario.

In the case of delay, see Figure 10(b), GBSR (last three columns) increases the delay as the AR operation changes from TAR to CAAR, i.e., AR+ produces delays higher than AR and CAAR shows delays higher than AR+. See the STS values in the second row of Table 4 in which the positive STS values indicate an increasing tendency in the delay as the AR scheme changes. This is because the improvement in packet losses produces that packets from farther positions can reach destination. This occurs with AR+ and CAAR (see Figure 10(a)), so the average delay in those schemes increases (packets coming from farther sources take longer to reach destination). The difference of delay between AR+ and CAAR is only statistically significant for the high density scenario. This can be seen in the 11th row of Table 5, for the delay with GBSR and N=140 nodes. For CAAR-AR+ on $N = 140$, the adjusted significance value below 0.05 means that the delays with both AR schemes behave statistically differently; while for CAAR-AR+ for $N = 60$ (fifth row) and $N = 100$ (eight row), the adjusted significance is above 0.05, meaning that they behave similarly. The reason is that CAAR maintains the collisions low while AR+ cannot avoid them owing to the high number of nodes. The delay with GBSR increases very much for long distances (which are much possible using AR+ or CAAR than using TAR) because it performs an conservative next hop selection, storing packets in a buffer instead of dropping them when there is no proper forwarding node. This feature improves packet losses in GBSR (see Figure 10(a)) but increases its average end-to-end delay (see Figure 10(b)). On the other side, AODV (first three columns) shows no significant difference in the delay among the three AR schemes even for the low density scenario ($N = 60$ vehicles). This is because AR mainly works during path constructions and once the path is established the transmission of packets is fast, hence the delay is not much affected by using TAR because AR+ and CAAR only help to a faster path creation.

Regarding the number of hops, depicted in Figure 10(c), in the case of GBSR (three last columns of each group) they are completely related with the delay and the packet losses. With AR+ and CAAR destination nodes can receive packets from farther sources, generating longer paths with higher number of hops. Looking at Table 5, we can see a significant difference in the average number of hops between AR+ and CAAR when there is also a significant difference in the delay. For instance, the aforementioned significant difference between CAAR-AR+ in delay for N=140 is also present in the behaviour of hops, as it can be seen in rows 11th and 12th of Table 5 with an adjusted significance below 0.05. In opposition, for lower densities there is no significant difference as the reader can see in Table 5 in rows 5th and 6th for $N = 60$ (AS > 0.05) and in rows 8th and 9th for $N = 100$. No substantial change in the average number of hops can be seen for AODV as a function of the AR scheme (see the three first columns of Figure 10(c)). The reason is that the number of hops is directly related with the delay, which is also rather constant (see Figure 10(b)). Notice that the results of the average number of hops also allows us to realise about a well-known behaviour of AODV specially in high density areas that is the presence of longer paths compared to a geographical protocol (e.g., GBSR), for the same level of losses (compare AODV-AR+ to GBSR-AR+ for $N = 140$ in Figures 10(a) and (c)).

We decided to include Figure 10(d) that depicts the average number of neighbours obtained from the neighbours' table used by each protocol, because it shows two important things. The first one is that GBSR is more restricted than AODV in selecting neighbours and therefore GBSR has less candidates for choosing next forwarding nodes. And secondly, it allows us to understand why AODV with TAR (first column) performs so poorly in the path establishment in the low density scenario ($N = 60$). The reason is that only nodes with active routes send hello messages in AODV and in the low density area most of the nodes do not have as much active routes with AODV-TAR than with AODV-AR+ or AODV-CAAR. Consequently, the number of registered neighbours in AODV-TAR will be lower than using either AODV-AR+ or AODV-CAAR schemes.

We also included an analysis of the AR signalling in our evaluation. The incurred AR signalling was calculated in the case of CAAR as the extra bytes added to the routing messages (e.g., RREP, RREP and HELLO) to carry the MAC address. On the other hand, the TAR and AR+ signalling were computed using the total number of REQ/REP messages of AR transactions. Figure 11 presents the results with the three densities evaluated. For GBSR (three last columns) there is no significant difference among the signalling traffic introduced by the three AR resolution schemes (see the significance values in the last row of Table 4, which are above the significance level 0.05 meaning that the three results follow the same distribution).

Based on these results analysed so far, we can conclude that GBSR-CAAR improves the packet-losses performance thanks to a coherent and straightforward mechanism to perform the AR process without introducing extra signalling overhead. On the other side, the reactive protocol AODV maintains the same level of overhead for low-density and medium-density scenarios, although its AR signalling tends to decrease with AR+ and CAAR schemes in the 140-nodes simulations (see STS = –2.053 in the last row of Table 3 for $N = 140$, meaning that the tendency in AR signalling decreases as the AR scheme changes). Therefore, at the sight of these results AODV with either AR+ or CAAR improves the performance of traditional AODV-TAR for low density scenarios without adding any signalling, while decreases the overhead in high density scenarios maintaining the level of packet losses. To conclude, we summarise the major results drawn from our experimental evaluation:

- First, our simulations show that the performance evaluation results of the two routing protocols under study, AODV and GBSR, have a statistically significant dependence with the AR scheme implemented.
- Secondly, the results show that our proposal GBSR-CAAR under a unicast traffic always behaves better than the traditional AR or than our improvement AR+ in terms of packet losses. As consequence, the average number of hops and the average delay increases

Table 3 Hypothesis Test Summary of AODV for unicast traffic

Parameter hypothesis	*Test 60/100/140*	*N = 60* *STS*	*Significance*	*Decision* $p > 0.05$	*N = 100* *STS*	*Significance*	*Decision* $p > 0.05$	*N = 140* *STS*	*Significance*	*Decision* $p > 0.05$
Losses	K-W/K-W/K-W		0.029	*Reject*		0.72	Retain		0.301	Retain
Delay	K-W/K-W/K-W		0.536	Retain		0.51	Retain		0.191	Retain
Hops	K-W/K-W/K-W		0.191	Retain		0.1	Retain		0.876	Retain
Neighbours	J-T/K-W/K-W	2.75	0.006	*Reject*		0.41	Retain		0.829	Retain
AR signaling	K-W/K-W/J-T		0.847	Retain		0.16	Retain	–2.05	0.04	*Reject*

J-T = Jonckheere-Terpstra tend test. K-W = Kruskal-Wallis test. STS = Standardised Test Statistic. We performed J-T test for all the performance metrics. When the J-T test retains the null hypothesis, we performed K-W to look for any difference. Null hypothesis: The distribution of the parameter's result is the same across ARP, ARP+ and CAAR. J-T Alternative hypothesis: The distribution of the result follows an order across ARP, ARP+ and CAAR. K-W Alternative hypothesis: There is at least one scheme for which its distribution is different from the other schemes. In all the tests, we rejected the null hypothesis when the significance value is lower than the *significance level of 0.05*.

Table 4 Hypothesis test summary of GBSR for unicast traffic

Parameter hypothesis	*Test 60/100/140*	*N = 60* *STS*	*Significance*	*Decision* $p > 0.05$	*N = 100* *STS*	*Significance*	*Decision* $p > 0.05$	*N = 140* *STS*	*Significance*	*Decision* $p > 0.05$
Losses	J-T/J-T/J-T	–3.15	0.002	*Reject*	–3.23	0	*Reject*	–4.53	0	*Reject*
Delay	J-T/J-T/J-T	4.11	0	*Reject*	4.53	0	*Reject*	4.72	0	*Reject*
Hops	J-T/J-T/J-T	4.37	0	*Reject*	4.87	0	*Reject*	4.64	0	*Reject*
Neighbours	K-W/K-W/K-W		0.726	Retain		0.95	Retain		0.671	Retain

J-T = Jonckheere-Terpstra tend test. K-W = Kruskal-Wallis test. STS = Standardised Test Statistic. We performed J-T test for all the performance metrics. When the J-T test retains the null hypothesis, we performed K-W to look for any difference. Null hypothesis: The distribution of the parameter's result is the same across ARP, ARP+ and CAAR. J-T Alternative hypothesis: The distribution of the result follows an order across ARP, ARP+ and CAAR. K-W Alternative hypothesis: There is at least one scheme for which its distribution is different from the other schemes. In all the tests, we rejected the null hypothesis when the significance value is lower than the *significance level of 0.05*.

Table 5 Pairwise comparison of performance metrics for common unicast traffic (see online version for colours)

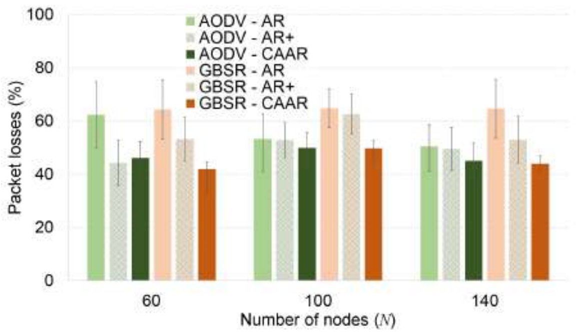
# row	*# Nodes*	*Protocol*	*Parameter*	*Adjusted Significance (AS)* *CAAR-AR*	*CAAR-AR+*	*ARP-AR+*
1	60	AODV	Losses	0.202	1	*0.029*
2	60	AODV	Neighbours	*0.006*	1	*0.006*
3	140	AODV	AR Signaling	0.144	1	*0.029*
4	60	GBSR	Losses	*0.01*	0.088	0.144
5	60	GBSR	Delay	*0*	0.818	*0*
6	60	GBSR	Hops	*0*	0.226	*0.001*
7	100	GBSR	Losses	*0.004*	*0.006*	0.895
8	100	GBSR	Delay	*0*	0.226	*0*
9	100	GBSR	Hops	*0*	0.051	*0*
10	140	GBSR	Losses	*0*	*0.01*	*0.019*
11	140	GBSR	Delay	*0*	*0.004*	*0.012*
12	140	GBSR	Hops	*0*	*0.001*	0.074

The comparisons were performed using the Mann-Whitney test. Null hypothesis: The results of the two AR schemes come from the same distribution. Alternative hypothesis: The distributions of results of the two AR schemes are different. In all the tests, we rejected the null hypothesis when the adjusted significance value is lower than the *significance level of 0.05*..

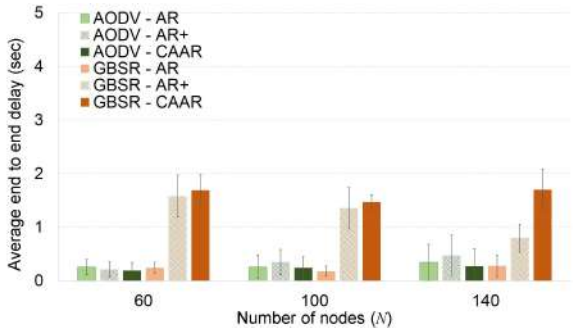
because packets from longer paths reach the destination. On the other hand, AODV-CAAR outperforms in terms of packet losses to the other AR schemes in low density cases, and maintains the same results as AODV-TAR in denser scenarios.

- Finally, our statistical analysis shows that CAAR in any case introduces more overhead, keeping it on the same level as classical AR over GBSR and even lower over AODV in high density areas.

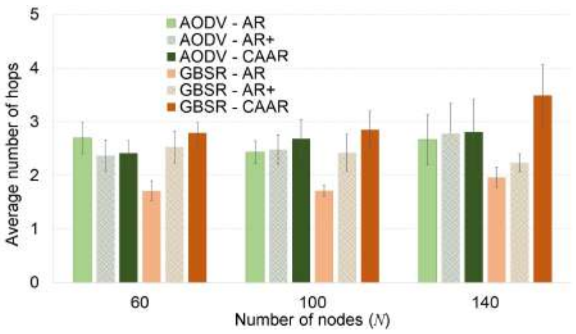
Figure 10 Evaluation of AODV and GBSR for unicast traffic using the three address resolution schemes (CI 95%): (a) percentage of packet losses; (b) average end-to-end delay; (c) average number of hops and (d) average number of neighbours (see online version for colours)



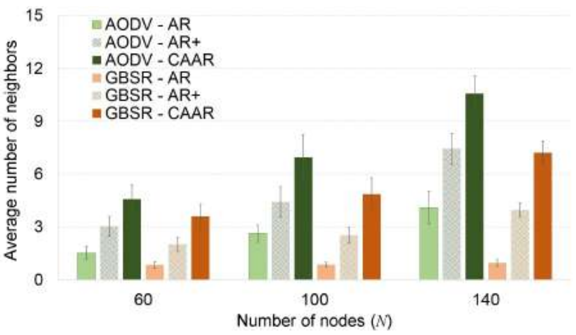
(a)



(b)



(c)

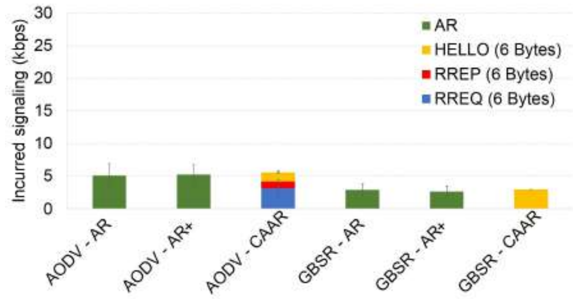


(d)

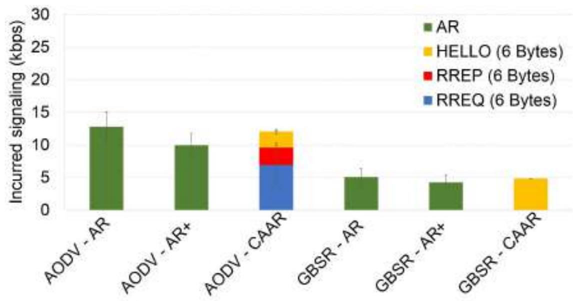
6 Conclusions

The main contribution of our work is a new scheme to carry out the AR process in Ad Hoc networks, which we called coherent, automatic Address resolution (CAAR), which improves the traditional AR in vehicular scenarios.

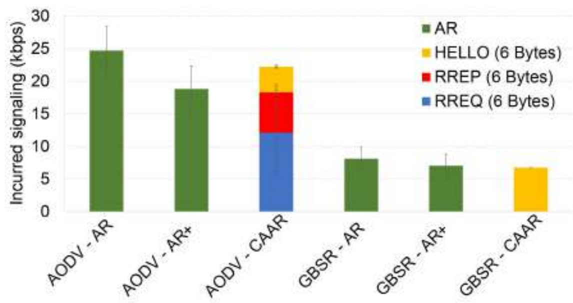
Figure 11 Signalling traffic incurred only by the AR process in the unicast traffic scenario using the three Address Resolution schemes (CI 95%). AR signalling in traditional AR and AR+ involves all AR REQ/REP messages. AR signalling in CAAR only consists of 6 bytes added to each routing message (RREQ, RREP and HELLO in AODV and HELLO in GBSR) to carry the MAC address: (a) scenario with $N = 60$ nodes; (b) scenario with $N = 100$ nodesand (c) scenario with $N = 140$ nodes (see online version for colours)



(a)



(b)



(c)

We evaluated the behaviour of CAAR and compared it to traditional AR and to our improved AR, called AR+, in a realistic VANET scenario for three different vehicle densities. We obtained results in terms of packet losses, average end-to-end packet delay, average number of hops and average number of neighbours, to compare the behaviour of each AR scheme. Moreover, the differences and behaviours reported in this work are sustained on non-parametric statistical tests.

We noticed that CAAR, when implemented over geographical protocols like GBSR, has an improved performance in terms of packet losses compared to the other schemes we evaluated. However, when implemented over classical topological protocols like AODV, CAAR only shows significant improvements in low-density scenarios, where path establishment is more difficult and where an AR transaction failure may end up being costly because of the limited number of alternatives to build a path. When implementing GBSR, CAAR obtains the highest values of delay (around 1.8 s instead of 0.8 s for other schemes). The reason is related to the lower packet losses (around 20% fewer losses), which is owing to the absence of AR signalling avoiding collisions, which increases the number of packets using longer paths to arrive at the destination. However, packets coming through long distances could be stored in the routing buffer longer than packets coming through short distances until suitable forwarding nodes are found. Packets coming from far away are, consequently, suffering high delay.

In the case of AODV, the delay does not significantly change among the AR schemes since the AR process occurs mainly during the establishment of the paths. The number of hops is tightly related to the routing protocol behaviour, as shown in the results.

Furthermore, we included a study of the signalling traffic when using the three AR schemes evaluated, in terms of AR transactions, HELLO, RREP and REQ messages. We can conclude that our proposal CAAR improves the packet losses performance without introducing any extra overhead when used over GBSR. CAAR, however, needs less or at most the same amount of signalling traffic as traditional AR in all the vehicle densities we tested, when it is implemented over GBSR.

Future work may be oriented to do a comparative study of the address auto-configuration mechanisms for MANETs surveyed (Bernardos et al., 2010) by using a network simulator. Such study would include an analysis of parameters like average time to obtain a valid configuration, number of auto-configuration attempts, signalling traffic introduced by the mechanism, among others. Also, we are interested in the impact of the IP configuration changes in the performance of routing protocols.

Acknowledgement

We would like to thank Professor Antoni Miñarro for a good number of helpful comments in the use of statistical tests. This work was partly supported by the Spanish Government through projects TEC 2013-47665-C4-1-R Emergency Response In Smart Communities 'EMRISCO' and TEC2010-20572-C02-02 Continuity of Service, Security and QoS for Transportation Systems 'CONSEQUENCE'. Luis Urquiza-Aguiar is the recipient of a full scholarship from the Secretaria Nacional de Educación Superior, Ciencia y Tecnología (SENESCYT) with the sponsorship of Escuela Politécnica Nacional (EPN) (Ecuador). Carolina Tripp-Barba would like to thank the support of the 'Universidad Autónoma de Sinaloa (Mexico)'. D. Rebollo-Monedero is the recipient of a Juan de la Cierva postdoctoral fellowship, JCI-2009-05259, from the Spanish Ministry of Science and Innovation. Ahmad Mezher is recipient of an FI-AGAUR grant

References

Aswathy, M.C. and Tripti, C. (2012) 'An enhancement to AODV protocol for efficient routing in VANET: a cluster based approach', in Wyld, D.C., Zizka, J. and Nagamalai, D. (Eds.): *Advances in Computer Science, Engineering and Applications SE – 99*, Vol. 167 of *Advances in Intelligent Systems and Computing*, Springer Berlin Heidelberg, pp.1027–1032.

Bernardos, C., Calderón, M. and Moustafa, H. (2010) *Internet-Draft: Survey of IP Address Autoconfiguration Mechanisms for MANETs*, Internet Engineering Task Force – Adh-Hoc Network Autoconfiguration pp.1–57.

Braden, R. (1989), *Request for Comments (RFC): 1122 Requirements for Internet Hosts – Communication Layers Status*, Internet Engineering Task Force – Network Working Group pp.1–116.

Broch, J., Maltz, D.A., Johnson, D.B., Hu, Y-C. and Jetcheva, J. (1998) 'A performance comparison of multi-hop wireless ad Hoc network routing protocols, *Proceedings of the 4th Annual ACM/IEEE Int. Conf. on Mobile Computing and Networking', MobiCom '98*, ACM, New York, NY, USA, pp.85–97.

Bychkovsky, V., Hull, B., Miu, A., Balakrishnan, H. and Madden, S. (2006) 'A measurement study of vehicular internet access using in situ Wi-Fi networks', *Proceedings of the 12th Annual International Conference on Mobile Computing and Networking, MobiCom '06*, ACM, New York, NY, USA, pp.50–61.

Caloca, C. and Garcia Macias, J.A. (2010) 'Adaptive solutions in multihop communication protocols for vehicular ad hoc networks;, in Watfa, M. (Ed.): *Advances in Vehicular Ad-Hoc Networks: Developments and Challenges*, IGI Global, chapter Adaptive S, pp.301–322.

Carter, C., Yi, S. and Kravets, R. (2003) 'ARP considered harmful: manycast transactions in ad hoc networks', *Wireless Communications and Networking. WCNC '03 IEEE*, Vol. 3, pp.1801–1806.

Choi, J., Khaled, Y., Tsukada, M. and Ernst, T. (2008) 'IPv6 support for VANET with geographical routing', *2008 8th International Conference on ITS Telecommunications*, IEEE, pp.222–227.

Davies, J. (2012) 'Understanding IPv6', 3rd ed., Microsoft Press, Chapter 6 Neighbor Discovery, p.716.

Estinet-Technologies (2014) *EstiNet Network Simulator and Emulator*, http://www.estinet.com/products.php?lv1=1andsn=2

Ferreiro-Lage, J.A., Gestoso, C.P., Rubinos, O. and Agelet, F.A. (2009) 'Analysis of unicast routing protocols for VANETs', *Fifth International Conference on Networking and Services, 2009. ICNS '09*, Valencia, Spain, pp.518–521.

Fogue, M., Garrido, P., Martinez, F.J., Cano, J-C., Calafate, C.T. and Manzoni, P. (2012) 'A realistic simulation framework for vehicular networks', *Proceedings of the 5th International ICST Conference on Simulation Tools and Techniques, SIMUTOOLS '12*, ICST, Brussels, Belgium, pp.37–46.

Gerla, M. and Kleinrock, L. (2011) 'Vehicular networks and the future of the mobile internet', *Computer Networks*, Vol. 55, No. 2, pp.457–469.

Haberman, B. and Thaler, D. (2002) *Request for Comments (RFC): 3306 Unicast-Prefix-based IPv6 Multicast Addresses*, Internet Engineering Task Force – Network Working Group, pp.1–7.

Haerri, J. and Filali, F. and Bonnet, C. (2006) 'Performance comparison of AODV and OLSR in VANETs urban environments under realistic mobility patterns', *Proceedings of the 5th IFIP Mediterranean Ad-Hoc Networking Workshop*, pp.14–17.

Hartenstein, H., Laberteaux, K. and Ebrary, I. (2010) '*VANET: Vehicular Applications and Inter-Networking Technologies*, Wiley Online Library.

Hinden, R. and Deering, S. (2006) *Request for Comments (RFC): 4291 IP Version 6 Addressing Architecture*, Internet Engineering Task Force – Network Working Group pp.1–22.

Hinden, R. and Haberman, B. (2005) *Request for Comments (RFC): 4193 Unique Local IPv6 Unicast Addresses*, Internet Engineering Task Force – Network Working Group pp.1–14.

Holland, G. and Vaidya, N. (1999) 'Analysis of TCP performance over mobile ad hoc networks', *Proceedings of the 5th Annual ACM/IEEE International Conference on Mobile Computing and Networking, MobiCom '99*, ACM, New York, NY, USA, pp.219–230.

IBM (2014) *SPSS Statistics*, http://www-01.ibm.com/software/analytics/spss/.

IEEE Standard for Information technology–Telecommunications and Information Exchange Between Systems Local and Metropolitan Area Networks–Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (2012).

IEEE Standard for Wireless Access in Vehicular Environments (WAVE) 00 Networking Services (2010).

Imadali, S., Petrescu, A. and Janneteau, C. (2013) *Internet-Draft: Vehicle Identification Number-Based Unique Local IPv6 Unicast Addresses (VULA)*, Internet Engineering Task Force – Network Working Group pp.1–17.

Karp, B. and Kung, H.T. (2000) 'GPSR Greedy perimeter stateless routing for wireless networks', *Proceedings of the 6th annual international conference on Mobile Computing and Networking – MobiCom '00*, ACM Press, New York, New York, USA, pp.243–254.

Krajzewicz, D., Erdmann, J., Behrisch, M. and Bieker, L. (2012) 'Recent development and applications of SUMO - Simulation of Urban MObility', *International Journal On Advances in Systems and Measurements*, Vol. 5, Nos. 3 and 4, pp.128–138.

Krauss, S., Wagner, P. and Gawron, C. (1997) 'Metastable states in a microscopic model of traffic flow', *Phys. Rev. E*, Vol. 55, No. 5, pp.5597–5602.

Lee, K., Lee, U. and Gerla, M. (2009) 'Survey of routing protocols in vehicular ad hoc networks', *Advances in Vehicular Ad-Hoc Networks: Developments and Challenges, Information Science Reference*, pp.149–170.

Mohammad, S., Rasheed, A. and Qayyum, A. (2011) 'VANET Architectures and protocol stacks: a survey, in Strang, T., Festag, A., Vinel, A., Mehmood, R., Rico Garcia, C. and Röckl, M. (Eds.): *Communication Technologies for Vehicles SE – 9*, Vol. 6596 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp.95–105.

Morgan, Y.L. (2010) 'Notes on DSRC and WAVE standards suite: its architecture, design, and Characteristics', *IEEE Communications Surveys and Tutorials*, Vol. 12, No. 4, pp.504–518.

Narten, T., Nordmark, E., Soliman, H. and Simpson, W. (2007) *Request for Comments (RFC): 4861 Neighbor Discovery for IP version 6 (IPv6)*, Internet Engineering Task Force – Network Working Group, pp.1–97.

Naumov, V., Baumann, R. and Gross, T. (2006) 'An evaluation of inter-vehicle ad hoc networks based on realistic vehicular traces', *Proceedings of the seventh ACM International Symposium on Mobile Ad Hoc Networking and Computing – MobiHoc '06*, pp.108–119.

Open Street Maps (2014), http://www.openstreetmap.org/

Perkins, C., Belding-Royer, E. and Das, S. (2003) *Request For Comments: 3561 Ad hoc On-Demand Distance Vector (AODV) Routing*, nternet Engineering Task Force - Network Working Group pp.1–13.

Plummer, D. (1982) *Request For Comments: 826 An Ethernet Address Resolution Protocol*', Internet Engineering Task Force – Network Working Group p.11.

Rappaport, T.S. (2001) '*Wireless Communications: Principles and Practice*, 2nd ed., Prentice-Hall, Upper Saddle River, NJ, USA.

Sheskin, D. (2000) '*Handbook of Parametric and Nonparametric Statistical Procedures*, 2nd ed., Chapman and Hall/CRC, Boca Raton.

Sommer, C., Eckhoff, D., German, R. and Dressler, F. (2011) 'A computationally inexpensive empirical model of ieee 802.11p radio shadowing in urban environments', *Wireless On-Demand Network Systems and Services (WONS)*, pp.84–90.

Spaho, E., Ikeda, M., Barolli, L. and Xhafa, F. (2013) 'Performance comparison of OLSR and AODV protocols in a VANET crossroad scenario', in Park, J.J.J.H., Barolli, L., Xhafa, F. and Jeong, H-Y. (Eds.): *Information Technology Convergence SE – 5*, Vol. 253 of *Lecture Notes in Electrical Engineering*, Springer Netherlands, pp.37–45.

Tripp Barba, C., Urquiza Aguiar, L. and Aguilar Igartua, M. (2013) 'Design and evaluation of GBSR-B, an improvement of GPSR for VANETs', *IEEE Latin America Transactions*, Vol. 11, No. 4, pp.1083–1089.

Wakikawa, R., Malinen, J., Perkins, C. and Tuominen, A. (2006), *Internet-Draft: Global connectivity for IPv6 Mobile Ad Hoc Networks*, Internet Engineering Task Force – Mobile Ad Hoc Networking Working Group, pp.1–20.