

VANBA: a simple handover mechanism for transparent, always-on V2V communications

José Antonio Olivera, Ismael Cortázar, Carolina Pinart, Alberto Los Santos, Iván Lequerica

Abstract— This paper presents VANBA, a simple, ready-to-use mechanism to perform vertical handovers in vehicular networks that integrates the essential functionalities of the Always Best Connected (ABC) paradigm in a transparent way both to the user and the upper-layer applications even in terminals with limited processing capabilities. This mechanism operates at the network layer and achieves quasi-continuous communication during the handover process by monitoring the reachability of the destination node. A basic handover scenario using VANBA has been implemented at Telefónica I+D’s labs on Linux-based carPC and laptops enabled with WiFi and 3G as communication media. This scenario, which emulates an in-car dual-network terminal that uses 3G as backup communications for WiFi, is used to compare the main features of VANBA with respect to other handover solutions being proposed in emerging standards.

Index Terms— Vertical Handover, VANET, Always Best Connected, vehicular networks.

I. INTRODUCTION

In the last years, Intelligent Transport Systems (ITS) and traffic telematics have drawn much attention both from academia and industry. There is a growing number of consortia and projects are related to vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communications. The major standards in the field, such as IEEE 802.11p (WAVE) or ISO’s Communications Architecture for Land Mobile environment (CALM), are advancing. Such inter-vehicular communications will form vehicular networks, composed of mobile nodes (vehicles) which will form a communication infrastructure that shall provide always-on connectivity. Always Best Connected (ABC) solutions are emerging in order to provide the best communication channel available to the mobile nodes in these vehicular networks. ABC refers to being not only always connected, but also being connected in the best possible way, i.e., combining several communications technologies [1]. The definition of ‘best’ depends on a number of aspects such as personal preferences, limitations of the device (terminal), application requirements, security, policies, or network resources and coverage.

The ABC philosophy involves vertical handovers (VHO), that is, seamless switching between two network technologies maintaining the same connection (e.g., a TCP-connection). Generically, a VHO encompasses the search for a second communication network to switch to, the transmission of information about the terminal to this second network, as well as of temporary channel information of the second network to the terminal. Also, the control of a temporary channel connection between the terminal and the second network is

necessary. The main initiatives in this area that can be applied to the vehicular environment are the Media Independent Handover (MIH or IEEE 802.21) [2] and the CALM architecture [3]. These emerging standards, described in Section II, are defining the elements and functionalities necessary for VHOs. While MIH is focused on the VHO, CALM involves the functionalities necessary to support transparent, continuous communications for vehicular users.

To the best of our knowledge, these initiatives are focused on developing new architectures to provide ABC connectivity, which are in general based on complex methods, combining large number of modules. In contrast, we propose simple VHO mechanism that provides the essential functionalities of ABC (transparent, always-on connectivity) and applies the term ‘best’ to the limitations of the terminal requiring the VHO. Our proposed mechanism, called VANET Backup Communications (VANBA), relies on simple components, and on a simple interface with the applications to perform VHOs. VANBA’s simplicity is its major strength, and it also makes it a ready-to-use alternative to MIH and CALM for VHOs in low-processing terminals. The remainder of this paper is organized as follows. In Section II we overview the two emerging standards for VHOs in vehicular networks – MIH and CALM – and the R&D initiatives that are carrying out and/or evaluating them. Section III describes the proposed VHO mechanism (VANBA) in terms of architecture, elements and implementation. In Section IV we provide and discuss an implementation example of VANBA for two communication media (WiFi and 3G). Finally, in Section V we discuss the advantages and shortcomings of VANBA with respect to the emerging MIH and CALM approaches for VHO in V2V.

II. VERTICAL HANDOVERS IN VEHICULAR NETWORKS

There are two main standardization initiatives in charge of specifying vertical handovers: the 802.21 Working Group for Media Independent Handover and Interoperability of the Institute of Electrical and Electronics Engineers (IEEE) [2], which applies to generic mobile/wireless networks, and the CALM standards’ suite [3], which is specific to the vehicular environment. Through CALM, the International Organization for Standardization (ISO) is defining a standard framework for V2V and V2I communications. In parallel, the IEEE 802.21 Group is responsible for the specification of MIH, which provides handover mechanisms across the same (horizontal) or different networks (vertical).

Handing off connections is a key function of CALM. The ISO 21210 standard [4] specifies the network protocols for

media management and transfer within CALM. Fig. 1 illustrates the scheme (protocol) to support the handover of a session using the same or different CALM media. The CALM management entities select the best link available for each application taking into account the application requirements, which have been previously set. The available media capabilities are compared to the application requirements by merging both capabilities and requirements under predefined media selection policies. Note that, if needed, this mechanism changes the routing table.

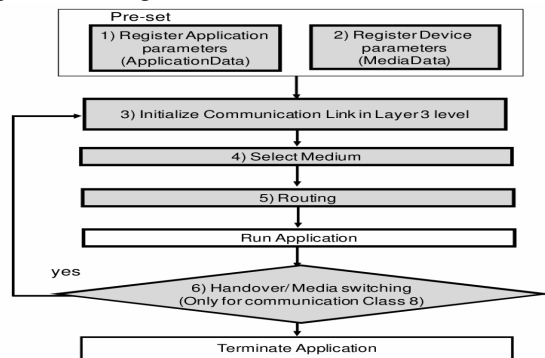


Fig 1.CALM handover scheme

Besides ISO’s CALM Working Group, the four major consortia in the field of V2V communications are the Car to Car Communications Consortium (C2C-CC) [5] in Europe, and the Vehicle Safety Communications Consortium (VSCC) and Vehicle Infrastructure Integration (VII) [6] in the US. There are also a number of projects that are cooperating with, integrating and/or validating the CALM architecture. The most relevant ones are the Japanese CALM Proof-of-concept, the European Cooperative Vehicle-Infrastructure Systems (CVIS) [7] and the European Co-operative Systems for Road Safety (SAFESPOT) [8] projects. It is worth noting that CVIS’ vehicular communication infrastructure is implemented on the basis of the ISO CALM standards. CVIS integrates a specific interface for application management, which includes Quality of Service (QoS) requirements and media selection policies. Such policies are used for coupling the QoS requirements and available communication media to decide the best medium for each application and handover situation.

The IEEE 802.21 Working Group is focused on standardizing an interface for interoperability and handover in heterogeneous networks, that is, MIH [2]. MIH relies on an intermediary layer between layers 2 and 3 of the Open Systems Interconnection (OSI) stack, which is commonly referred to as ‘layer 2.5’. This layer is in charge of converting communication-system-specific commands (from layer 2) into events used by the network selector to trigger a handover. This selector takes the handover requirements from the applications, considers the status of all the different available interfaces (media) and determines the best medium to support the communication. By itself, this layer neither controls the handover nor manages the mobility, but it provides a complete overview of the available media and their radio conditions.

There are three types of VHOs: those controlled by the terminal, those initiated by the terminal and assisted by the

network, and those initiated and controlled by the network [2]. Fig. 2 illustrates a terminal-initiated handover according to MIH, which is also known as mobile-initiated MIH handover. If a handover is needed, the terminal, or Mobile Node (MN), obtains information about neighboring networks (potential secondary networks, or Candidate Points of Service [PoS]) by querying a predefined Information Server. Then, the MN requests the handover initiation to its Serving PoS, which sends the handover preparation request messages to the Candidate PoSs to query the availability of the requested resources in the secondary networks. Upon response by the Candidate PoSs, the Serving PoS notifies the MN about the handover initiation result, and the MN chooses the secondary network to switch to. This handover decision is based on the information about resource availability sent by the Serving PoS. Finally, the MN performs the handover by exchanging MIH_Handover_Commit messages with its Serving PoS.

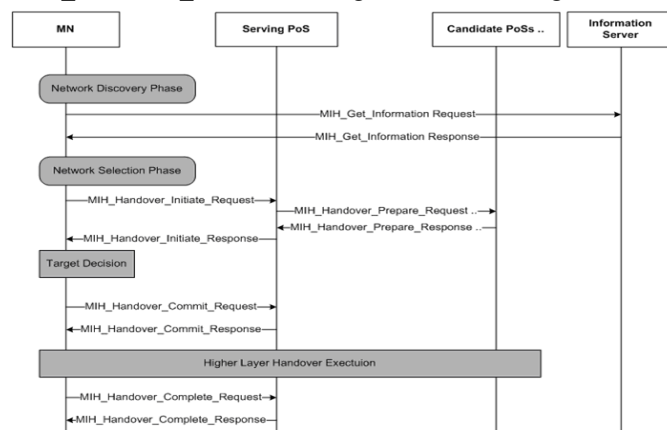


Figure 2.Terminal-initiated MIH handover procedure

In the literature we may find several handover solutions based on MIH. For example, the vehicular communication system in [9] integrates the MIH function (MIHF) to provide ABC communications. The media considered in [9], which include IEEE 802.3, IEEE 802.11, HiperLan/2, IEEE 802.16, 2/2.5G and 3G, are set under MIH’s ‘layer 2.5’. The Mobility Management and Network Selection functions are built on top of the MIH layer in order to switch among the above-listed media according to the events received from the MIH. In [10], the authors implement the IEEE 802.21 framework [2] to test the MIHF in different handover scenarios with MNs.

III. VANBA: VANET BACK-UP COMMUNICATIONS

We propose VANBA as a simple alternative to the MIH and CALM VHO mechanisms described in Section II. VANBA’s simplicity lies in the fact that it provides quasi-continuous communications among vehicle On-Board Units (OBU) with minor software modifications and ‘burden’ in the OBUs, while keeping disruptions unnoticed by applications.

A. Framework and assumptions

Generically, an OBU is defined as a terminal installed in a vehicle that implements vehicular communication protocols and algorithm/s and that is equipped with at least: i) a short-range wireless network interface operating at dedicated

frequencies, and ii) a wireless or wired network interface to which application unit/s (AU) can be attached. In this work, we assume that the OBU is enabled with at least two different communication media (CM) so that a VHO can be executed between them. Examples of CM technologies are WiFi (IEEE 802.11b/g/p), cellular (2/2.5/3G) or WiMAX.

VANBA falls within the scope of terminal-initiated VHOs, in which the OBU initiates the handover process. On the other hand, VHOs in MIH [2] and CALM [4] are based on the behavior of the CMs at layer 2 (data link) of the OSI stack, e.g., they rely on state change events, predictive events or network initiated events. Since VANBA operates at layer 3 (network), it does not need to alter the behavior of layer 2. Instead, it defines layer-3 VHO triggers based on the IP reachability of the destination.

As specified in [4], the selection of the best secondary network (secondary CM) for a VHO shall be done by matching both the capabilities of the available CMs in the OBU and the communication requirement/s of the applications running on the AU/s for which a VHO is required, based on predefined CM selection policies, e.g. minimum bandwidth, and cost. VANBA considers a single CM selection policy: the CM for V2V communications (Vehicular Ad-hoc Network, VANET) is defined as default for initiating communication between the AU and any given destination. The remaining CMs available in the OBU are defined as secondary or backup CMs. The rule to prioritize the secondary CMs in VANBA combines cost (in euros) and bandwidth (in bit/s).

B. Architecture

As depicted in Fig. 3, VANBA complies with the OSI reference model; it is composed of two components: one at layer 3 (network layer) and one at layers 6-7, namely the VHO process (depicted as ‘VANBA VHO’ in Fig. 3) and the link monitoring process (depicted as ‘VANBAD’ in Fig. 3). Note that in MIH [2] and CALM [4], the VHO functions are embedded mainly in layer 2. This happens because these standards ‘break’ the OSI reference model in their cross-layer interactions to perform VHOs.

VANBA’s VHO process is in charge of performing VHOs between the default and the secondary CMs, as we will describe in Section III.C. VANBAD is a daemon process in charge of layer-3 monitoring of the established link between a source-destination pair over a given CM. This monitoring is used to trigger the VHO process. Monitoring entails layer-3 CM link characteristics, from basic (i.e., IP reachability) to service-quality-related (e.g. considering the application’s requirements). Finally, VANBAD performs an additional function: if the status of the running (default) CM changes to ‘inactive’ (e.g., destination unreachable, or link quality below the application’s QoS threshold), which will lead to a VHO, VANBAD monitors when this CM becomes ‘active’ again to triggers a new VHO initiation back to using the default CM.

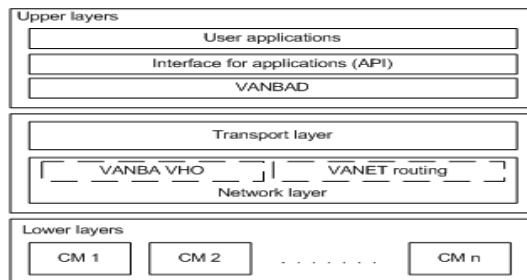


Figure 3. Architecture of the vehicle’s OBU

C. VANBA’s handover mechanism

VANBA triggers a VHO initiation process whenever the default CM (i.e., VANET) is not able to support the communication requirements from the running application/s or if VANET connectivity is lost (“destination unreachable”). This VHO initiation process looks for the target secondary CM that is able to support the communication requirements from the application, according to VANBA’s CM selection policy. Then, VANBA completes the handover execution process and transfers the connection to the target secondary CM. This process is illustrated in Fig. 4.

Fig. 4 depicts the parallel flowcharts of the application (AU) and VANBA (OBU) in the event of a VHO. Initially, the application must register the communication requirements (bandwidth and cost) that shall be satisfied by the OBU between the source (the AU) and any given destination. VANBA must also register the CM parameters (i.e., bandwidth and cost) for all available CMs and associate the default CM for initiating any communication with the destination previously registered, which will be monitored by VANBA through the VANBAD process (see Section III.B). This registry process is marked in Fig. 4 as ‘Pre-set’. The application process depicted in Fig. 4 is standard-procedure; whenever an application must send or receive information to/from a destination, it opens a socket and transmits the information in packets (i.e., TCP or UDP). If it detects a problem in the transmission or reception of packets, it applies a suitable policy, such as re-sending or buffering.

VANBA’s handover mechanism is composed of seven steps:

1. VANBA monitors the IP reachability of the destination
2. Whenever a “destination unreachable” or link quality below the application’s QoS threshold is detected, VANBA triggers a VHO initiation process, which consists of the following steps:
3. A secondary CM is selected according to VANBA’s CM selection policy. In parallel, the packets being transmitted by the application are buffered in a transparent way with respect to the application.
4. The connection is transferred to the secondary CM.
5. If the VHO process is successfully completed, VANBA goes to step 1 and starts monitoring the link established over the secondary CM.
6. Else, the next secondary CM in the prioritized list is attempted, and the VHO process goes back to step 3.
7. If no secondary CM is available after attempting as many CMs as the buffer length or as the application requirements allows (e.g., in case of latency-dependence), the VHO cannot take place and hence the packets that were buffered in step 3 must be dropped.

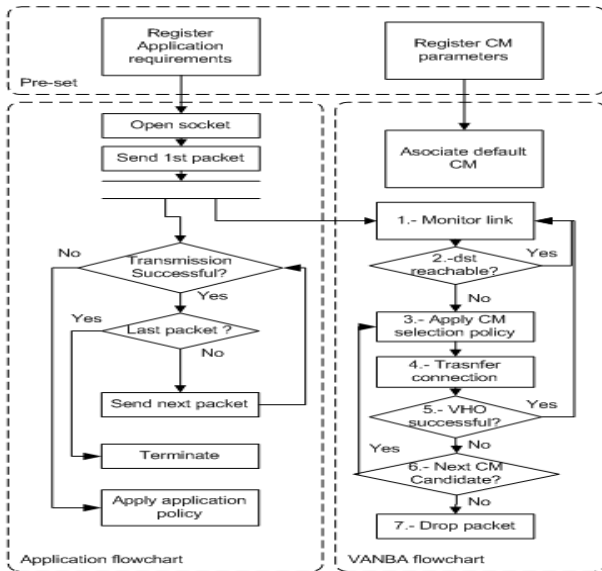


Figure 4. VANBA's vertical handover scheme

D. Interaction with the AU

CALM-aware applications shall define their communication requirements before starting any communication; this way, CALM works transparently for achieving always-on communications according to the application's requirements [4]. In any case, both CALM-aware and other types of applications shall be agnostic about the underlying network stack so that VHOs are transparent to them. Within VANBA, this is achieved by two elements:

- The Application Programming Interface (API): this API is established between the AU and the OBU in the 'Pre-set' phase of the VHO (see Fig. 4), in which applications register their communication requirements.
- The socket-based transmission of information: this way, applications are agnostic about the underlying network stack (layer 1 to 3) because they can continue using the well-known BSD socket API at upper layers. In parallel, VANBA's VHO will work transparently at layer 3 with the TCP or UDP packets (see Fig. 3).

IV. VANBA IMPLEMENTATION EXAMPLE

There are several initiatives close to the market that support V2V communications, e.g., CALM's M5 interface [3] or C2C-CC's IEEE 802.11p WiFi interface [5], also known as Wireless Access for the Vehicular Environment (WAVE). On the other hand, many cars are enabled with cellular communications since the 1990s to enjoy telematics services. The number of such 'connected' cars will grow dramatically in Europe thanks to the future eCall standard [11], which requires 2/2.5G connectivity. That is, in the near future we will see cars embedded with WiFi and cellular-capable OBUs. Here is where a solution like VANBA will prove useful.

As defined by the C2C-CC [5], the default CM for V2V communications is IEEE 802.11p (WAVE). Whenever the WAVE CM is unavailable, for example due to low density in the VANET network, VANBA switches to 3G as a backup CM for V2V communications. As an example, Fig. 5 depicts a V2V communication over three cars (A, B and C). Each car is embedded with a VANBA-enabled OBU that features WiFi

(IEEE 802.11b/g) and 2.5/3G CMs. Note that here we use 802.11b/g instead of 802.11p due to the commercial unavailability of WAVE network cards, and because 802.11b/g has proven to be applicable for V2V communications in some scenarios, e.g., [12].

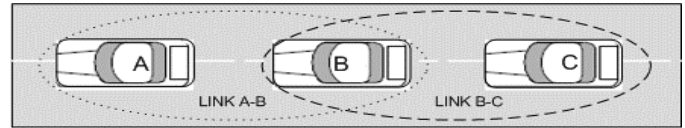


Figure 5. V2V communication example



Figure 6. Prototype implementation of VANBA for car A (DRIVE testbed)

Fig. 6 illustrates the VANBA prototypes implemented at Telefónica I+D's laboratories. For car A, the OBU is implemented in a carPC with Open Embedded Linux. The AU is implemented in the same carPC, and runs an ITS messaging application. This implementation is part of Telefónica I+D's "Demonstrator for Intelligent Vehicular Environments" (DRIVE) testbed [13]. For cars B and C, the OBU is implemented on Debian Linux laptops. Fig. 7 shows the implementation for cars B and C, as well as the V2V services mentioned above. In both cases, the default CM is implemented through a IEEE 802.11b/g Ubiquiti Xtreme Range 2 600 mW network card, while the secondary or backup CM is implemented through a Novatel 3G modem, featuring High Speed Downlink Packet Access. For cars A to C, VANBA VHO was integrated into the OBU's Linux operating system as an in-kernel component, and it featured a network packet filter over Netfilter for packet handling.

VANBAD was implemented as a user-space daemon that raises a layer-3 'destination reachable' event if TCP/UDP packets reach their destination through the established CM. Else, the CM is flagged as 'inactive' until it comes back to activity. The implemented applications (e.g., video streaming) use the Linux IPv4/IPv6 socket communication interface. Finally, we implemented a management interface to access VANBA-specific management functionalities such as registering the applications requirements (i.e., API):

```

vanba_mgmt
+register_app_requirements( src:struct sockaddr *,
                           dst:struct sockaddr *,
                           req: struct app_req* ): int

```

V. DISCUSSION OF VANBA'S FUNCTIONALITIES

VANBA's main advantage with respect to MIH and CALM is simplicity, which is discussed here including an analytical

performance comparison in terms of VHO delay, and outline how VANBA can provide quasi-continuous communication.

A. Simplicity

VANBA's simplicity comes in three flavours: processing, terminal-independence and integration. VANBA results in little additional *processing* burden in the terminal because the VHO mechanism is only focused on the layer-3 events, and in its simplest implementation scenario, it only needs to monitor IP reachability of the destination. That is, VANBA does not require control functions at layer 2 or layer 2.5, as CALM and MIH require. Within VANBA, the VHO is only processed upon the reception of the first destination unreachable ICMP packet over the CM in use, which results in low handover delay in the carPC and laptop implementations (Figs. 6 and 7).

Moreover, VANBA is *terminal-independent* for any device with GNU/Linux operating system and suitable CMs. Note that VANBA VHO is an in-kernel component and VANBAD is a user-space component. Concerning *integration*, CMs are just plugged on the OBU for VANBA to detect them and include them in its CM list. Hence, VANBA's behaviour is much simpler than CALM's [14], which requires a terminal framework additional to the VHO for management and communications functions. VANBA's architecture requires only three software components to support VHOs (see Fig. 3) – VANBA VHO, VANBAD and the VANBA-applications API – while CALM requires several components: CME, NME and IME for media management [4], and medium service access points for communications [14]. In turn, MIH also requires several functions [2], as outlined in Section II.

We can compare the performance of VANBA with respect to the MIH and CALM approaches in terms of latency (T). T is defined as the time elapsed between the down and up trespassing of the QoS threshold in the communication, and is directly related to the service disruption time. Assuming a given OBU for the realization of VHO with CALM, MIH and VANBA, the processing latency at the OBU to restore the IP connectivity (T_{ns}) is equal for all the approaches. Although the latency to detect the need for a VHO (T_{hd}) depends on the parameters required to do so, being a local OBU process, we assume that differences are negligible. The network selection delay (T_{ns}) involves local processing in all the approaches, ($T_{ns-proc.latency}$), and also transmission and 'remote' processing delays ($T_{ns-TX-latency}$) in MIH, which are a function of the number of candidate PoS queried by the serving PoS (Fig. 2). Finally, MIH VHOs have two additional phases (Fig. 2), which we split in network discovery (T_{nd}) and target decision (T_{td}) delays. Taking the above into account, T can be expressed as follows:

$$T_{CALM} = T_{hd} + T_{ns-CALM} + T_{ns-sw} ; T_{VANBA} = T_{hd} + T_{ns-VANBA} + T_{ns-sw}$$

and $T_{MIH} = T_{hd} + T_{nd} + T_{ns-MIH} + T_{td} + T_{ns-sw}$, where

$$T_{ns-MIH} = T_{ns-MIH-proc-latency} + T_{ns-MIH-TX-latency} ;$$

$$T_{ns-VANBA} = T_{ns-VANBA-proc-latency}(M) ; T_{ns-CALM} = T_{ns-CALM-proc-latency}(N) ;$$

$1 \leq M \leq 2$ (Section III) and $N \geq 2$ [3, 4], where N and M are the medium selection policies. From these expressions we can infer that $T_{MIH} > T_{ns-CALM} \geq T_{ns-VANBA}$.

B. Quasi-continuous communications

Transparent always-on communications is a requirement for any solution to VHOs in generic mobile/wireless networks and vehicular networks. Like CALM and MIH, VANBA also supports these features through providing an application API (see Section III) and quasi-continuous communications. To illustrate the latter, cars A and C in Fig. 5 share an ITS messaging application. They are connected through a VANET using WiFi as the default CM. If the B-C link is broken, VANBA is used as the VHO mechanism to switch to the backup 3G CM so that the ITS messaging application does not notice the VHO. We performed several qualitative tests of this scenario and found that VANBA provides quasi-continuous communications, that is, the VHO is sufficiently rapid and the packet buffering accurate enough so that the messaging (Fig. 4 right) does not notice the network switch. Note that one of the keys to the short VHO delay is the fact that VANBA does not require handover initiation and preparation processes, as opposed to CALM and MIH (see Figs. 1 and 2).

On the down side, VANBA only considers layer-3 network connectivity at layer 3. This might result in QoS violations, e.g., when IP reachability exists but the packet loss rate is higher than the QoS threshold for this or layer-1/2 performance parameters. Besides, VANBA does not consider non-IP based communications, e.g., using geographical based networking.

ACKNOWLEDGMENT

This work was supported by Telefónica and by the Castilla y León Regional Government (project SCV-ADE 2007).

REFERENCES

- [1] E. Gustafsson y A. Jonsson, "Always best connected," Wireless Communications, IEEE, Vol. 10, pp. 49-55, 2003.
- [2] IEEE Draft Standard P802.21/D13, "Local and Metropolitan Area Networks: Media Independent Handover Services," 2008.
- [3] ISO/DIS 21217, "Intelligent transport systems-Communications access for land mobiles (CALM)-Architecture", 2006.
- [4] ISO/DIS 21210, "Intelligent transport systems-Communications access for land mobiles (CALM)-Networking Protocols", 2006.
- [5] C2C-CC official site, <http://www.car-2-car.org/>.
- [6] VII official site, <http://www.vehicle-infrastructure.org/>.
- [7] CVIS official site, <http://www.cvisproject.org/>.
- [8] SAFESPOT official site, <http://www.safespot-eu.org/pages/page.php>.
- [9] E. Weiss *et al.*, "Architecture of an Always Best Connected Vehicular Communication Gateway," in Proc. 64th VTC (Fall), 2006
- [10] J.-H. Seol *et al.*, "IEEE 802.21 MIH based handover for next generation mobile communication systems", in Proc. 4th IIT conference, 2007.
- [11] eCall Driving Group, "Recommendations of the DG eCall for the introduction of the pan-European eCall", 2006.
- [12] V. González *et al.*, "Experimental demonstration of the viability of IEEE 802.11b based inter-vehicle communications," in Proc. TRIDENTCOM 2008. Innsbruck (Austria), 2008.
- [13] C. Pinart *et al.*, "DRIVE: a reconfigurable testbed for advanced vehicular services and communications," in Proc. TRIDENTCOM 2008.
- [14] ISO Standard 21218, "Intelligent transport systems-Communications access for land mobiles (CALM)-Medium service access points", 2006.