

Agnostic and Modular Architecture for the Development of Cooperative ITS Applications

Bruno Dias, A. Santos, A. Costa, B. Ribeiro, F. Gonçalves, J. Macedo, M. João Nicolau, O. Gama and S. Sousa

Abstract—A Vehicular Adhoc Network (VANET) is a generic conceptualisation that can be applied to the communications domain of an Intelligent Transportation Systems (ITS). It defines requirements that allow the exchange of information between applications and services running on equipment that form a VANET, mainly moving vehicles, fixed road infrastructure's systems and mobile personal devices. In this unique and demanding environment, interoperability is attained through the use of specific wireless communications technologies and protocols. The great heterogeneity on the capabilities of the interconnected devices, the scale of the number of devices that could be involved, the various degrees of density and the complexity on the mobility patterns, makes interoperability at the application level a challenging problem. This paper presents an agnostic VANET architecture, adapted from ETSI and ISO modern standards, to be deployed on ITS as a mean to overcome those limitations. This new approach fosters the use of different communication technologies and network/transport protocol stacks in an open and modular framework, facilitating the creation of ITS cooperative applications and services, using traditional network programming paradigms where applications use the available medium transparently. The usefulness and versatility of the proposed architecture have been illustrated by means of two different applications that have been taken from simulation scenarios into use cases, really deployed on top of commercially available On Board Units.

Index Terms—Intelligent Transportation Systems; Vehicular Adhoc Networks; Agnostic Communications Architecture.

I. INTRODUCTION

The increasing mobility of people is having a relevant economic and social impact. The growing number of moving vehicles and pedestrians, mainly in the urban environment, leads to an increased number of traffic jams, to higher levels of pollution and consumption of energy. Traffic accidents have been taking thousands of lives each year and caused critical injuries to millions of persons[1], although vehicles, as standalone machines, are getting safer, more efficient, more comfortable and providing an overall better user experience with increasing levels of automation. To further improve global safety, enhance efficiency and bring automation to a higher level, it is necessary to create and deploy new cooperative mechanisms, applications and services. New approaches on

communication networks should be pursued so that vehicles, road infrastructures, personal smart devices and other type of mobile equipments are able to communicate and exchange useful informations such as traffic conditions, safety warnings and infotainment data. Cooperative Intelligent Transportation Systems (C-ITS) deal with these scenarios making use of specific communications frameworks to support interoperability, thus many research efforts have been directed to this C-ITS application domain in the latest years.

In VANETs, Vehicular Ad-Hoc Networks, vehicles communicate directly when they are within radio signal range, or using multi-hop paths between source and destination nodes, computed in a cooperative way using specific routing protocols. Since vehicles move at very high speeds along restricted and predictable road paths, VANETs have distinctive characteristics and communication requirements; short contact time, connectivity disruption, packet losses, frequent topology changes and frequent high channel loads, are examples of the characteristics that make VANETs a very challenging communication environment. Vehicles use On-Board Units (OBUs) in order to communicate with each other in the lack of any infrastructure support, but also with fixed Road Side Units (RSUs) on the road infrastructures. VANETs may use Vehicle to Vehicle (V2V), Vehicle to Infrastructure (V2I) and Vehicle to Pedestrian (V2P) interactions, or, more generically, Vehicle to X (V2X) communications, and nodes communicate directly in broadcast mode, if in radio range, or use multi-hop paths. Since vehicles move at high speeds along restricted and predictable road paths, VANETs have distinctive characteristics and challenging communication requirements when in dense (eg urban) or sparse environments. RSUs may play an important role in VANETs, since they facilitate communication between vehicles when direct communication is not possible, and enables vehicles to connect to the backbone network where C-ITS services support is easier to implement. The distribution of RSUs depends on the communication protocol or use case - some environments require an even distribution over the whole network, while others require RSUs only at intersections or a reduced distribution set in a given region.

Nowadays, it is common for vehicles to be equipped with several internal communication capabilities, materialised in multiple and various wired and wireless interfaces. Wired communication technologies such as CAN [2], LIN [3] and FlexRay [4] tend to coexist well in vehicle environments, although the tendency is for manufacturers to start implementing Ethernet-based solutions, due to their benefits in terms of

Manuscript received May 2, 2018; revised August 23, 2018. Date of publication September 17, 2018. Prof. Nikola Rožić has been coordinating the review of this manuscript and approved it for publication.

Authors are with the University of Minho, Portugal (e-mails: {bruno.dias, alex, costa, macedo}@di.uminho.pt, {b7214, b7207, b2583, b7215}@algoritmi.uminho.pt, joao@dsi.uminho.pt).

Digital Object Identifier (DOI): 10.24138/jcomss.v14i3.550

complexity, cost or even cabling weight. Wireless technologies like WiFi, ZigBee, Bluetooth [5] or LiFi [6] are considered for other types of connectivity inside the vehicle, like connecting the vehicle to the driver or passenger's smart devices. As for external communications, by now only prototype vehicles tend to include this type of capabilities and the IEEE 802.11p [7] is the most popular technology for wireless V2X connectivity. When applications require longer range communications (more than one kilometer) or higher bandwidths (which could be common for infotainment services), it may be necessary to resort to other solutions such as cellular networks or WiMax [8].

Several standardisation institutions around the world have defined deployment architectures to model various aspects of an ITS. These standards instantiate communication models defined for VANETs into existing standard technologies and protocols. They take into account real world implementation scenarios faced by (and also forced by) the automotive industry, which impose additional technological, social, legal and security constraints when deploying communications architectures. So, the most common approach when developing ITS stations and services is to define a closed solution inside an OBU, conforming to standard specific access technologies for low level communications, a network and transport protocol stack compatible with this type of environment, and an application software complying with broader functional requirements for some well known VANET applications. Thus, each manufacturer relies on an adapted proprietary architecture that actually renders the interoperability between applications by different manufacturers hard to achieve, if not impossible. Furthermore, in this closed development context, it is very difficult for software makers to implement C-ITS applications to compete in the automotive market independently of the manufacturers. All software is developed and deployed as an entire block, from the user interface in the applications down to the low level medium access technologies and everything in the middle.

Vehicular applications can be classified into three groups according to their major goal: (i) safety, (ii) efficiency and management, and (iii) comfort/infotainment. According to its type, applications may interact with the vehicle and with passengers in different ways. The Safety domain refers to applications or systems that increase the protection of the people inside the vehicle as well as the vehicle itself; efficiency domain refers to improving and managing the traffic flow, and this group of applications is of great interest, since traffic congestion is becoming an increasingly severe problem; finally, infotainment and advanced driver assistance provide entertainment or information to drivers and passengers.

Given the wide range of existing (and upcoming) applications, one of the most critical tasks is to standardize the diverse communication requirements for each type of application. In this paper we describe a new approach on communications architectures for C-ITS, firstly introduced in [9] by the same authors, that intends to open up the development process of C-ITS applications and services to a larger community of software makers while keeping safety and security on all critical ITS environments. The underlying

strategy is to adapt existing communications models defined by the most important standardisation institutions into a more modular architecture that could be more easily adopted by the manufacturers and still be open to integration of independent application level software. This strategy is more supportive of research and development processes of C-ITS applications in a modular way, following diverse types of methodologies, from the traditional TCP/IP client-server paradigm to more recent ones like Named Data Networks (NDN) [10].

This paper complements and extends the works [9] by presenting in Section II a comprehensive and detailed analysis of standardised architectures currently used in ITS, justifying the need for an agnostic approach, leading to real C-ITS application integration. In Section III the proposed architecture, and each one of its main modules and associated functions, are described with special emphasis on the ITS-LCI interface that exposes the vehicles' available services to any third party application developers. The architecture enables services availability to developers, without compromising any security or safety constraint imposed by the manufactures. Section V presents the deployment of two completely new case studies - a C-ITS Platooning Management Controller and the establishment of a hybrid testbed, enabling the integrating of real hardware with VEINS vehiculars' development framework; both use cases comply with the new architecture and real implementations have been derived and tested with existing commercial OBU hardware. Finally, Section VI presents the conclusions and planned future works.

II. STANDARD VANET ARCHITECTURES

Vehicular networks are heterogeneous. In the future, and regarding V2X communications, vehicles may be equipped with a diversity of wireless communication interfaces. Probably, the most common will be based on IEEE 802.11p, which is a special version of the IEEE 802.11 medium access standard [7] for vehicular environments, or based on cellular networks. This technology is the heart of the Wireless Access in Vehicular Environment (WAVE) [11] specification, as it can be seen on Figure 1, that was adopted by the Society of Automotive Engineers (SAE) for the Dedicated Short Range Communications (DSRC) [12][13][14] family of standards and by the European Telecommunications Standards Institute (ETSI) for the ITS-G5 standards [15].

Considering the diversity in the vehicular environment, to allow a rapid and incremental development of VANET applications and technologies, it is desirable that the software on the application level be independent to the underlying medium-access communications framework. Furthermore, if possible, the same principle should be applied to the transport and network levels. To attain this objective, the concept of an agnostic middle-ware communications layer could be introduced. This should provide independent information management for multiple data sources, allowing heterogeneous communication interfaces to be transparently supported over different communications stacks. Notice that this insight into independency of applications from the underlying frameworks and communication technologies is also a requirement for

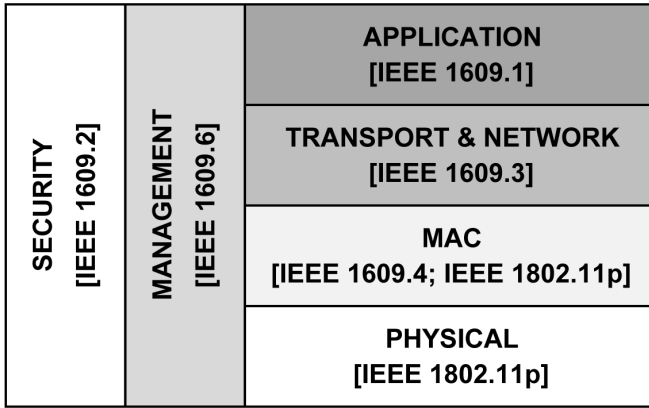


Figure 1: WAVE Communications Stack

the future of vehicular Internet [16]. Also, plenty of the research efforts involving ITS application development, either for safety applications for intelligent vehicle control in urban road intersections [17] or even more specific security approaches to Internet based applications to be used in vehicular networking [18], although dependent from the characteristics of the underlying ITS architecture, should not depend on any specific implementation. So, this type of agnostic middle-ware communications layer should give freedom to vehicular applications developers and communications stacks providers to focus on the optimization of its components or modules. This middle-ware layer must be implemented and supported by the vehicle's OBU that will be able to choose the best interface or use multiple interfaces in parallel to communicate with other OBUs or RSUs, requiring the simultaneous use of different protocols at the lower-level communication layers.

Indeed, an effort to provide such a type of middle-ware has already been made and included on existing standard architectures. That solution tries to mask the diversity and distribution of the vehicular resources and to provide common services to different V2X applications using a unified application programming model. Unfortunately, this approach relies on the requirement that all applications must use directly the same middle-ware layer between the applications and the transport layer. So, it must be integrated on all communication stacks on the ITS station and coordinated by some common lower-level layer inside the OBU through some type of inter-layer signalling, which is a very taunting task. Also, with this strategy, management of application multiplexing between different communications stacks is difficult to implement. In general, to overcome these problems, manufacturers tend to present complete closed solutions that support only one communications stack and which higher-level applications are incompatible with other manufacturers applications.

A. CALM

Communications Access for Land Mobiles (CALM) is a communications architecture defined by the ISO Technical Committee 204 - Working Group 16 (TC204 WG16). This model, as depicted on Figure 2, is designed to allow interoperability between C-ITS stations and it abstracts applications

and services from the underlying communication layers. The set of CALM standards include specifications for: ITS Station Management; Communications Security; Facilities, Station Networking and Transport layer protocols; Communication Interfaces and Services; Interfacing Technologies into ITS Stations; Distributed Implementations of ITS Stations; Interfacing ITS Stations to existing communication networks.

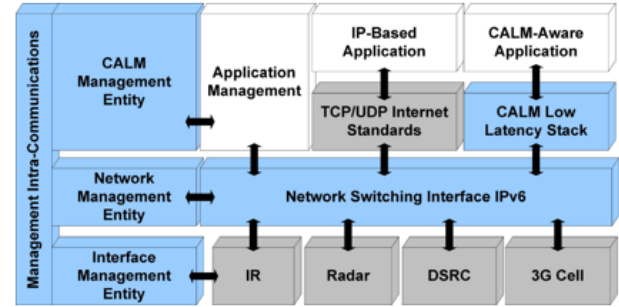


Figure 2: CALM Architecture (from [19])

Its main goal is to establish an integrated technology that is able to provide a set of protocols and parameters for C-ITS communications using several different media [20]. Theoretically, the CALM architecture should enable vehicles to use any kind of communication media with seamless media handover. As such, CALM supports the integration of different wireless technologies for V2X communications but does not present detailed solutions for their simultaneous use on the same ITS station by several applications, although it's implied that the challenge could be overcome implementing network switching mechanisms inside the OBU.

B. ETSI ITS-G5

The ISO 21217:2014 standard [21] describes the ITS station reference architecture, represented on Figure 3, consisting on six parts: Applications, Management, Facilities, Networking & Transport, Access and Security. Although this standard specifies all the ITS station elements, whether or not a particular element is implemented is dependent on the specific communications requirements.

Until now, the most relevant middle-ware support solution for application agnosticism on this architecture was introduced by the standard ETSI ITS-G5 Facilities Layer [15]. More recently, some other alternatives were also proposed [22] [23] [24] [25], but they can be seen as forms of instantiation of the generic ETSI proposal, adding a set of specific service (information, communication and application) modules between the application and the transport layer. An implementation of such a facilities middle-ware might be a very complex task and software makers would lose the ability to choose a development paradigm for their applications, at least without any type of signalling mechanism supported by the entire communications stack. Application development would have to take into account many technological details of lower-level layers (announcements, notifications, channel attribution and prioritisation, etc.). Also, the usage of a particular

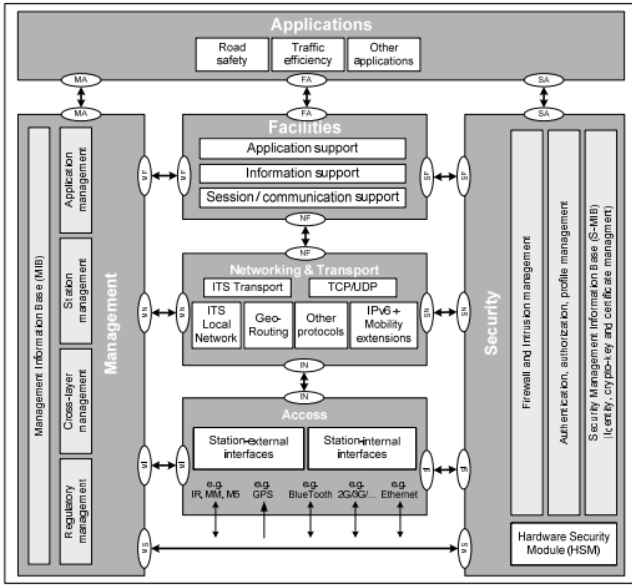


Figure 3: ETSI ITS-G5 Architecture (from [15])

transport layer would be bound to the usage of a particular medium access technology. This favours the development and deployment of non-modular solutions with problematic interoperability issues between implementations of different automotive manufacturers of the same VANET cooperative application or service.

III. AGNOSTIC ITS COMMUNICATIONS ARCHITECTURE: A NEW APPROACH ON STANDARDS

The development environment for in-vehicle applications should support the industry trend of clearly separating (hardware and software wise) the services that are responsibility of the vehicle manufacturer and the rest of the services and applications that can be added by third party developers. It is most likely that the former services will be developed and implemented on closed OBUs and installed by the manufacturers (or third party specialists under the responsibility of the vehicle manufacturers) during the vehicle manufacturing process. The latter, which would implement higher-level C-ITS applications and services, could be developed by software developers independently of the vehicle manufacturer and installed during the vehicle manufacturing process or later on by the vehicle owner. This third party software could be run on additional computational resources either inside the vehicle or at some fixed RSU. So, a new approach on an agnostic ITS architecture could adapt the ETSI ITS-G5 proposal to solve the referred constraints and make use of the full potential of future multi-medium access capabilities and without the integration of a facilities layer directly on top of the transport layer.

An ITS communications architecture for the development of agnostic vehicular applications should present the OBU modelled like a black-box, implementing all lower-level in-vehicle communication services, information services and function services. These service modules would share the same interface technologies and the implemented functionalities

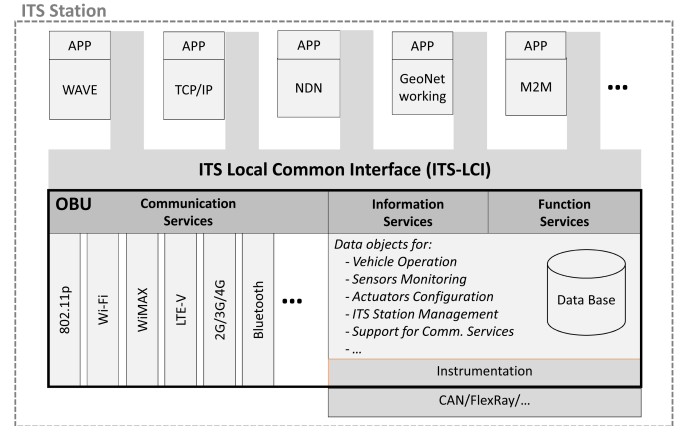


Figure 4: Agnostic Architecture for VANETs

would, at least, cover the same functionalities referred by the ETSI ITS-G5 facilities layer. As such, the proposed agnostic ITS architecture, as presented in Figure 4, would support all types of vehicular applications developed using any type of programming paradigm and taking full potential of a multi-medium access capable OBU.

The main component of this ITS station architecture is the OBU, implemented as a black-box, that is, it does not matter how the OBU developer implements the internals. Furthermore, due to the vehicle's industry safety and security requirements, these internals should be physically and logically protected from external direct manipulation that could lead to tampering the functional behaviour originally intended by the original developer or manufacturer. Every existing vehicle should be able to assume the role of sender, receiver and router and/or switch (depending on which protocol layer the forwarding process is implemented), so that information can be disseminated in the network while assuring a safe and smooth traffic flow at the same time. To enable V2X communications, vehicles must be equipped with, at least, one OBU that, typically, should provide, as minimum, a CAN bus, a power supply and some wireless communications technology like WAVE, Bluetooth or LTE.

For this new modular approach, the OBU should provide a group of services implementing all common features needed for access to vehicle internal data sources (information services), V2X communication mechanisms (communication services) and lower-level automotive functions (function services). The interface to these services, identified as ITS-Local Communication Interface (ITS-LCI), should be available through widely used standard technologies that can easily be supported by all automotive manufacturers implementing these new OBUs. The set of technologies defined for this interface would be used by software applications implemented on the ITS station outside the OBU (or ITS station host). This usage could be direct, when the applications themselves explicitly use the ITS-LCI interface to obtain access to the OBU services, or indirect, when the applications have indirect access to the OBU services by means of the underlying operating system. The former method should be preferred when accessing information or function services (which are, in nature, higher-level

application functionalities) and the later one, which would be transparent to the applications, should be favoured when accessing communication services, although this mandates that the operating systems where the ITS applications are developed and deployed implements the needed support for ITS-LCI interface.

It should be emphasised that this solution does not use classic network routing or lower-level switching inside the ITS station to attain the intended features and functionalities as these mechanisms are difficult to pursue for nodes on a VANET. Nor it uses a complex facilities layer in between the applications layer and the transport layer of each communications stack on each operating system deployed on computational resources outside the OBU. Again, all interactions with the three services modules implemented internally on the OBU should be done through the ITS-LCI using standard communications protocols and access technologies.

A. Communication Services Module

This interface module will permit sharing of all medium-access technologies supported by the OBU by all application environments in the ITS station and deployed on resources outside the OBU, or ITS hosts; it should implement adequate algorithms for multi-homing (the simultaneous use of several network providers on the same medium-access technology) and vertical hand-off (the simultaneous use of several medium-access technologies by the same application stack); this module should also implement all related functions to medium-access addressing, monitoring, parametrisation and security, and also antennas configuration, when available. In general, due to the specific nature of the low-level functional requirements, including hardware requirements, of the medium-access technologies, their internal implementations will be done on a dedicated Central Control Unit (CCU). Direct access to communication functions by external processes (outside the OBU) should not be allowed. Instead, indirect access should only be possible through the communication services module (which is a logical module with higher-level functionalities, implemented on top of the available CCU functions) and using standard communication technologies. This module acts as a point of access to the medium technologies. This means that, whenever a packet of data is generated at the network level on any of the systems outside the OBU, regardless of the type of application stack used, the packet is sent to the medium using the best-performing communication technology at the moment. Although Figure 4 only shows internal blocks/channels for medium-access technologies, external logical blocks/channels, implemented on devices outside the OBU, could be registered and used by the module, as long as the manufacturer's defined interface is supported. The communication services module requires a bidirectional adaptation, that is, whenever a Protocol Data Unit (PDU) arrives through one of the communication blocks/channels, its type must be identified so it can be correctly forwarded through the ITS-LCI to the adequate application stack at the network layer. Another requirement for this module is that it must implement efficient scheduling strategies and management of communications channels. It

should also support diverse communication paradigms like unicast, geocast, multicast and broadcast. All of these features should be implemented inside the OBU and supported transparently to external ITS applications (direct access) and operating systems (indirect access).

B. Information Services Module

This interface module will permit access and manipulation of data generated by all sensors, actuators and other devices in the vehicle, indirectly or directly connected to the OBU; this data includes, for example, sampled values from Electronic Control Units (ECU) connected to an internal Controller Area Network (CAN) of a vehicle, like speed, acceleration, engine regime, water and oil temperature, fuel and battery levels, distance to obstacles in the front and in the rear, tires pressure, vehicle longitudinal and traversal inclination or rotation, etc.). It could also include information about configuration parameters of the supported V2X medium-access communication technologies (like the amount of power applied to antennas, date and time, etc.), sampled values from ambient sensors on the vehicle (like interior and exterior temperature and humidity, altitude, etc.), video streams or photos images from video or image devices installed on the vehicle, relative and global position (using GPS, RF-ID, etc.), vehicle identification, malfunctions, biometric values of the driver, security parameters and information on relevant event dates (mandatory inspections, manufacturer check-ups, etc.). It should be noted that the data should be collected and made available at rates and with a precision adequate for proper use on the most demanding applications, some of them almost in real-time. As such, and dependent on the running ITS applications and on the type of the data itself, the sampling processes should be configurable (rate, precision, amount of sampled values retained, maximum retention time, etc.), including their security levels of access. Standard technologies for abstract definition and secure remote manipulation of all this data should be used. Alternatively, in the particular case of remote manipulation of OBU configuration parameters, it could also be available through configuration functions implemented on the function services module as support for direct manipulation through the information services module may be more complex and more difficult to ensure its proper and secured use.

C. Function Services Module

This interface module will permit access to lower-level functionality procedures. These are functions that the manufacturers, due to security, safety, performance and liability issues, should have the responsibility and the desire to implement (or closely control its implementation). These functions should implement atomic operational procedures as these should not be interfered or use sub-procedures external to the OBU. Some examples include manoeuvres to maintain a certain distance from front obstacle during a certain time, to maintain a certain velocity or acceleration during a certain time, to maintain or change lane, etc. It is also possible to support configuration procedures like setting the driving mode to sport, setting lighting mode to automatic, resetting

all malfunctions information, locking all doors, powering on emergency lights, setting interior temperature to a certain level, turn Bluetooth communications on, etc. From the ITS application development perspective, it is not important to know how the OBU internal instrumentation will implement these atomic procedures. It is only relevant to know the syntax and semantics of the particular functions needed to implement a concrete ITS application or service. This modular approach is very useful to implement external applications with higher-levels of functionality, like Platooning or Autonomous Driving.

D. ITS-Local Communication Interface (ITS-LCI)

This architecture is truly universal and modular, as it allows the integration of any protocol stack, such as DSRC, TCP/IP or NDN, implemented on any operating system of the ITS host. It is also adequate for development and deployment of all types of ITS applications and services (safety, efficiency, comfort, management, etc.). The applications and operating systems of the ITS hosts should be able to access any of the services modules through the ITS-LCI, either in direct or indirect mode. The ITS-LCI represents a standard bidirectional communication technology (or group of technologies, protocols and mechanisms), easily supported by any OBU as the context of its implementation and deployment is locally restricted to the vehicle, which makes its technical requirements less demanding (for example, no classic internal network routing or naming service is necessary). Direct access to the OBU services using application-level communications technologies is simpler to implement but is less transparent so this method should be favoured for access to the information services and function services modules. As for access to the communications service, indirect access through the operating systems communications protocol stack should be preferred because with this method it is easier to treat at the operating system level some complex lower-level mechanisms needed for efficient management of the available resources for V2X communications. At this mode, the ITS-LCI interface could be seen as a link-layer communications tunnel; another ITS-LCI application level interface is based on the SNMP management protocol and is used to access data and to configure operational parameters available. Also, MIB definitions convey the needed abstraction paradigm for an independent representation of the syntax and semantics of the data.

IV. AGNOSTIC ITS COMMUNICATIONS ARCHITECTURE: DEPLOYMENT SOLUTIONS

It seems appropriate, even critical, to implement a database, or repository, on the OBU to support the data access and manipulation requirements of the three services modules. Such strategy requires an adequate availability of computational power and energy supply that manufacturers should take into consideration. An instrumentation mechanism would need to be implemented in order to adapt the database technology to the available data access/manipulation technologies already available to vehicle manufacturers. Nevertheless, since neither the data nor the instrumentation mechanism can be directly accessed outside the OBU, their implementation strategy and

used technologies can be freely chosen by the OBU developer, as long as the standard technologies defined for the interface with the services modules are supported. Applications should also be able to store relevant information on this repository, mainly data that is important to share with other applications, like information from safety applications or sample values from external user devices. Nevertheless, this procedure should only be supported through the standard interface technologies supported by the information services module.

In this regard, a widely used database technology like a Management Information Base (MIB) of the Internet Simple Network Management Protocol (SNMP) [26] framework seems adequate as the core mechanism for implementation of the information services module and also as a complementary technology for the other two services modules (mainly for manipulation of configuration parameters or monitoring the execution of functions inside the OBU).

The ITS-LCI should use a large bandwidth, low cost medium wired technology like Gigabit Ethernet, support a simple communications stack like TCP/IP and a widely used application management protocol like SNMP. The SNMP framework seems to be a good solution to implement the interface technology to be supported by the three services modules for access and manipulation of the referred data. As stated before, its use would be complemented by the definition of a standard automotive OBU Management Information Base (MIB) implemented on an SNMP agent inside the OBU. Furthermore, the MIB objects could convey enough semantic meaning as to provide support for the interface with the function services module.

A more performing communication technology based on UDP encapsulation over IP could be used. A simple asynchronous and connectionless application protocol with primitives for direct access to the communication services module should be defined and supported by the OBU and the ITS applications and services deployed on ITS hosts. This ITS-LCI application protocol could also include generic primitives for encapsulation of a mechanism for direct access to procedures provided by the function services module.

V. AGNOSTIC ITS COMMUNICATIONS ARCHITECTURE: TWO CASE STUDIES

As a means of proving both the versatility and usefulness of this agnostic ITS communications architecture two application scenarios have been used as case studies: i) a hybrid ITS development, integrating real communication hardware (a generic OBU hardware, allowing for real-time WAVE / 802.11p communications) with the open source vehicular network simulation framework VEINS [27] (Vehicles in Network Simulation); ii) testing a Platooning Management Controller application.

A. Hybrid ITS Application Development

Firstly a hybrid ITS application development, able to integrate an ITS application, established within the simulation platform VEINS, with real-time WAVE / IEEE 802.11p communications by means of two OBU boards, real hardware prototypes.

A Cooperative Traffic Efficiency application has been developed and tested on a CAR2X Hybrid Simulation Platform, composed by two OBU prototype hardware boards interconnected with the VEINS Simulator framework, using SUMO as the interface for vehicular mobility. Within the simulation setup, two of the simulated vehicles, represented in the simulator with different colours, as presented in Figure 5, are to be permanently connected to real OBU hardware boards in order to enable sending/receiving all WAVE / 802.11p messages using real time communications. Simulator nodes interact with the OBU hardware boards using a simple API as interface, the ITS-LCI component depicted in Figure 6, also identified as "Simplified ITS-LCI communication API" in Figure 5.

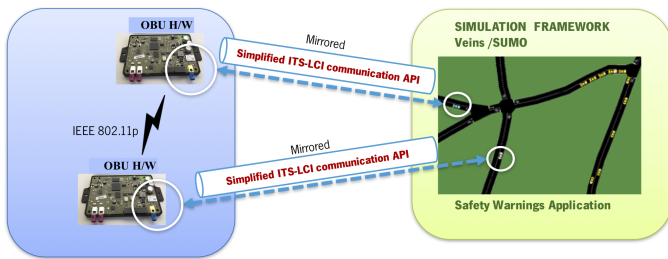


Figure 5: Hybrid testbed integrating VEINS framework with OBU boards

Whenever a message is to be sent in the VEINS simulation by one of the two vehicles connected to real OBU boards, the message is sent simultaneously in the simulator and, via the ITS-LCI interface, in real radio channel. The OBU prototype board receives the message and broadcasts it by means of real WAVE / IEEE 802.11p communications. Then, any other OBU board in radio range receives the message and sends it, again by means of the ITS-LCI interface, to the VEINS simulator. Whenever the VEINS simulator receives a message through the ITS-LCI interface, the virtual node associated with box handles it immediately. The node may receive the same message twice, once via the OBU board and another time from the simulated channel; the decision on which message to keep (or discard) is handled by a special decision algorithm¹ implemented within VEINS. In order to enable this hybrid environment, the agnostic ITS communications architecture had to be taken into the real OBU boards that were running an operating system image prepared previously, integrating the agnostic software developed, with ITS-LCI, and flashed into the boards. The ITS-LCI daemon within the OBU boards was deployed on top of a fully working V2X stack and this daemon listens for requests coming from external applications.

This case study demonstrates the integration in VEINS simulation platform of the ITS-LCI interface, proving with success the ability of using real time IEEE 802.11p communications between the OBU boards that are associated with vehicles whose movement is determined by the application (and SUMO) developed and tested in the simulated environment.

¹This decision algorithm deals with the analysis of vehicle node reachability, using the vehicle position in the simulated environment.

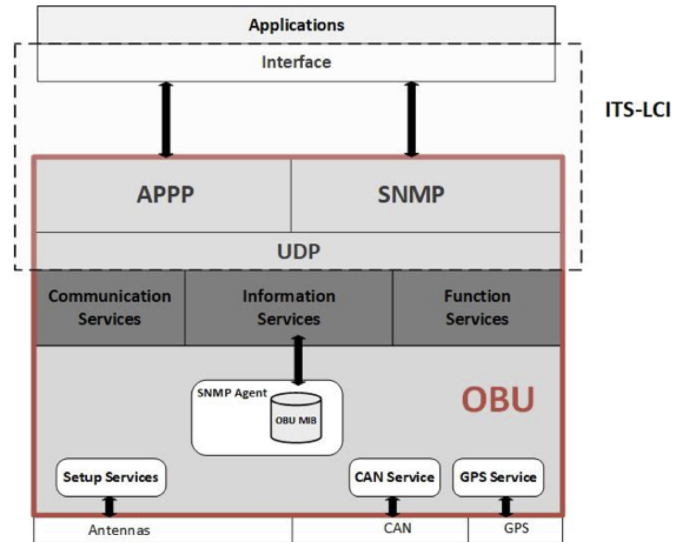


Figure 6: ITS-LCI support for a Platooning Application

B. Platooning Management Controller as an ITS Application example

Platooning is a well known C-ITS application that, as depicted in Figure 7, allows vehicles, normally trucks, to travel very close to each other with, at least, automated longitudinal distance control, although automated lateral distance control between them is also an objective. In theory, this collaborative application should enhance safety, traffic flow, road capacity, fuel savings and enhance reduction of harmful emissions. In practice, due to its complexity and tight safety requirements, there are still no commercially available solutions.



Figure 7: Platooning of trucks

This is a good example of a type of application that could benefit from a research, development and deployment paradigm as defined by the this new approach.

Figure 8 represents the platooning application deployment framework to be implemented on the architecture and that could be developed in a modular process. It includes a platooning management protocol implemented at the C-ITS application level and a set of standard supporting lower-level vehicle control functions implemented on the OBU, which are already implemented in many vehicles as standalone automotive features or as manoeuvres integrated on high-level procedures. Higher-level functionalities and security requisites could be implemented on an ITS host system with a common Linux operating system, albeit with an adaptation for transparent interaction with the OBU communication services module using link layer tunnel on the ITS-LCI. An even simpler alternative could be implemented using only direct access to all services modules of the OBU. In this case, as depicted in Figure 6, the ITS-LCI would support a simple application-level connectionless communications protocol encapsulated over an UDP/IPv4 protocol stack for direct access to the

communication services and function services modules and SNMP over the same UDP/IPv4 stack for direct access to the function services and information services module.

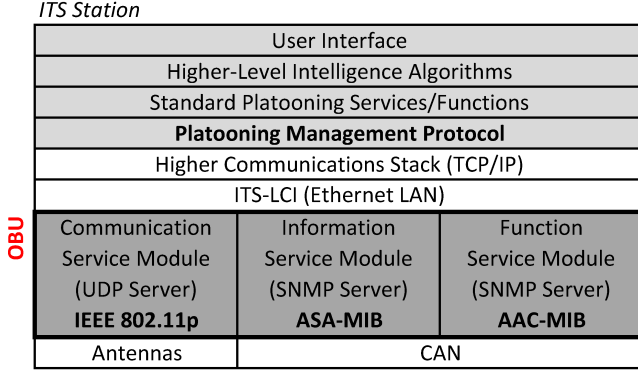


Figure 8: Platooning deployment model

The APPP communication interface may use `protobuf` or `ASN.1` [28] in order to specify and encode the PDUs and in the platooning application as it is an independent and well established technology and widely used for definition of V2x communication protocol standards. For external communication with the platooning host application, the SNMP protocol² has been used and a simple SNMPv2c communications module has also been implemented inside OBU.

The higher-level modules of the platooning application should take care of the algorithms for implementation of the more complex operations like drivers certificate verification, joining and leaving, overtaking, merge, leader substitution, etc. Standard lower-level procedures provided by the function services module of the OBU should implement basic functional requirements like atomic manoeuvres (like maintaining longitudinal distance to front, maintaining speed, changing lane, ID renumbering, time synchronisation, etc.) or configuration of platooning parameters (setting reference distance between vehicles, setting reference lane, setting reference lateral alignment, etc.). The platooning management protocol would include a set of dedicated service primitives and protocol data units for their encapsulation. All interactions between vehicles in the platoon should be performed using this protocol which could also be implemented on top of a simple UDP/IPv4 communications stack.

The vehicle manufacturer would be responsible for the correct, safe and secure implementation of the services modules inside the OBU. For example, it could use the WAVE framework for V2V communications and two MIB modules implemented on an SNMP agent/server. For example, the Automotive Sensors & Actuators MIB (ASA-MIB) module could be dedicated to monitoring vehicle sensors data needed for platooning, like speed, acceleration, global positioning (GPS) and relative positioning (through radar and video camera technologies) and the Automotive Automatic Controllers MIB (AAC-MIB) module would be used to access/configure

²Although this version of the SNMP protocol has no security features, that is not relevant in the context of this prototype implementation

lower-level control functions like Cooperative Adaptive Cruise Control, Automatic Steering Control and Cooperative Adaptive Emergency Braking.

The Platooning application was previously developed in simulation framework VSimRTI [29], and tested in simulation only scenarios [30] while the agnostic communications interface software was being designed and tested. When the first OBU board prototypes became available for testing, the Platooning Management Controller was also deployed in a hybrid testbed, integrating the simulation with the real time communications hardware. The main goal was to be able to test the Platooning Management Controller closer to a real prototyping scenario, able to include hardware boards associated with two or more virtual vehicle objects in the simulator. The access to the vehicles' real time data and the communication interface with the commercial OBU boards have been accomplished using the ITS-LCI communication layer and its implementation is presented in Figure 9.

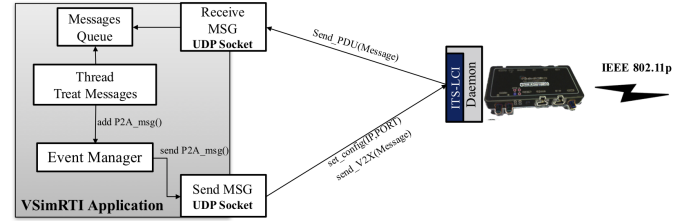


Figure 9: Hybrid Scenario for Platooning deployment

This new architectural approach has already been applied with a simulated implementation of such a platooning management protocol [30] using the V2X Simulation Run-time Infrastructure (VSimRTI) [29], a simulation framework able to couple and integrate other well known simulators, such as SUMO and Network Simulator (version 3) on top of a common infrastructure.

Finally, this C-ITS application has been successfully tested with a car following application, where a car leader (establishing the acceleration and velocity) commands and determines the adaptive real time behaviour of the car that is following.

VI. CONCLUSIONS AND FUTURE WORK

Current standard ITS communications architectures are mainly aimed at the development of applications in closed ecosystems, in general controlled by automotive manufacturers or manufacturers consortia. There is no separation of development ecosystems (and its responsibilities) on smaller isolated components, modules or protocol stacks (such as applications on one side and OBUs on the other, for example). This generates less opportunities for research and development independently of the automobile industry and turns difficult the development of third party solutions.

This paper presented a new approach for the deployment of a communications architecture for VANETs, using the service paradigm. The proposed architecture is modular and open, as it allows the integration of any application protocol stack or network paradigm. The applications and communication protocols should be able to use any of the services

modules - Communications Services, Information Services and Function Services - by means of the ITS-LCI. Standard and common communication technologies and protocols are used to implement the ITS-LCI, which enables access to the service modules on the OBU. Keeping the lower-level medium-access communication and lower-level vehicle control functions inside the OBU, we can maintain the responsibility of its implementation on the automotive industry because its correct implementation is a delicate issue with both legal and social repercussions.

Using commercial CCU, made available by the automotive industry, a simplified version of the presented architecture has already been deployed and tested, including an ITS-LCI implementation that uses connectionless UDP/IPv4 to interface the OBU Communication Services and a SNMPv2c/IPv4 primitives to interface with the OBU Information and Function Services.

Several modules of this Agnostic ITS Communications Architecture have been deployed into commercially available OBUs and these prototypes were tested with two C-ITS applications. The paper presents two case study applications: i) a hybrid ITS platform, integrating real OBU boards with the open source vehicular network simulation framework VEINS; ii) a Platooning Management Controller application, that have been tested on top of different OBU boards.

Tests have been performed with success in hybrid setups; also, some C-ITS application developments have been carried out and some of these prototypes were used on real vehicles, albeit on private and controlled road scenarios.

As future steps, a simple HMI application is to be deployed on Android platforms, able to support the ITS-LCI communication module, in order to allow other external applications to interface with real OBU boards. Another research direction is the development, evaluation and testing of a strong security model to be used in the context of this Agnostic ITS Communications Architecture.

ACKNOWLEDGMENT

This work has been partially sponsored by the European Structural and Investment Funds in the FEDER component, through the Operational Competitiveness and Internationalization Programme (COMPETE 2020) Project n. 002797, Funding Reference: POCI-01-0247-FEDER-002797, and also by COMPETE: POCI-01-0145-FEDER-007043 and FCT - Fundação para a Ciência e Tecnologia within the Project Scope: UID/CEC/00319/2013.

REFERENCES

- [1] World Health Organization, "Global Status Report On Road Safety 2015," World Health Organization, Tech. Rep., 2015.
- [2] I. S. Organization, *ISO 11898-1:2015 - Road vehicles - Controller Area Network (CAN) - Part 1: Data link layer and physical signalling*, ISO Std., 2015.
- [3] —, *ISO 17987-1:2016 - Road vehicles - Local Interconnect Network (LIN) - Part 1: General information and use case definition*, ISO Std., 2016.
- [4] BMW Ag, Daimler Chrysler A. G., *FlexRay Communications System Protocol Specification Version 2.1 Revision A*, F. Consortium, Ed. Daimler Chrysler A. G., Dec. 2005.
- [5] J. Lee, Y. Su, and C. Shen, "A comparative study of wireless protocols: Bluetooth, uwb, zigbee, and wi-fi," in *IECON 2007 - 33rd Annual Conference of the IEEE Industrial Electronics Society*, Nov 2007, pp. 46–51. [Online]. Available: <http://doi.org/10.1109/IECON.2007.4460126>
- [6] H. Haas, L. Yin, Y. Wang, and C. Chen, "What is lifi?" *Journal of Lightwave Technology*, vol. 34, no. 6, pp. 1533–1544, March 2016. [Online]. Available: <http://doi.org/10.1109/JLT.2015.2510021>
- [7] IEEE, *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, Amendment 6: wireless access in vehicular environments*, IEEE Std., 2010.
- [8] A. Ghosh, D. R. Wolter, J. G. Andrews, and R. Chen, "Broadband wireless access with wimax/802.16: current performance benchmarks and future potential," *IEEE Communications Magazine*, vol. 43, no. 2, pp. 129–136, Feb 2005. [Online]. Available: <http://doi.org/10.1109/MCOM.2005.1391513>
- [9] S. Sousa, A. Santos, A. Costa, B. Dias, B. Ribeiro, F. Gonçalves, J. Macedo, M. J. Nicolau, and Óscar Gama, "A new approach on communications architectures for intelligent transportation systems," *Procedia Computer Science*, vol. 110, pp. 320 – 327, 2017, 14th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2017) / 12th International Conference on Future Networks and Communications (FNC 2017). [Online]. Available: <https://doi.org/10.1016/j.procs.2017.06.101>
- [10] L. Zhang, D. Estrin, J. Burke, V. Jacobson, J. D Thornton, D. Smetters, B. Zhang, G. Tsudik, K. Claffy, D. Krioukov, D. Massey, C. Papadopoulos, T. Abdelzaher, L. Wang, P. Crowley, and E. Yeh, "Named Data Networking (NDN) Project," "", 05 2012.
- [11] IEEE 1609 Working Group, *IEEE Guide for Wireless Access in Vehicular Environments (WAVE) - Architecture*, IEEE Std., 2014.
- [12] CEN EN 12253: *Road transport and traffic telematics - Dedicated Short Range, Communication (DSRC) - Physical layer using microwave at 5,8 GHz*, European Committee for Standardization Std., 2004.
- [13] CEN EN 12834: *Road transport and traffic telematics - Dedicated Short Range Communication (DSRC) - DSRC application layer*, European Committee for Standardization Std., 2003.
- [14] CEN EN 12795: *Road transport and traffic telematics - Dedicated Short Range, Communication (DSRC) - DSRC data link layer: medium access and logical link control*, European Committee for Standardization Std., 2003.
- [15] ETSI EN 302 665 V1.1.1 (2010-09): *Intelligent Transport Systems (ITS); Communications Architecture*, ETSI Std., 2010.
- [16] M. Gerla and L. Kleinrock, "Vehicular networks and the future of the mobile internet," *Computer Networks*, vol. 55, no. 2, pp. 457–469, 2011.
- [17] M. Cano, R. Sanchez-Iborra, B. Freire-Viteri, A. Garcia-Sanchez, F. Garcia-Sanchez, and J. Garcia-Haro, "A self-adaptive approach for traffic lights control in an urban network," in *2017 19th International Conference on Transparent Optical Networks (ICTON)*, July 2017, pp. 1–4. [Online]. Available: <https://doi.org/10.1109/ICTON.2017.8025051>
- [18] F. Gonçalves, B. Ribeiro, O. Gama, S. Barros, S. Sousa, V. Hapanchak, A. Santos, A. Costa, B. Dias, J. Macedo, and M. J. Nicolau, "Hybrid model for secure communications and identity management in vehicular ad hoc networks," in *2017 9th International Congress on Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT)*, Nov 2017, pp. 414–422. [Online]. Available: <https://doi.org/10.1109/ICUMT.2017.8255170>
- [19] T. Willke, P. Tientrakool, and N. Maxemchuk, "A survey of inter-vehicle communication protocols and their applications," *IEEE Communications Surveys and Tutorials*, vol. 11, no. 2, pp. 3–20, 2009. [Online]. Available: <http://doi.org/10.1109/SURV.2009.090202>
- [20] K. Dar, M. Bakhouya, J. Gaber, M. Wack, and P. Lorenz, "Wireless Communication Technologies for ITS Applications (Topics in Automotive Networking)," *Communications Magazine, IEEE*, vol. 48, no. 5, pp. 156–162, 2010.
- [21] ISO Technical Committee 204 WG16, *ISO 21217:2014, Intelligent Transport Systems - Communications Access for Land Mobiles (CALM) - Architecture*, ISO Std., 2014.
- [22] S. Nour, R. Negru, F. Xhafa, F. Pop, C. Dobre, and V. Cristea, "Middleware for Data Sensing and Processing in VANETs," in *Proceedings - 2011 International Conference on Emerging Intelligent Data and Web Technologies, EIDWT 2011*. IEEE, 2011, pp. 42–48. [Online]. Available: <http://doi.org/10.1109/EIDWT.2011.33>
- [23] R. Costa, "VadM U a common api for agnostic application development within vanets," in *Proceedings ELMAR-2013*, Sept 2013, pp. 365–368.

- [24] F. A. Silva, T. R. M. B. Silva, L. B. Ruiz, and A. A. F. Loureiro, "ConProVA: A smart context provisioning middleware for VANET applications," in *IEEE Vehicular Technology Conference*, IEEE, 2013, pp. 1–5. [Online]. Available: <http://doi.org/10.1109/VTCSpring.2013.6692488>
- [25] I. Jawhar, N. Mohamed, and H. Usmani, "An overview of inter-vehicular communication systems, protocols and middleware," *Journal of Networks*, vol. 8, no. 12, pp. 2749–2761, 2013. [Online]. Available: <http://doi.org/10.4304/jnw.8.12.2749-2761>
- [26] D. Harrington, R. Presuhn, and B. Wijnen, *RFC3411 - An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks*, IETF Std., 2002.
- [27] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011. [Online]. Available: <http://doi.org/10.1109/TMC.2010.133>
- [28] ITU-T, *ISO/IEC 8824-1 - ITU-T Recommendation X.680, Information technology - Abstract Syntax Notation One (ASN.1): Specification of basic notation*, ITU-T Std., 2015.
- [29] B. Schünemann, "V2X simulation runtime infrastructure VSimRTI: An assessment tool to design smart traffic management systems," *Comput. Netw.*, vol. 55, no. 14, pp. 3189–3198, Oct. 2011. [Online]. Available: <http://dx.doi.org/10.1016/j.comnet.2011.05.005>
- [30] B. Ribeiro, F. Gonçalves, A. Santos, M. J. Nicolau, B. Dias, J. Macedo, and A. Costa, "Simulation and testing of a platooning management protocol implementation," in *Wired/Wireless Internet Communications: 15th IFIP WG 6.2 International Conference, WWIC 2017, St. Petersburg, Russia, June 21–23, 2017, Proceedings*, Y. Koucheryavy, L. Mamatas, I. Matta, A. Ometov, and P. Papadimitriou, Eds. Springer International Publishing, 2017, pp. 174–185. [Online]. Available: https://doi.org/10.1007/978-3-319-61382-6_14



ests focus on Network and Services Management Technologies and Protocols, Applied Multimedia Technologies and Cooperative ITS/VANET Communications Architectures and Applications.



several IEEE, ACM and other international conferences. He is IEEE Senior Member within IEEE Computer Society and IEEE Communications Society.



the same university

Bruno Dias was graduated in Systems and Informatics Engineering by University of Minho, Portugal, in 1991, and obtained a PhD Degree (2005), in Informatics (Computer Communications) in the same University. He has been with the University of Minho as a lecturing and research staff member, since 1991. During his career in the University of Minho, he has lectured or researched in Operating Systems, Computer Networks, Data Communication Systems, Network Management and Multimedia Technologies and Services. Current research inter-

Alexandre J. T. Santos is Associate Professor in the Informatics Department, Engineering School, University of Minho. He received his PhD in Computer Communications from the University of Minho in 1996. In the lead of the Computer Communications and Networks (CCN) research group he has several participations in R&D international projects on computer communications, protocols and services. He has authored or co-authored dozens of scientific international refereed Journals and Conferences. He served as a technical program committee member for

António D. Costa, PhD, is an Assistant Professor at Department of Informatics, University of Minho, Portugal, where he develops teaching and research activities in the fields of Computer Networks and Computer Communications since 1992. As a researcher, he currently integrates the Computer Communications and Networks (CCN) research group, at Centro Algoritmi, University of Minho. He graduated in Systems and Informatics Engineering in 1992, obtained a MSc Degree in Informatics in 1998 and a PhD Degree in Computer Science in 2006 at



Bruno Ribeiro, completed his Informatics Engineering degree at the Informatics Department from University of Minho in 2014, and later the Networks and Telematic Services Engineering Master degree in 2016, at the same University. Since then, he has been working as a researcher in the ALGORITMI Center (University of Minho) on the CAR2X Communications project, within the scope of the INNOVATIVE CAR HMI research and development program, particularly focused on Intelligent Transportation Systems communications and applications.



Fábio Gonçalves, MSc, graduated in Communications Engineering at the University of Minho in 2014 and has been working as a researcher since then. He started research work at the Instituto de Telecomunicações in Aveiro, researching and building a security framework for an Internet of Things (IoT) platform. Since 2016 he has been working as a researcher in the University of Minho, where he is also a PhD student, focused in security for Vehicular ad Hoc Networks.



of Minho. His teaching and research interests are Information Retrieval, Computer Networks and the use of ICTs in developing countries. In this context, he has participated in several research projects, supervising master and doctoral students. He has more than four dozen peer-reviewed papers in journals and scientific conferences. He collaborated for more than 10 years with the Catholic University of Angola, as Visiting Professor.



technologies and protocols, routing in IP networks, group Communications and multicast routing protocols, Quality of Service in IP networks and simulation of TCP/IP networks.



Óscar Gama, holds a PhD in Electronics and Computers by the University of Minho in 2011 and has been working as a researcher and invited professor since 2012. He has joined several research projects, mainly focused in sensors, protocols and recently in ITS simulation environments for Vehicular ad Hoc Networks, in the scope of the CAR2X Communications project.



Susana Sousa, has a MSc degree in Communications Engineering by the University of Minho in 2013 and has been working since then, firstly as a Telecommunications Engineer and later as a researcher the ALGORITMI Center (University of Minho) within the CAR2X Communications project, a sub-project of the INNOVATIVE CAR HMI research and development program, particularly focused on Intelligent Transportation Systems communications architectures.