An Auxiliary V2I Network for Road Transport and Dynamic Environments

Jorge Godoy\textsuperscript{a}, Vicente Milanés\textsuperscript{b}, Joshué Pérez\textsuperscript{c}, Jorge Villagrá\textsuperscript{a}, Enrique Onieva\textsuperscript{d}

\textsuperscript{a}AUTOPIA program at Center for Automation and Robotics-Spanish National Research Council (CAR-CSIC), 28500 Madrid, Spain.
\textsuperscript{b}California PATH, University of California at Berkeley, Richmond, CA 94804-4698, USA.
\textsuperscript{c}IMARA team at INRIA research center, Paris - ROCQUENCOURT, France.
\textsuperscript{d}Deusto Institute of Technology (DeustoTech), University of Deusto, Bilbao 48007, Spain.

Abstract

Since vehicle-to-infrastructure (V2I) communications require a major initial outlay and continuous maintenance, they have been less frequently implemented than vehicle-to-vehicle (V2V) communications in applications in the field of Intelligent Transportation Systems (ITS). Nevertheless, making use of the information provided by the infrastructure – traffic signals, traffic panels, and so on – can help improve the performance of various ITS applications. The present work describes the design and implementation of a low-cost infrastructure network based on ZigBee technology to alert drivers (or even override them by using active safety systems) in the case of some unexpected circumstance on the road so that they can prepare for the appropriate handling of their vehicles. Its implementation as an auxiliary network within a larger communications scheme allows the network’s load to be reduced and its performance to be improved. The proposed architecture was tested on a real car in a real scenario with quantified results for different applications.

Keywords: Intelligent vehicles, Road traffic control, Traffic management, Road vehicle control, Inter-vehicle communications.

1. Introduction

The number of vehicles on the roads has long been steadily increasing worldwide. Just in the EU27, the number of passenger cars (vehicles de-
signed to seat no more than nine persons, including the driver) grew by 12.77% – around 26 million cars – between 2000 and 2006. Despite this growth, the annual road accident deaths decreased in the same time interval by around 23% (EUROSTAT, 2010), indicating that both vehicles and roads are becoming safer.

The first safety systems developed for road vehicles were such passive systems as seat belts and airbags that operate when a critical situation occurs, e.g., when a crash is unavoidable. Later, with the introduction of the Anti-lock Braking System (ABS) which reduces braking distances on slippery surfaces, researchers and manufacturers made the first major step towards Advanced Driver Assistance Systems (ADAS) designed to aid drivers during driving. Commercial vehicles today incorporate several safety-oriented ADAS solutions which can either alert drivers to hazardous scenarios or act on the vehicle if the driver is distracted. Two examples are the Lane Keeping (LK) system available on Mercedes S and E Classes which causes the steering wheel to vibrate when the vehicle detects that it is leaving its lane unintentionally, and the more recent City Safety system developed by Volvo and incorporated into their XC60 model which, at speeds slower than 20 km/h, is able to stop the vehicle when it detects some obstacle in front, the aim being to reduce the number of rear-end collisions in urban environments.

Despite these encouraging advances, the information obtained via on-board sensors is insufficient in itself for the detection of potential risk situations. In this sense, one of the most challenging lines of research in the ITS arena is the development of communications technologies for ADAS systems based on inter-vehicular cooperation. Recent years have seen the design of the first architectures for this purpose. In 2003, the United States Department Of Transportation (US DOT) launched the Vehicle Infrastructure Integration initiative (VII) with the vision of using Dedicated Short-Range Communications (DSRC) between vehicles in order to achieve improvements in safety and mobility (U.S. DOT, 2010). As a result of this initiative, the Federal Communications Commission (FCC) reserved a frequency band at 5.9 GHz for transportation safety and mobility applications. Subsequently, the IEEE created standards IEEE 1609 and IEEE 802.11p to define a basic architecture for Wireless Access in Vehicular Environments (WAVE). The WAVE architecture comprises three basic elements: (i) the On-Board Unit (OBU), designed to be installed in vehicles and to guarantee connectivity while moving; (ii) the Road-Side Unit (RSU), designed to be installed in such road elements as traffic lights and signals; and (iii) the service channels to
allow bidirectional V2V or V2I connectivity (Uzcategui and Acosta-Marum, 2009). In 2009, after six years of research, the US DOT decided to re-brand VII as the IntelliDrive initiative, the purpose being to study the application of other wireless technologies for mobility applications while keeping DSRC for safety applications (U.S. DOT, 2010).

At the same time, the International Standards Organization (ISO) defined an architecture for V2V and V2I communications known as CALM (first Communication Air-interface, Long and Medium range but later re-branded as Communication Access for Land Mobile) which allows permanent V2x communications through various wireless technologies such as 2G/3G cellular networks, infra-red, WiFi (IEEE 802.11 a/b/g), WAVE, and microwaves, among others (Toulminet et al., 2008; Williams, 2003). This architecture has been implemented in the Cooperative Vehicle-Infrastructure Systems project (CVIS), and the results of mid-project experiments have already demonstrated the viability of its implementation (Ernst et al., 2009).

So far, the use of communications systems has extended the ADAS development scope, allowing to face more complex scenarios. Collision Warning Systems (CWS), Collision Avoidance Systems (CAS) and Cooperative Adaptive Cruise Control (CACC) are some examples of applications which take advantage of the shared information among vehicles (Sengupta et al., 2006; Milanés et al., 2010d, 2012a; Rajamani et al., 2000). Nevertheless, most of these applications focus on completely automated vehicles, neglecting the potential of the infrastructure. On the other hand, the Advanced Traffic Management Systems (ATMS) make use of the infrastructure information to manage the traffic flow, minimizing traffic jams on highways, at intersections, and at merging points (De Schutter et al., 1999; Koller et al., 1994). To that end, the ATMS control different infrastructure elements such as traffic signals, traffic lights, dynamic information panels; based on the incoming information from sensors installed as part of the infrastructure – e.g. electromagnetic loops, cameras, radars. It is clear that, with a solid V2I communications architecture such as CALM, the two sets of applications – ADAS and ATMS – could be perfectly merged, allowing a central system to combine the information coming from vehicles and from the infrastructure in order to improve road safety and traffic efficiency.

This paper deals with the communications architecture developed by the AUTOPIA program, a Spanish research group focused on autonomous driving. The final goal of this program is to create a reliable communications network where information exchanges among all the present agents – i.e. ve-
vehicles, traffic lights, traffic signals and pedestrians – leading to a safer traffic system. In this sense, there are some previous results that are worthy to highlight as the cooperative adaptive cruise control based on V2V communications (Milanés et al., 2012a) and the V2I-based management system for complex traffic situations as merging or intersections (Milanés et al., 2011, 2012b).

Continuing with this line, the present work focuses on the integration of infrastructure information into the existing scheme, with the goal of increasing the amount of data available for controlling the vehicles inside an urban area. To that end, the inclusion of an auxiliary layer into the communications architecture is proposed. This layer will be composed of several road units, which will constantly send relevant information to the nearby vehicles, acting as road beacons. The implementation of the auxiliary layer has been done using ZigBee technology, due to the low-cost and low-power consumption of the devices.

In brief, the following issues will be addressed in the present communication:

- Design of an auxiliary network to be embedded into the current architecture.
- Implementation of the new layer using ZigBee devices, which will require stretch the limits of the technology for an application which it was not intended.
- Validation of the new communication scheme through different applications: (i) information alerts concerning nearby road incidents and (ii) on-demand traffic light control for priority vehicles.

The rest of the paper is organised as follows. Section II introduces the AUTOPIA program and its traffic control scheme. Section III describes the design and implementation of the secondary layer and Sec. IV presents the test platform and the results of the experiments demonstrating its functionality. Finally, Sec. V presents the concluding remarks and future work.

2. AUTOPIA’s Traffic Control Scheme

Beginning in 1998, the AUTOPIA program has been conducting research into autonomous vehicles, its main goal being the development of a control
architecture capable of emulating how a human drives a car. The program currently has a fleet of five fully-automated vehicles: two electric Citroën Berlingo vans known in the program as Babieca and Rocinante; two Citroën C3 cars - one of them convertible - known as Clavileño and Platero; and a recently added electric minibus known as Molinero (Milanés et al., 2011; Godoy et al., 2011; Milanés et al., 2010b). Although each vehicle has been equipped with different specific sensors and actuators, the control architecture is in essence the same for all of them. Its backbone is an On-Board Unit that handles sensor inputs, and a fuzzy logic controller that manages individual actions on throttle, brake, and steering wheel (Pérez et al., 2009).

In recent years, AUTOPIA has gone a step further, focusing on the development of cooperative control techniques based on communications systems. Nevertheless, the idea of a permanent communication link among all the vehicles on a set of several roads, allowing real-time information exchange for the performance of high-risk manoeuvres autonomously, is still beyond the capacity of any existing communications system. For this reason, it is necessary to work towards the development of an architecture capable of processing the traffic data coming from the different vehicles, limiting to a fairly small zone around each one. In this regard, (Milanés et al., 2010a) described a local zone-based communications architecture that has been designed and implemented on the AUTOPIA program test platform.

This communications scheme defines a set of local zones, similar to the cellular arrays used in mobile telephony. For each zone, all the incoming information from vehicles and infrastructure within the range of the zone is routed to a central unit. As first step, this unit pre-process the data, evaluating the level of risk – or collision probability – for the area. Thereafter, it filters the information, transmitting only the relevant data to each vehicle. The performance of this scheme was analysed in (Milanés et al., 2012b) for determining parameters such as the maximum vehicle occupancy for the covered area, network transmission delays and packet-lost probability. The analysis was performed both in simulation and with real vehicles using the IEEE protocols 802.11p and 802.11g, respectively. In recent works, different automated manoeuvres have been implemented using this communications scheme – e.g. Collision Warning System, Merging, and ACC – (Milanés et al., 2010d, 2011, 2012a). These first results proved the validity of the local zone scheme as a first approach to V2I communications. Nonetheless, various additions could be included to work towards a better solution.

Previous AUTOPIA results were mainly focused on V2V communications.
This work describes how infrastructure information can be integrated into this architecture using an auxiliary layer. The objective of this layer is to give advance warning of unexpected traffic circumstances, as might be blocked road, traffic accident, or traffic jams situations. For the layer implementation, the ZigBee technology has been selected.

Figure 1 shows an image of the proposed AUTOPIA’s communications architecture. The cellular division of the environment is shown at the top of the figure. On its turn, the bottom part shows the structure of both the current and the proposed scheme for the local zones. For the current scheme, one can appreciate that only vehicles, along with the central unit, are part of the network. On the other hand, the proposed scheme incorporates several infrastructure elements. Moreover, while the current scheme only considers direct links between the network elements and the central unit, the proposed scheme adds links among the different elements, leading to a communication mesh.

The next section describes the design and implementation of the new layer into the current communications architecture.
3. V2I Support Network

From a perspective of the rate at which the data changes, the relevant road information can be conveniently divided into two types: High Rate (HR) information (e.g., position, orientation, speed, and state of each vehicle), and Low Rate (LR) information (e.g., speed limits, state of the traffic, and weather conditions). In a fully centralised communications scheme such as that described in the previous section, LR transmission of information could cause problems unless each scenario were properly managed. For example, consider a case in which the central unit broadcasts LR information periodically to all the vehicles in its zone. There would then be a group of vehicles which have recently entered the zone, and are having to wait until the next broadcast message for their information to be updated, thus severely restricting their perspective on the environment. Also, all the vehicles in the zone will be receiving and processing the information corresponding to the entire area, even though they might be in a sub-zone in which some of the data would be irrelevant at that moment. If one attempts to resolve these problems by having the central unit periodically send a personalised message to each vehicle entering or traversing the zone, then the unit’s processing load will be greatly increased, affecting its capacity to analyse and manage the HR information.

Today, while there is still no final solution as to the communications system to implement on roads, static and dynamic traffic signals provide drivers with some road information when the signals are within visual range. Static signals normally indicate speed limits and various warnings under normal road conditions, while dynamic signals alert drivers to changes in status of the roads such as accidents, weather conditions, or traffic jams. From the vehicles’ perspective, such static and dynamic traffic signals act as roadway beacons transmitting specific information to all nearby vehicles. This behaviour could be perfectly well emulated without overloading the main network by implementing a secondary network of basic V2I communications involving Road-Side Units (RSUs). The main task of these RSUs would be to broadcast within their range short messages containing the same information as today’s traffic signals, but under the real-time control and management of the central unit. Moreover, if vehicles received real-time alerts of any road incident in their vicinity, this would allow them to create dynamic maps of the zone, and thus re-evaluating and re-planning their routes to avoid problems on the road ahead.
This section will describe the development and implementation of a ZigBee-based auxiliary local network that emulates traffic signal behaviour.

3.1. ZigBee networks

ZigBee is a wireless communications standard developed by the ZigBee Alliance. It is based on the standard IEEE 802.15.4 which is focused on low cost, low power consumption, two-way communications (Kinney, 2003). Although this standard was intended to be a solution for such applications as home automation, industrial control, or even toys, the literature reflects its great acceptance also for other applications. In (Wu et al., 2006) for instance, a bus priority control system was developed using ZigBee technology with a listener network installed at intersections – including control nodes in traffic light regulators – receiving messages from approaching buses in order to change traffic light status in their favour. Moreover, (Tachwali and Refai, 2009) proposed a collision avoidance system for intersections based on signal power measurement of a wireless sensor network. (Wu and Zhao, 2009) and (Zhou et al., 2010) proposed two different ZigBee-based management systems as solutions to control vehicle access to community garages and parking lots. In (Kulshrestha et al., 2009), a ZigBee network was implemented to manage and control charging stations for plug-in hybrid electric vehicles in a parking deck. And (Chumkamon et al., 2010) presented an analysis of a vertical handoff between GPRS and ZigBee networks for vehicular communications.

A conventional ZigBee network allows up to 255 nodes to be connected in different topologies (e.g., star, tree, or mesh). Within each network, there may be up to three types of node: a single coordinator node which manages network connectivity and permits information to be shared between different networks; router nodes which communicate the network’s various devices; and end nodes that can only transmit information to their parent (coordinator or router) node.

From the perspective of message routing, ZigBee networks can be categorised as static or dynamic. In static networks, the transmission routes for each node are pre-defined by the user during the network’s development, and can only be changed by reprogramming the network. In dynamic networks, however, each node’s transmission routes are constructed in real time, thus allowing the network to reconfigure itself in the case of a node failure. There have been various proposals of algorithms designed to solve the routing problem in ZigBee networks (Sun et al., 2007; Ha et al., 2007), but none of them is capable of solving the problem of routing in highly dynamic environments,
as is the case of an urban traffic scenario in which some of the nodes are implemented in moving vehicles.

3.2. Proposed architecture

An auxiliary network has to fulfil two main requirements: (i) to alert drivers as soon as possible to any unusual road condition such as a traffic jam, traffic accident, or vehicle breakdown; and (ii) to deal with a highly dynamic environment with a great number of nodes in continuous movement. The present architecture proposal is thus divided into two main sections – static and dynamic – to allow full advantage to be taken of the properties of ZigBee networks. This proposal will use router and coordinator nodes in a mesh topology to ensure the network’s robustness in the case of failure or disconnection of some node, since there will exist several paths connecting the coordinator node with the router nodes.

3.2.1. Static section

This section of the network covers all the elements which are essentially static relative to the displacement of the vehicles. It thus consists of the central unit of the local zone, the coordinator node, and the router nodes (henceforth Road Nodes, RN). The functions of these elements will be described in the following paragraphs.

The coordinator node will be installed in the central unit. At the application level, this node will be programmed to act as a bidirectional gateway between the ZigBee network and the central unit. The coordinator node and the central unit together will henceforth be termed the Central Node (CN). The RNs need to be fairly uniformly distributed over the local zone, being installed on various elements of the road infrastructure such as traffic signals or traffic lights. They will be programmed as beacons, periodically broadcasting a set of messages to all their one-hop neighbour nodes. A status message needs to be sent periodically from each RN to the CN reporting its state and that of its listened-to neighbours, allowing the central unit to control and check the overall state of the network. As a secondary function, the RN will listen for any incoming message from the dynamic section, and retransmit it to the central unit.

The CN, specifically the central unit, will be responsible for receiving and analyzing the incoming information from the local zone through the main and auxiliary networks, and from neighbouring zones, determining the state of
each RN from the set of messages it has transmitted. The message transmission frequency is set by the state of the nodes. A hierarchical structure is created in order to handle different road events by defining three warning levels as possible node states – alarm, warning, and normal modes, with respective transmission intervals of one, three, and five seconds. In the case of an event occurring, the central unit will first define a zone around the event – the alarm zone – within which all nodes will be in the alarm state, allowing rapid notification to all vehicles inside the area. Then a warning zone will be defined around the alarm zone in order to communicate the event to all vehicles approaching the alarm zone, allowing them to re-plan their route avoiding the event or to adapt their state as they enter on the alarm zone. The RNs that are not within a warning or alarm zone will maintain their normal state. This general scheme is illustrated in Figure 2.

Since this static section operates in the same way as a common ZigBee network, any basic routing algorithm can be implemented to manage and optimise the transmission paths of each node, ensuring that, in the case of a node failure or disconnection, the network would be able to reconfigure itself, thus avoiding any major failure.

3.2.2. Dynamic section

The dynamic section comprises the nodes installed in vehicles, henceforth termed Mobile Nodes (MNs). In an urban scenario, vehicles are in continuous movement within the local zone and between it and neighbouring zones, and it is impossible in principle to determine the configuration (Application ID and Network ID) for each MN without implementing complex maps containing the parameters of each local zone. Moreover, such an implementation of

Figure 2: Example of alarm and warning zones defined by the static section.
maps would make the system highly location dependent, losing the desired characteristic of network adaptability. Likewise, the displacement speed of the nodes makes it impossible to maintain an optimal routing table for each node of the network without overloading the system. It is therefore necessary to implement a solution which guarantees that vehicles can receive and transmit messages regardless of whichever zone and network they find themselves in. To this end, the MNs are programmed as promiscuous gateways between the ZigBee network and each vehicle’s On-Board Unit (OBU).

Being configured in a promiscuous mode allows the MN to receive any message from any network independently of its configuration, and, more importantly, without being part of that network. As the MN would be acting as a gateway, all the messages will be retransmitted to the vehicle’s OBU for analysis. From this analysis, the OBU extracts the principal configuration parameters of the networks currently in its vicinity so as to be able to configure itself with the right parameters in the case that a message needs to be transmitted. As was mentioned in the previous section, the RNs are responsible for listening for any incoming messages from the dynamic section, so as to retransmit them to the CN. This is a strictly required function since the MNs do not belong to any network, and hence there is no routing information for them created by the RNs’ routing algorithm.

Since the MNs do not belong permanently to any network, and in order to overcome the limitation represented by the maximum number of nodes allowed by a conventional ZigBee network, a short range of node IDs has to be reserved exclusively for MNs. This range, designed equally in all the local zones, will be shared by all the MNs as follows. When a MN has to send information to the infrastructure, it configures itself with the nearest network configuration received, takes a random value from the designed range for each of the messages it has to send, and then sends them. The messages will be received by the RN and retransmitted to the CN. This random ID designation does not cause any conflict in the network since this field is used only by the RN to recognise incoming messages from the dynamic section. Of course however, the OBU must include in each message a field containing the Vehicle ID – defined by the main network – to allow the CN to identify the vehicle sending the information.

3.3. Implementation

To evaluate the functionality of the proposed architecture, a test version of the auxiliary network was installed in the infrastructure of the AUTOPIA
program test track. From a previous study with wireless sensors applied to V2I communications (Milanés et al., 2010c), it was determined that IRIS motes, manufactured by CrossBow, were the best commercial option with which to implement the network due to their low cost, long range – 250 metres – and low power consumption in sleep mode – $8 \mu$A. Additionally, IRIS motes integrate a routing algorithm known as XMesh, specially designed by CrossBow for ad-hoc mesh networks. This algorithm is able to elaborate and update an entire network structure in real time, adding and removing nodes without human intervention or network reset (Teo et al., 2006).

The test network comprises a dozen nodes: 11 for the static section – 1 coordinator node and 10 road nodes – and 1 for the dynamic section – installed in one of the AUTOPIA test vehicles. Two MIB520 boards, also manufactured by CrossBow, were used as interfaces, one between the coordinator node and the central unit, and the other between the mobile node and the on-board unit. Figure 3 shows the distribution of the static section nodes around the track. As can be seen, the coordinator node is labeled as node 0, and the road nodes are numbered from 1 to 10. The interval of node IDs between 201 and 254 was reserved for the mobile nodes.

Since ZigBee networks have a narrow bandwidth, message definition is a critical factor in their performance. Moreover, since the maximum payload length per message is 29 bytes, it is necessary to define short message structures but which at the same time include all the essential data for each scenario. In the present test, for which dynamic adaptation of the speed was taken to be one of the main applications of the network, four message structures were defined (see Table 1):

- **Report**: As was described in Sec. 3.2.1, the RNs periodically send a status message to CN informing of their current state. This structure has four fields: $MessageID$, $State$, $EventID$, and $Age$. For all report

<table>
<thead>
<tr>
<th>Struct</th>
<th>Report</th>
<th>Normal</th>
<th>Warning</th>
<th>Vehicle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fields (#bytes)</td>
<td>MessageID = 0x10 (1)</td>
<td>MessageID = 0x21 (1)</td>
<td>MessageID = 0x54/0x55 (1)</td>
<td>MessageID = 0x42 (1)</td>
</tr>
<tr>
<td></td>
<td>State (1)</td>
<td>EventID (1)</td>
<td>EventID (1)</td>
<td>EventID (1)</td>
</tr>
<tr>
<td></td>
<td>EventID (1)</td>
<td>Age (2)</td>
<td>East (4)</td>
<td>East (4)</td>
</tr>
<tr>
<td></td>
<td>Age (2)</td>
<td>North (4)</td>
<td>North (4)</td>
<td>North (4)</td>
</tr>
<tr>
<td></td>
<td>Length</td>
<td>6</td>
<td>3</td>
<td>15</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Vehicle</th>
<th>Pair ID</th>
<th>Warp</th>
<th>Radius</th>
<th>North</th>
<th>East</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vehicle</td>
<td>0x42 (1)</td>
<td>0x54/0x55 (1)</td>
<td>0x21 (1)</td>
<td>0x54/0x55 (1)</td>
<td>0x21 (1)</td>
</tr>
<tr>
<td></td>
<td>East (4)</td>
<td>Left (4)</td>
<td>Right (4)</td>
<td>North (4)</td>
<td>East (4)</td>
</tr>
<tr>
<td></td>
<td>North (4)</td>
<td>West (4)</td>
<td>South (4)</td>
<td>North (4)</td>
<td>East (4)</td>
</tr>
<tr>
<td></td>
<td>East (4)</td>
<td>South (4)</td>
<td>North (4)</td>
<td>East (4)</td>
<td>South (4)</td>
</tr>
<tr>
<td></td>
<td>North (4)</td>
<td>West (4)</td>
<td>South (4)</td>
<td>North (4)</td>
<td>West (4)</td>
</tr>
</tbody>
</table>

Table 1: Message structures defined for network implementation.
messages, the field $MessageID$ is set to 0x10, while the fields $State$, $EventID$, and $Age$ indicate the current node state, the event broadcast by the node, and the time since the last update from CN, respectively.

- **Normal**: Broadcast by the RNs in normal mode. It contains two fields: $MessageID$, set to 0x21; and $Age$, containing the time since the last update from CN.

- **Warning**: Broadcast by the RNs in alarm/warning mode. It has seven fields: $MessageID$, $EventID$, $East$, $North$, $Radius$, $Speed$, and $Age$. $MessageID$ is set to 0x54 in warning state or 0x55 in alarm state. The field $EventID$ indicates the current problem being broadcast, allowing vehicles to deal with several events at the same time. $East$ and $North$ are the UTM coordinates of the geometric centre of the event’s area.

Figure 3: (a) Aerial view of the AUTOPIA’s test track. (b) Node distribution around the test track.
Radius is the radius – in metres – of the affected zone, and Speed indicates the recommended speed inside the defined area for as long as the problem persists. As in the other messages, the Age field contains the time – in seconds – since the last CN update.

- **Vehicle**: Used only by the MN to transmit information to CN. This structure has five fields: MessageID, VehicleID, East, North, and VehicleEvent. For all messages, MessageID is set to 0x42. VehicleID contains the vehicle’s IPv4 in the main (WiFi) network. East and North indicate the vehicle’s current position, and VehicleEvent indicates the type of event notified by the vehicle to CN.

The normal and warning structures are also implemented in CN as possible order structures to be sent to the RNs. This reduces the order analysis process load in the RNs.

Finally, in order to monitor network performance and package transmission – including packages sent by the routing algorithm – an additional promiscuous gateway was connected to an external computer running a monitoring program.

4. Network Validation and Application

This section will describe three experiments validating the functionality of the proposed architecture in a real scenario: network response, dynamic speed adaptation, and on-demand traffic light commutation.

4.1. Network response

To validate the proposed architecture as a real-time solution to informing vehicles about traffic incidents, an experiment was performed to evaluate its response time to MN events. For this test, CN was programmed to switch automatically to the alarm state those RNs that re-transmit a message from each MN. The time lapse between MN transmission and RN commutation is measured and logged by the external monitoring computer. For robustness analysis, the procedure was performed 500 times for each MN location around the test track – see Figure 4. The minimum, maximum, and mean response times for the first and last switched nodes are shown in Figure 5. Since the number of switched nodes varies according to the location of the MN, the fourth switched node was considered to be the last in order to be able to relate the results for all locations with each other.
Figure 4: MN locations for the network response test.

Figure 5: Network response time: (a) entire zone declaration; (b) first node response.
As can be seen in Figure 5a, the maximum time lapse between the MN transmission and the declaration of the four-node alarm zone is around 260 milliseconds in the worst scenario – maximum value on position 4. On the other hand, the minimum value remains within the 150-170 milliseconds range for all cases. On its turn, Figure 5b illustrates the response-time for the first switched node. One observes that the longest delays are around 150 milliseconds – locations 4 and 8 –, while the shortest one is around 50 milliseconds for all locations. From the data logged by the monitoring program, it was determined that the delays in the networks response are proportional to the number of RNs retransmitting the MN message to CN.

According to (Olson and Sivak, 1986; Kestinga et al., 2008), an average human driver’s reaction time is between 1 and 2 seconds, meaning that, even in the worst scenario, the network's response is at least four times faster. From these results, one can conclude that various safety and mobility applications – such as emergency braking, dynamic speed maps, or real-time path planning – could be implemented by means of this network, the main limiting condition being the short byte length of the transmitted messages.

4.2. Speed adaptation

After having validated the network functionality, we designed and implemented an application dynamically adapting the speed in the test vehicle. For this experiment, the MN was installed in Platero. As was mentioned in Sec. 2, all the automated vehicles of the AUTOPIA program are equipped with a fuzzy-based control system capable of performing fully automated driving manoeuvres. For this experiment, only longitudinal control was used. A detailed description of the control system can be found in (Onieva et al., 2010).

When AUTOPIA vehicles are in standalone mode, the speed reference for each sector of the test track is defined by a digital map. In order to allow the reference to be modified in accordance with track status, a sub-process – henceforth termed WSNMap – was added to the control program. WSNMap is responsible for listening for and processing all incoming information from the auxiliary network, updating the speed map in real time. The EventID field defined within the warning and alarm messages allows WSNMap to handle several road incidents at the same time, while skipping those already processed. A general scheme of AUTOPIA’s architecture including the auxiliary network implementation is shown in Figure 6. One observes that both the main and the auxiliary networks allow permanent information exchange.
between the central unit and the vehicles within the local zone. In each automated vehicle, the data received from the on-board sensors and wireless communications are analyzed in the Planning and the Control stages. The central unit receives all the incoming information from the vehicles and infrastructure in order to manage the traffic flow and transmit orders and information to the vehicles.

For each test, an 80-metre alarm zone was defined around one of the track’s corners using RNs 1 to 3. This radio was selected according to the size of the larger edge on the central blocks of the test track – where the roundabout is built in –. Due to the test track’s small size, no warning zone was defined. With this zone definition, the test vehicle performs an automated standalone manoeuvre starting from the point farthest from the alarm zone, crossing the incident area, and returning to the starting point. A general scheme of this experiment is shown in Figure 7.

In Figure 8, for a typical experiment, the upper graph shows the time course of the vehicle’s distance to the restricted corner, and the lower graph shows the reference speed and the vehicle’s actual speed. One observes that the reference speed changed from 40 to 15 km/h as soon as the vehicle entered the restricted area, returning to the normal value when it exited. On average, the vehicle detected the restricted area at around 160 metres before entering it – corresponding to between 12 and 14 seconds ahead of time.

To further improve the vehicle’s behaviour, the control system could be modified to include a planning stage so that the vehicle slows down as it approaches the restricted zone. However, modification of the control system...
Figure 7: Alarm zone definition and vehicle trajectory.

Figure 8: Evolution of the distance and speed.
exceeded the goals defined for the present architecture, and will be considered in future work.

4.3. On-demand traffic light control

Traffic jams are a common problem in today’s urban scenarios, with Priority Vehicles (PVs) – e.g. fire engines, police cars, and ambulances – being affected in the same way as passenger cars. In some cases, such as a congested intersection, the problem is exacerbated by traffic light cycles keeping some vehicles blocking the way through for PVs. Given this context, and in order to test the bidirectional viability of the network, an on-demand traffic light control system was developed for PVs.

The application’s design was for a PV approaching the intersection to send a message through the auxiliary network to CN, informing of its current position and intention at the intersection – to carry straight on through or to make a turn. After receiving the request, CN would analyze the traffic light state and any other zone information available, and then send a specific state command to the traffic light regulator so as to leave a path free for the PV.

For this experiment, Platero’s control system automatically transmits a vehicle message when approaching the intersection. The message was programmed to be sent when Time-to-Intersection (TTI) was less than 5 seconds (Misener, 2010), and the intention at the intersection was codified using the field VehicleEvent of the message. Three possible values were permitted: Right Turn (RT), Straight-through Trajectory (ST), or Left Turn (LT). This intention is chosen manually on the OBU’s screen by the driver. The traffic light control is performed by another application running in CN which, taking a state input, commutes the lights logically from the current to the desired state. In order to have the same time reference, the regulator program was synchronised using the GPS time from the base station with a 200ms resolution.

Tests were performed at an X-shaped intersection on AUTOPIA’s track – ignoring the entry to the roundabout (Figure 9). The figure shows the numbering of the entry points and the traffic lights available at the intersection (represented by arrows and cross-walks in the figure). The numbering allows different vehicle trajectories at the intersection to be labeled in an ENTRY-EXIT format – e.g., 1-4 and 3-2 are examples of a RT trajectory. For each test, the traffic lights were initialised in the worst scenario for the
forthcoming PV demand. The system’s responses were similar for all cases, so that results for only one of them will be presented.

Figure 10 shows the time course of the traffic light states for a 4-1 trajectory. One observes that at time zero – image (a) – the vehicle has yet to reach the notification threshold and the traffic lights are in an unfavourable state. As soon as the vehicle gets beneath the TTI threshold – image (b) – a request is sent to CN. Less than 200 milliseconds after the message was sent – image (c) – the regulator program starts commuting the traffic lights until the desired state is finally reached – image (d). Although the amber state is
not available for pedestrian traffic lights, it is used in image (c) to show the transition between the green and red states, implemented in the real version as a blinking green state.

5. Conclusions and Future Work

Taking into account the different data update rates in road information, and the present evolution of V2I and V2V communications standards, the authors have proposed a novel auxiliary ZigBee-based architecture for the AUTOPIA communications scheme. The design allows, with minimal modification, bidirectional data exchange between static and highly dynamic nodes. With this architecture, vehicles are able to obtain road information from the infrastructure and adapt themselves to the current zone configuration without implementing complex routing maps.

The proposed architecture was implemented at the AUTOPIA program’s test track, and tested in real scenarios with a real vehicle. The network’s applicability was validated in the first of three experiments, in which it was observed that its response was up to four times faster than a human driver’s. Integration of the network with one of AUTOPIA’s test vehicles allowed two further applications to be developed which were tested in the other two experiments: dynamic speed adaptation, and on-demand traffic light control for priority vehicles. These experiments demonstrated the network’s functionality and its adaptability to different ITS applications.

In future work, several features will be added to the auxiliary network as a way to expand the communications scheme to deal with real-time traffic planning and control. Also, an extended analysis will be made of the network’s performance in critical scenarios such as traffic jams (high number of nodes), multiple node failures (network adaptability), and multiple traffic events (high network load).

Acknowledgements

This work was supported by the Spanish Ministry of Science and Innovation through Research Grant TRANSITO TRA2008-06602-C03 and the Spanish Ministry of Development through Research Grant GUIADE P9/08.

J. Godoy wants to especially thank the JAE program (Spanish National Research Council – CSIC) for its support in the development of this work.
References


