Conference Paper

WiCAR - Simulating towards the Wireless Car

DECSoS 2020 is being held as part of the 39th International Conference on Computer Safety, Reliability and Security (SafeComp 2020).

Harrison Kurunathan
Ricardo Severino
Eduardo Tovar

CISTER-TR-200504

2020/09/15

WiCAR - Simulating towards the Wireless Car

Harrison Kurunathan, Ricardo Severino, Eduardo Tovar

CISTER Research Centre
Polytechnic Institute of Porto (ISEP P.Porto)
Rua Dr. António Bernardino de Almeida, 431
4200-072 Porto
Portugal
Tel.: +351.22.8340509, Fax: +351.22.8321159
E-mail: hhkur@isep.ipp.pt, sev@isep.ipp.pt, emt@isep.ipp.pt
https://www.cister-labs.pt

Abstract

Advanced driving assistance systems (ADAS) pose stringent requirements to a system's control and communications, in terms of timeliness and reliability, hence, wireless communications have not been seriously considered a potential candidate for such deployments. However, recent developments in these technologies are supporting unprecedented levels of reliability and predictability. This can enable a new generation of ADAS systems with increased flexibility and the possibility of retrofitting older vehicles. However, to effectively test and validate these systems, there is a need for tools t hat can support the simulation of these complex communication infrastructures from the control and the networking perspective. This paper introduces a co-simulation framework that enables the simulation of an ADAS application scenario in these two fronts, analyzing the relationship between different vehicle dynamics and the delay required for the system to operate safely, exploring the performance limits of different wireless network configurations.

WiCAR - Simulating towards the Wireless Car

Harrison Kurunathan, Ricardo Severino, Ênio Filho, Eduardo Tovar

CISTER/INESC TEC and ISEP-IPP, Porto, Portugal
`{hhkur, rarss, enpvf, emt}@isep.ipp.pt`

Abstract. Advanced driving assistance systems (ADAS) pose stringent requirements to a system's control and communications, in terms of timeliness and reliability, hence, wireless communications have not been seriously considered a potential candidate for such deployments. However, recent developments in these technologies are supporting unprecedented levels of reliability and predictability. This can enable a new generation of ADAS systems with increased flexibility and the possibility of retrofitting older vehicles. However, to effectively test and validate these systems, there is a need for tools that can support the simulation of these complex communication infrastructures from the control and the networking perspective. This paper introduces a co-simulation framework that enables the simulation of an ADAS application scenario in these two fronts, analyzing the relationship between different vehicle dynamics and the delay required for the system to operate safely, exploring the performance limits of different wireless network configurations.

Keywords: Automotive· Safety · ADAS · Intra-vehicle communication · DSME · robotic-network co-simulation. ·

1 Introduction

In the past decade, Wireless Sensor Networks (WSN) have been widely adopted and supporting several innovative applications in a multitude of domains, such as in health, security, and agricultural. Nowadays, the increasing miniaturization of modern embedded systems, together with the advancements in the area of WSNs and energy harvesting, have opened up new possibilities to fit wireless communications into an unexpected series of applications. The automotive industry, has understandably been reluctant to adopt WSN, mostly pointing out its non-deterministic communication behaviour, unreliability due to interference and security issues. Therefore, wireless has been confined to some limited functionalities of infotainment systems and its adaption in critical systems has been non-existent in vehicles, although it has been already enabling a series of critical scenarios in other industrial domains.

The day-to-day automobile has gradually evolved from fully mechanical design to a fully electronically equipped modern car. The existing subsystems of a modern car consist of several sensors and actuators that are coupled with hundreds of Electronic Control Units (ECU) that are interconnected through thick wired harnesses and communicate based on real-time communication protocols. These wired harnesses can increase the overall weight of the car resulting reduction of the performance of the vehicle in terms of fuel consumption. Thus, the excess weight of the car also can be extrapolated to an environmental issue.

Current trend, is to continue to increase the number of application modules and complexity in the vehicle, by fitting newer models with improved advanced driving assistance systems (ADAS) to increase their safety. However, this effort is not being applied to the millions of older vehicles that will continue to share the roads in the next 15 years, partially due to the tremendous complexity involved in retrofitting such vehicles. Wireless communications can potentially become an enabling technology to support such possibility, considering its flexibility and ease of deployment, by exploring the innovative plug-and-play possibilities introduced by these networked sensor networks. Ideally, additional additional sensing arrays could be introduced into the vehicle with minimum complexity, and without requiring complex re-wiring. However, ADAS pose stringent requirements to a system's control and communications, in terms of timeliness and reliability, and these properties must be ensured by the communications technology. The improvements to the low-power, low-rate IEEE 802.15.4 standard [1], introduced by the .e amendment, enables interesting features such as guaranteed bandwidth, deterministic delay and several other improved reliability support via the introduction of multi-channel techniques. These characteristics turn this communication technology as a prominent candidate to support wireless ADAS as well as other non-critical applications.

However, to effectively test and validate these systems, there is a need for tools that can support the simulation of these complex communication infrastructures from the control and the networking perspective, focusing on the interplay between these two dimensions. This paper introduces a co-simulation framework that enables the simulation of an ADAS application scenario in these two fronts, analyzing the relationship between vehicle dynamics, i.e. speed and braking force, and the delay required for the system to operate safely, exploring the performance limits of different network configurations of the DSME protocol.

The main contributions of this paper are as follows:

- We provide a co-simulation framework that joins a network simulator fitted with a DSME communications stack i.e. OmNet++/OpenDSME, with a robotics simulator i.e. Gazebo, that simulates the control and dynamics of a real vehicle.
- We implement a proof-of-concept Parking Assistance ADAS systems that relies on external sensors and wireless communications.
- We investigate the adequacy of the DSME MAC behavior of IEEE 802.15.4 for supporting the ADAS, and from the application perspective, we determine speed limits that guarantee the safety of the system.

2 Related Work

The research community has continuously looked into the possibility of using Wireless Sensor and Actuator Networks (WSANs) in intra-car communication.

One of the foremost motivation for its implementation is to reduce the weight of the car and increase the overall performance in terms of fuel economy and reliability. Researchers in [2] investigated the design aspects of WSANs in intra-car systems and if whether they could become a viable solution to partially replace or enhance current wired measurement and control subsystems.

In [3], authors used IEEE 802.15.4 Compliant and ZigBee Ready RF Transceivers to create a Blind Spot Information System (BLIS). BLIS systems implemented by many car manufacturers (e.g., General Motors, Ford, and Volvo) are based on costly hardware components such as cameras and radars. The proposed intra-car system in this work was non-intrusive at the same time cost-efficient. This work provided important information on the ideal location for sensors in an intra-car system, which we have adopted in our intra-car scenario depicted in Figure 3.

Case studies such as [4] have proven that multi-hop has the potential for providing additional reliability, robustness, and energy usage improvements over existing single-hop approaches. In their study, they state that aggregating data in one or several processing centers in the vehicle is critical for the monitoring capabilities of the sensors, which are constrained by both energy and computational power. Multi-hop systems, despite its large overhead, can enhance system reliability, robust performance, and reduce communication energy. In our work, we look into a communication technology which features multi-hop and multi-channel capabilities and hence can enhance the performance of the network.

There have also been several simulation studies [5], [6] on implementing low power and low rate wireless sensor networks for intra-vehicle communications. These authors considered ZigBee to be a good candidate because of its mesh networking capabilities and low power consumption. Zigbee solves multi-path fading using Direct Sequence Spread Spectrum (DSSS) technology and interference resilience using Carrier Sense Multiple Access (CSMA). The propagation channel inside a vehicle is closed and is affected by the mechanical vibrations caused by the movement of the vehicle. Hence authors propose a simulation of the physical layer of the ZigBee network and the propagation channel inside a vehicle along with an adaptive equalizer at the receiver. Though Zigbee had mesh capabilities, determinism is not assured in such networks due to the usage of a contention-based mechanism for transmission. From our previous works [7], [8] we were able to confirm that DSME had the capability to communicate under strict time bounds and support time-critical applications. In this work, we rely on DSME which supports both a contention-based to be a possible candidate for intra-car communication systems.

3 Co-simulation Framework

Simulation of integrated application and network models can be done in diverse ways either by co-simulating with two different simulators, by expanding a network simulator with physical models [9] or by expanding the physical simulator with network model [10]. However, joining two, or more, well-proven simulators, in each particular area, can offer significant advantages. Kudelski et al. in their work [11] propose a an integrated framework to support multi-robot and network simulation. In this work the authors propse an integration of three simulators namely ARGoS [12], NS-2 and NS-3 that can be used in co-simulation scenarios. ARGoS is a ARGoS is a multi-physics robot simulator that can simulate large-scale swarms of robots of multiple variants. Similarly to the Gazebo simulator, which we use in our work, ARGoS can be extended with plugins, however, the integration of Gazebo with ROS constitutes an undeniable advantage, by providing flexibility, modularity, and easing robotics integration. NS-2 and NS-3 are legacy network simulators that can simulate a network stack. In this work the authors propose a synchronization approach between the simulators in which the number of nodes, characteristics of the equipment and simulation area a synchronized together. At every simulation step the ARGoS sends the updated robot position to the network simulator and the communication is carried our and is transferred back to the robotic simulator that the data packets are carried out. In our work, we take a similar approach by integrating Gazebo with OMNeT++++. In our case, we use the ROS sync application to handle the synchronization in our simulations as it will be shown in the next section.

BARAKA [13] is another co-simulator tool introduced by Thomas Halva Labella et al. In this work they provide a tool that is able to perform integrated simulation of communication networks using OMNeT++ and robotic aspects using Open Dynamics Engine (ODE) [14] for rigid-body physics simulation. The steps for integration in this simulation is done in two steps, they first integrate the collision/detection step loop in the OMNeT++. Then they create modules that simulate the robots and motes both in the physical aspects. Finally these modules are accessed by an agent program to control the behavior of the agents in the simulated world. The ODE loop in the OMNeT++ in this case has no connection to any other module in the simulation. In our work, for every simulation step, the simulators are synchronized in a seamless way by relying upon the ROS middleware and its topics. The flexibility of such middleware is tremendous and we use it for exchanging information between the simulators.

In this work we built a Wireless-ADAS co-simulation framework that combines the network simulation capabilities of OMNeT++/INET and the ability to emulate the vehicle physics and sensors behaviour in 3D scenarios using the Gazebo robotics simulator. This will enable us to analyze the mutual impact between the control and the networking aspects. The integration is done over the Robotics Operating System (ROS), based our previous works in [15], [16] which focused on inter-vehicle communications (i.e. using ETSI ITS-G5) to enable a cooperative platooning function. A general Architecture for our framework is presented in Figure 1. The integration of the network model is supported by the openDSME open-source framework [17] to implement the DSME protocol on top of the IEEE 802.15.4 physical layer. Two kinds of nodes are implemented in OMNeT++/INET simulation: the sensor nodes and the sink, corresponding to 8 end-devices and a PAN Coordinator respectively. In the OmNet++/INET side, the displayed outward 8 nodes (sensor nodes - IEEE 802.15.4 End Devices) correspond to the wireless radar/sonar modules implemented in the Gazebo vehicle model to achieve a 360 degree coverage of the vehicle without any blind spots. At the center of the layout, the ”sink” node (IEEE 802.15.4 PAN Coordinator) is also displayed and corresponds to the Application Unit (AU) wireless interface. The AU is responsible for the ADAS system control implementation. It processes the sensor inputs and reacts accordingly, by interfacing the vehicle’s steering and braking systems. To handle the synchronization between the two simulations, we developed a ROS Sync Application, which we describe next.

Synchronization Approach

OMNeT++ is an event-driven simulator and Gazebo a time-driven simulator, therefore synchronizing both simulators represented a key challenge. In order to accomplish this, a synchronization module was implemented in OMNeT++ to carry out this task, relying upon the ROS ”/Clock” topic as clock reference. The OMNeT++ synchronization module subscribes to ROS’ ”/Clock” topic, published at every Gazebo simulation step (i.e. every 1ms) and proceeds to schedule a custom made OMNeT++ message for this purpose (”syncMsg”) to an exact ROS time, which allows the OMNeT++ simulator engine to generate an event upon reaching that timestamp and be able to execute any other simulation process that must be run.

Data Workflow

In order to support data flowing between the Gazebo and OMNeT++ simulators, the ROS publish/subscribe middle-ware support was crucial. For each node in the OmNet++/INET simulation, there is a corresponding sensor in the Gazebo vehicle model which publishes its relevant data into a rostopic i.e. ”/car1/sensors/sonar1”. In the OmNet++/INET side, each node subscribes to the corresponding rostopic and prepares a message that is en-queued into the openDSME MAC layer to be transmitted to the sink node, which role is assumed by the network PAN Coordinator. OpenDSME handles the transmission and, if successful, the sink node publishes a rostopic with the sensor data that is subscribed by the AU. The AU then uses this input to feed its control loop. As for the Gazebo model, a Toyota Prius car model (visible at fig.1) is used as the baseline deployment for this WSN layout with 7 sonars and a radar. With this general layout architecture, different ADAS scenarios can be implemented, by changing sensors or their characteristics, the vehicle model, the track and the surrounding environment, enabling the possibility to extensively test and validate a ADAS behaviour and explore its performance limits pre-deployment.

For the upcoming ADAS, vehicles are increasingly being equipped with a wide variety of sensors, in order to get a good awareness of their surroundings. In addition, Sensors are already being deployed in current ADAS to evaluate the status of some of the vehicle components (i.e., steer, brakes) to detect stress and prevent any failure. In this framework, all these sensors, can be implemented in a vehicle model, and later be integrated into the network model as a new node

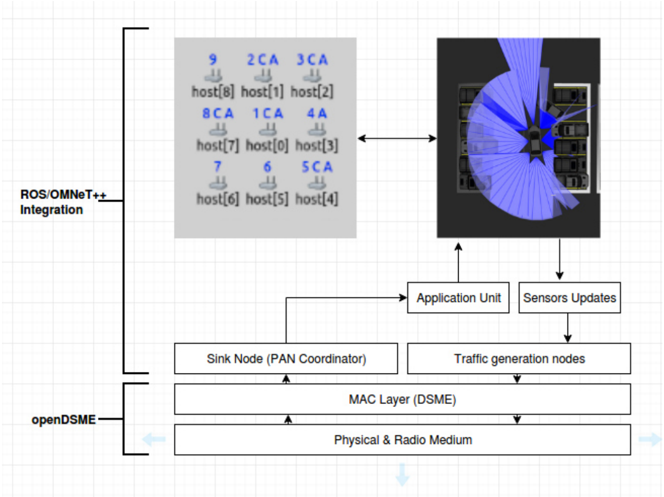

Fig. 1. Integration Architecture

that feeds data into the AU, for a integrated perspective of the system on a multitude of scenarios.

4 Network Specification

For our intra-car system, we used the DSME MAC behavior of IEEE 802.15.4e because of its deterministic capabilities. The DSME network provides deterministic communication using its beacon-enabled mode. This mode is supported by multisuperframes that may contain stacks of superframes, as shown in Fig. 2. Each superframe comprises a Contention Access Period (CAP) in which the nodes contend to access the channel and a Contention Free Period (CFP) in which the nodes send the data using Guaranteed timeslots (GTSs). It is in this period that the vehicle's sensors are accommodated, for guaranteed service.

The superframe is defined by BO, the Beacon Order which is the transmission interval of a beacon in a superframe. MO is the Multi superframe Order that represents the Enhanced Beacon interval of a multi-superframe, and SO is the Superframe Order that represents the beacon interval of a superframe. The number of superframes in a multisuperframe is given by 2^{MO-SO}. These values are conveyed to the nodes by an Enhanced Beacon (EB) at the beginning of each Multisuperframe. Reducing the values of SO and MO reduces the size of the timeslots and the number of superframes in a multi superframe duration, but also decreases the network's latency. In what follows we evaluate the relationship between such network settings and latency in the context of a ADAS application as a proof-of-concept.

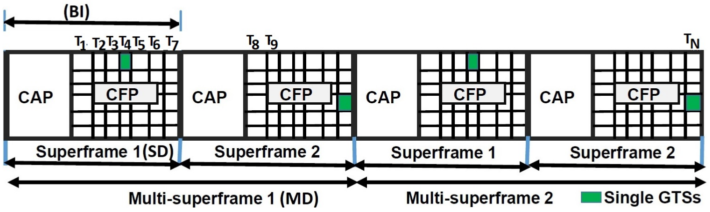

Fig. 2. DSME superframe

5 Performance analysis

To showcase our proposal and simulation tool, we evaluate a parking lot Wireless ADAS scenario presented in Figure 3. When roaming inside a parking lot while searching for a parking spot, a driver can suffer from decreased perception of the overall environment. As his attention diverges from the driving actions into his visual search for the parking space, his ability to respond to unexpected situations is hindered, and may not be capable of perceiving an obstacle in time to avoid it. In this case, we consider the obstacle as a car that suddenly exits a parking space form the right-hand side of our vehicle. We push the requirements of the scenario to a point in which a typical driver would be unable to stop the car in time due to his reaction times. In this scenario, we consider the car can be traveling up to 30 Km/h (typical maximum speed inside a parking lot) and is fitted with an array of sensors covering a 360 degree field of view.

We evaluate this scenario from the two complementary perspectives. Firstly, we take the application perspective, by varying the braking capacity of the vehicle and its speed, and then the network perspective, by varying the MO and SO settings, and thus its worst-case delay. This is one of the greatest advantages of our co-simulation tool, which enables a multi-dimensional assessment of an application scenario.

5.1 Impact of braking force

Braking capacity is one of the common parameters in any car that deteriorates over time. This is a result of the loss of friction in the clamping mechanism while actuating a brake. In a 100% operational brake, the clamping load is assumed to act on all friction surfaces equally. The loss in this force is only generated when the wheel does not lock because the friction of a sliding wheel is much lower than a rotating one.

In this experiment, we study the limits of this system by averaging the results for several trials for different braking forces and calculating the maximum acceptable delay for the vehicle to operate without a crash. From the results in Figure 4, it is evident and expected, that the braking force and vehicle speed impose different requirements into the network delay. Decreased braking capacity or higher speeds demand lower communication delays to avoid the crash. At

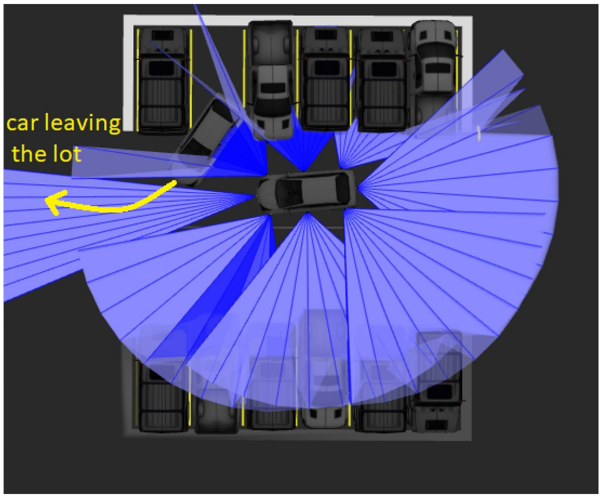

Fig. 3. Scenario taken for evaluation

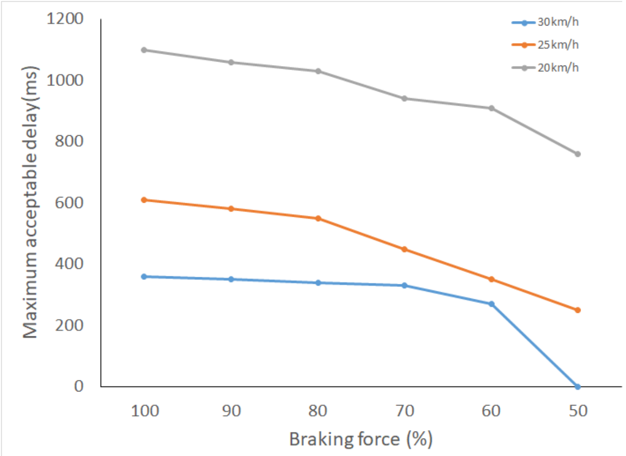

Fig. 4. Maximum acceptable delay for the braking force applied

30 km/h, with a 50% braking capacity, the vehicle is unable to avoid hitting the car leaving the parking space, independently of the delay. This is the point where we reach the performance limit of the control system as dictated by vehicle dynamics.

5.2 Impact of network settings

We carried out several trials for these application settings, and different network MO/SO settings, to explore the performance limits of the Wireless ADAS scenario. Figure 5 presents the communication's delay tolerances, for different speeds (25 and 30 km/h) and braking capacities (100% to 50%), to prevent a crash, superimposed by the overall bounded delay at different network settings.

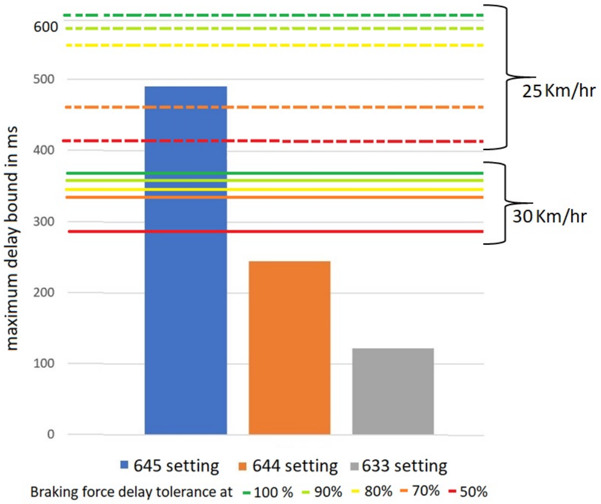

Fig. 5. Impact of static scheduling and braking force on the crash rate

As observed in Figure 5, if the vehicle travels at 25 km/h in the parking lot, and has its braking capacity at 80%, it can still allow approximately 550 ms of delay in the ADAS communications; therefore, a (BO/SO/MO) = (6/4/5) setting suffices. This is important considering the usage of a higher MO can support the allocation of additional superframes and support additional nodes, particularly if CAP reduction is activated, increasing the scalability of the system. Thus finding this trade-of between delay and scalability, in parallel with speed and braking capacity, can lead to increased efficiency and safety. When the braking capacity reduces to 70 or 60 %, the maximum acceptable delay decreases steeply and can only be met by lower MO/SO network settings. This is also the case for a speed of 30Km/h, that even at 100% braking capacity, only (BO/SO/MO) = (6/4/4) settings or lower, can meet the imposed delay requirement of approximately 360 ms. These results show us that for those settings, at the targeted speed for our scenario of 30 km/h, our system can still guarantee the safety of the vehicle even with its braking capacity impaired by 50%

5.3 Impact of delay

One of the important prerequisite for the safe functioning of this scenario is the ability to adhere to a maximum speed limit of 30 Km/h. To achieve this we must be able to provide a maximum delay bound of 350 ms which is a crucial aspect for safe functioning. Hence, we must verify the determinism of the network. In the worst-case scenario, the maximum time a superframe can take to accommodate a transmission will be the size of the superframe. Hence by varying the size of the superframe, we will be able to control the latency of the network and determine definite bounds. The following experiment is carried out with (BO/MO/SO) = (6/4/4) setting with fixed static schedule. As previously mentioned, the results strictly adhere to the limit of the worst-case delay. We experience a maximum delay of 0.23s and it is bounded as seen in Figure 6. This also means we will be also able to operate the application at a steady speed of 30 km/h with a fixed delay using this setting.

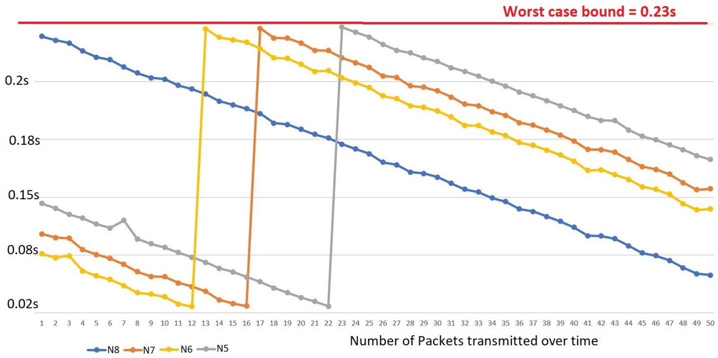

Fig. 6. Delay with Static scheduling for MO=4 and SO=4

The fluctuations of delay values in these static settings can be attributed to the arrival time of the packet. The packets that are served immediately with respect to its arrival result in a much lower delay. The worst-case delay is produced when the sensor data arrival happens at the end of the first superframe and gets scheduled for its adjacent superframe. One significant advantage of static scheduling is that the user has the possibility to vary the network settings and fix a steady worst-case bound based on the network prerequisite.

6 Conclusion and future scope

In this work we introduce a co-simulation tool that can combine both the network and the application perspectives of a realistic ADAS scenario. As a proof of concept, we provide a detailed delay analysis with the DSME network to evaluate its ability to meet the required deadlines for the control system. Furthermore, we also implemented a static scheduled network that can help in providing worst case delay deterministic bounds to ensure the safety of the system and explored it in our scenario at different speeds and braking capabilities. Preliminary results, show that the DSME network can cope with these, however, further exploration work is needed, and particularly a new more demanding scenarios with different sensors, vehicles, and speeds can and will be studied. Overall, the co-simulation framework proved to be up to the evaluation task and we are confident it will become mandatory tool to carryout a serious analysis of such networked ADAS systems.

Acknowledgements

This work was partially supported by National Funds through FCT/MCTES (Portuguese Foundation for Science and Technology), within the CISTER Research Unit (UIDB/04234/2020).

References

1. H. Kurunathan, R. Severino, A. Koubaa, and E. Tovar, "Ieee 802.15. 4e in a nutshell: Survey and performance evaluation," *IEEE Communications Surveys & Tutorials*, vol. 20, no. 3, pp. 1989–2010, 2018.
2. G. Stamatescu, D. Popescu, and I. Stamatescu, "Modeling for deployment techniques for intra-car wireless sensor networks," in *2014 18th International Conference on System Theory, Control and Computing (ICSTCC)*. IEEE, 2014, pp. 501–505.
3. J.-R. Lin, T. Talty, and O. K. Tonguz, "Feasibility of safety applications based on intra-car wireless sensor networks: A case study," in *2011 IEEE Vehicular Technology Conference (VTC Fall)*. IEEE, 2011, pp. 1–5.
4. M. Hashemi, W. Si, M. Laifenfeld, D. Starobinski, and A. Trachtenberg, "Intra-car multihop wireless sensor networking: a case study," *IEEE Communications Magazine*, vol. 52, no. 12, pp. 183–191, 2014.
5. A. D. G. Reddy and B. Ramkumar, "Simulation studies on zigbee network for in-vehicle wireless communications," in *2014 International Conference on Computer Communication and Informatics*. IEEE, 2014, pp. 1–6.
6. H.-M. Tsai, O. K. Tonguz, C. Saraydar, T. Talty, M. Ames, and A. Macdonald, "Zigbee-based intra-car wireless sensor networks: a case study," *IEEE Wireless Communications*, vol. 14, no. 6, pp. 67–77, 2007.
7. H. Kurunathan, R. Severino, A. Koubâa, and E. Tovar, "Worst-case bound analysis for the time-critical mac behaviors of ieee 802.15. 4e," in *2017 IEEE 13th International Workshop on Factory Communication Systems (WFCS)*. IEEE, 2017, pp. 1–9.
8. H. Kurunathan, R. Severino, Anis, and E. Tovar, "Symphony: routing aware scheduling for dsme networks," pp. 26–31, 2020.
9. D. Ramos, L. Oliveira, L. Almeida, and U. Moreno, "Network interference on cooperative mobile robots consensus," in *Robot 2015: Second Iberian Robotics Conference*. Springer, 2016, pp. 651–663.
10. W. Li, X. Zhang, and H. Li, "Co-simulation platforms for co-design of networked control systems: An overview," *Control Engineering Practice*, vol. 23, pp. 44–56, 2014.
11. M. Kudelski, L. M. Gambardella, and G. A. Di Caro, "Robonetsim: An integrated framework for multi-robot and network simulation," *Robotics and Autonomous Systems*, vol. 61, no. 5, pp. 483–496, 2013.

12. C. Pinciroli, V. Trianni, R. O'Grady, G. Pini, A. Brutschy, M. Brambilla, N. Mathews, E. Ferrante, G. Di Caro, F. Ducatelle *et al.*, "Argos: a modular, parallel, multi-engine simulator for multi-robot systems," *Swarm intelligence*, vol. 6, no. 4, pp. 271–295, 2012.
13. T. H. Labella, I. Dietrich, and F. Dressler, "Baraka: A hybrid simulator of sanets," in *2007 2nd International Conference on Communication Systems Software and Middleware*. IEEE, 2007, pp. 1–8.
14. R. Smith *et al.*, "Open dynamics engine," 2005.
15. B. Vieira, R. Severino, A. Koubaa, and E. Tovar, "Towards a realistic simulation framework for vehicular platooning applications." in *22nd IEEE International Symposium on Real-Time Computing (ISORC 2019)*. Institute of Electrical and Electronics Engineers, 2019.
16. R. Severino, E. Vasconcelos Filho, B. Vieira, A. Koubaa, and E. Tovar, "COPADRIVe - a realistic simulation framework for cooperative autonomous driving applications," in *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE) (IEEE ICCVE 2019)*, Graz, Austria, Nov. 2019.
17. F. Kauer, M. Köstler, T. Lübkert, and V. Turau, "Opendsme - a portable framework for reliable wireless sensor and actuator networks (demonstration)," in *Proceedings of the 3rd International Conference on Networked Systems (NetSys 2017)*, Mar. 2017.