Towards Adaptive Car-to-Cloud Communication

Stefan Herrnleben
Chair of Software Engineering
Universität Würzburg
Würzburg, Germany
stefan.herrnleben@uni-wuerzburg.de

Martin Pfannemüller
Chair of Information Systems II
Universität Mannheim
Mannheim, Germany
martin.pfannemueller@uni-mannheim.de

Christian Krupitzer
Chair of Software Engineering
Universität Würzburg
Würzburg, Germany
christian.krupitzer@uni-wuerzburg.de

Samuel Kounev
Chair of Software Engineering
Universität Würzburg
Würzburg, Germany
samuel.kounev@uni-wuerzburg.de

Michele Segata
Advanced Networking Systems
University of Trento
Trento, Italy
msegata@disi.unitn.it

Felix Fastnacht, Magnus Nigmann
Connected Car Innovations
Intedis GmbH & Co. KG
Würzburg, Germany
{felix.fastnacht,magnus.nigmann}@intedis.com

***Abstract*—Recent developments in communication technology have led to cloud resources becoming ubiquitous. These resources enable many new applications by offering computational power for remote embedded devices. In combination with advances in the area of smart driving, this seems to be especially beneficial for applications such as remote maintenance of vehicles or integration with smart city services. As autonomous driving continues to gain traction, Car-to-Cloud communication can support transferring collected data to the cloud, e.g., for dynamic learning of new map information. Additionally, passengers can benefit from novel entertainment services. All these developments require a stable connection between a mobile vehicle and the cloud resources. In this vision paper, we survey Car-to-Cloud communication applications. Based on the analysis of the varying requirements for these applications, we formulate research questions and challenges. Further, we discuss how these challenges can be addressed by means of an adaptive Car-to-Cloud communication middleware. We conclude with an overview on our activities in this area and an outlook on our planned future work on adaptive communication.**

***Index Terms*—vehicular communication, cloud computing, smart city, intelligent transportation systems, adaptive communication systems**

I. Introduction

In recent years, car manufacturers have demonstrated that autonomous driving is not science fiction anymore. This development comes in hand with new technologies in the area of vehicular communication. Further, the omnipresence of cloud resources enables new applications for vehicular communication, ranging from information services to dynamic routing, entertainment and driving safety. Especially cities might benefit from such applications, e.g., through dynamic routing or parking lot search.

However, such applications have many different communication requirements and characteristics, e.g., different types of workloads with varying real-time constraints. Further, some applications require a connection to the cloud while driving, e.g., dynamic routing, while others can operate while the car is parked, such as the transmission of collected camera data for updating navigation maps. Hence, a one-size-fits-all approach for Car-to-Cloud communication is infeasible.

In this paper, we present our vision of adaptive Car-to-Cloud communication, enabled by an adaptive communication middleware that is able to switch between different communication protocols and transfer media, as well as to adapt the communication behavior of applications. The paper provides three contributions. First, we characterize different relevant applications based on: (i) whether the data exchange happens online (i.e., while driving) or offline, (ii) it is constant or event-based, (iii) the data load, (iv) whether it is stream-based or file-based, (v) the frequency of data exchange, and (vi) its applicability for smart cities. Second, we present a system model for adaptive Car-to-Cloud communication middleware. Third, we discuss the challenges for adaptive cloud-based vehicular communication.

The remainder of the paper is structured as follows. Section II describes the foundations of vehicular communication. Section III presents an overview of Car-to-Cloud communication applications. Following this, Section IV presents our research agenda and elaborates different challenges for Car-to-Cloud communication based on analysis of the presented applications. In Section V, we analyze existing approaches and motivate the research gap. Section VI describes the current project status as well as planned activities. Finally, Section VII concludes the paper.

II. Background: Vehicular Communication

Vehicle-to-anything or *Vehicle-to-X (V2X)* communication summarizes any communication between a vehicle and a second party. In this context, a vehicle can be anything from cars to airplanes to trains [1]. In most cases, V2X communication can be divided into two subclasses: (i) vehicle-to-vehicle communication and (ii) vehicle-to-infrastructure communication. *Vehicle-to-Vehicle (V2V)* communication covers direct communication between two vehicles using any kind of wireless transmission channel based on the Wifi IEEE 802.11p

978-1-5386-9151-9/19/$31.00 ©2019 IEEE

TABLE I
Overview of different Car-to-Cloud applications.

	When?	Priority?	What?	How?	How frequent?	Where?	Which time frame?	Which direction?
Real-time navigation	Online	High	Medium	File	Regularly	Both	Soft real-time	Bidirectional
Real-time traffic information	Online	High	Medium	File	Regularly	Both	Soft real-time	Unidirectional
Real-time context information	Online	High	Low	File	On demand	Both	Hard real-time	Unidirectional
Parking lot search	Online	Low	Low	File	On demand	City	Soft real-time	Bidirectional
Charging station management	Online	Low	Low	File	On demand	Both	Soft real-time	Bidirectional
Remote diagnostic information	Offline	Low	Medium	File	Regularly	Independent	Delay-tolerant	Unidirectional
eCall system	Offline	High	Medium	Stream	On demand	Independent	Hard real-time	Bidirectional
Software update over air	Offline	High	Medium	File	On demand	Independent	Soft real-time	Unidirectional
Remote data collection for model facelift	Offline	Low	Medium	File	Regularly	Independent	Delay-tolerant	Unidirectional
Update of maps	Offline	Low	High	Stream	Regularly	Both	Delay-tolerant	Unidirectional
Remote control of autonomous vehicles	Online	High	High	Stream	Constant	Both	Hard real-time	Bidirectional
Machine learning for autonomic driving	Offline	Low	High	Stream	Regularly	Independent	Delay-tolerant	Unidirectional
Video streaming	Online	Low	High	Stream	Constant	Independent	Hard real-time	Unidirectional
Vehicle as an office	Online	Medium	High	Stream	Constant	Independent	Soft real-time	Unidirectional
Vehicle as gateway	Online	Low	High	Stream	Constant	Independent	Soft real-time	Unidirectional
Vehicular social networking	Online	Low	Medium	File	On demand	Independent	Delay-tolerant	Bidirectional
Insurance profiling	Offline	Low	Low	File	Regularly	Independent	Delay-tolerant	Unidirectional
Accident black box data	Offline	Low	Low	File	On demand	Independent	Delay-tolerant	Unidirectional
Section control	Offline	Low	Low	File	Regularly	Independent	Delay-tolerant	Unidirectional
Tolling	Offline	Low	Low	File	On demand	Independent	Delay-tolerant	Unidirectional
Online fleet management	Online	Low	Medium	File	Constant	Independent	Soft real-time	Bidirectional
Pricing of rental cars by wastage	Offline	Low	Low	File	Regularly	Independent	Delay-tolerant	Unidirectional
Platooning coordination	Online	High	Medium	File	Constant	Highway	Soft real-time	Bidirectional
Inner-city platooning	Online	High	Medium	File	Constant	City	Hard real-time	Bidirectional

standard, DSRC, or Bluetooth (IEEE 802.15.1). *Vehicle-to-Infrastructure (V2I)* communication provides the foundation for cloud-based services. The infrastructure can consist of so-called *road side units (RSUs)* – which are fixed communication stations deployed along the road – or cellular communication networks. In comparison to V2V applications, they can provide information to vehicles even in sparse traffic density [2]. Comparable to *mobile ad-hoc networks (MANETs)*, *vehicle ad-hoc networks (VANETs)* are characterized by the lack of any fixed infrastructure or base station. They are made up of communication nodes of vehicles forming a network and enabling nearby vehicles to communicate with each other. Although VANETs are technically MANETs, they inherit certain characteristics which impose new challenges on network stacks and routing protocols in comparison to MANETs. Sichitiu et al. [3] identify five main differences between MANETs and VANETs related to the addressing mechanisms, the rate of link changes, the mobility model, the energy efficiency, and the respective relevant applications.

In this paper, we focus on V2I communication proposing a new adaptive communication middleware that integrates cloud services and vehicles. Next, we present an overview of different applications for cloud-based V2I communication.

III. Applications for Cloud-based V2I Communication

V2X communication enables a large range of applications including platooning [4], collision warning on highways, traffic monitoring and traffic light control [5], increasing both traffic safety and efficiency. An exhaustive overview on such applications is provided in [6] and [7]. In the following, we present an overview on applications for cloud-based V2I communication, hence, we neglect V2V applications[1]. Table I provides an overview of the different applications and their characteristics. It uses the following questions for evaluating the communication characteristics of these applications:

- When? Should data be exchanged online (i.e., while driving) or offline.
- Priority? High versus low priority.
- What? High, medium, or low data load.
- How? Stream-based versus (single) file-based.
- How frequent? Constant, regularly, on demand.

[1] [1], [6] or [7] present overviews on V2V applications.

- Where? Applicability for smart cities, highways, both or independent.
- Which time frame? Are there real-time constraints or is the application delay-tolerant (e.g., due to low bandwidth) and can send the data later.
- Which direction? Bidirectional or unidirectional communication?

Cloud-based V2I applications serve different purposes such as routing, driver convenience, or safety of traveling. In the following, we briefly describe the most common applications[2].

Dynamic routing applications influence the routes of vehicles based on the current traffic conditions. V2I communication allows to send the current location of vehicles as well as their velocity to the cloud. This may be complemented by information about the current weather conditions or planned construction sites.

For smart cities, especially the real-time navigation optimized based on traffic flow information gained through V2I communication might be an interesting approach. Further, the search for parking lots can be optimized using V2I communication. This adds another benefit as drivers can search for suitable parking location for their vehicle. Also, drivers of electric vehicles can search for charging stations easily using cloud-based V2I communication.

Several cloud-based V2I applications target the maintenance of vehicles. This includes monitoring of the vehicle status by sending different driving parameters or sensor measurements. On the one hand, this can be used to indicate to the driver any issues with the vehicle. On the other hand, car manufacturers can use this information when planning the next facelift of their car. Since 2018, car manufacturers must equip all new vehicles in the European Union with the eCall system for automatic emergency calls [8]. Furthermore, updates of car software can be sent over-the-air. Additionally, the cars may exchange data that are relevant for autonomous driving. Camera pictures, LIDAR / RADAR, and sensor data can be used by machine learning techniques to improve the algorithms for autonomous driving as well as to update the necessary high-quality map data. Finally, remote control of autonomous cars from a management dashboard is another possible use case for V2I communication.

Especially in combination with autonomous driving, but also for passengers of conventional cars, a number of entertainment applications have emerged. One example is the streaming of media content using dedicated V2I communication infrastructure. This also enables further economic possibilities for financing the infrastructure, e.g., by advertisements. Another application is vehicular social networking, i.e., the virtual connection of drivers. In the context of smart cities, this can be complemented by connecting drivers and passengers with local shops or restaurants. Integration of user devices may be another attractive use case, either for entertainment or for work purposes.

Several applications might also be relevant for third parties such as governments, insurance agencies, or car rental companies. Using V2I communication, the government can collect information for tolling or use the driving information of vehicles for section control, e.g., to identify cases of speeding. Insurance companies can use information about the driving behavior and the usage of the vehicle for individual charging based on profiling. Rental car providers can use such information for pricing of rental cars based on actual usage. Further, online fleet management offers companies different possibilities, e.g., control of goods or optimization of public transportation status information.

Finally, the term platooning describes the formation of multiple vehicles driving with small gap sizes of 5-10 meters enabled through communication. Whereas platooning itself is a V2V application due to the required low latency, in the iCOD project, we research the coordination of vehicles in platoons [4]. This coordination is enabled by V2I communication and a server coordinating the platooning activities to make it work in settings with a low penetration rate of platoonable vehicles in which V2V-based approaches for searching platooning vehicles might not work. Additionally, we also plan to offer an inner-city platooning approach based on loosely-coupled platoons [9].

IV. Adaptive Cloud-based Vehicular Communication

Based on analyzing the varying requirements for Car-to-Cloud communication approaches presented in the previous section, this section presents the system model we assume for adaptive Car-to-Cloud communication and discuss the associated research challenges. In principle, four transmission alternatives are feasible. First, vehicles may use stationary transmission links, e.g., while parking at home. Second, vehicles may use cellular communication networks, such as 3G, 4G, or 5G connection links. Third, it is possible to use specific V2I communication infrastructure, such as DSRC based on Wifi IEEE 802.11p or IEEE 1609. Lastly, hop-based communication is possible, i.e., establishing a communication with the cloud through intermediary V2V communication. Figure 1 illustrates the different communication types.

The mobility of vehicles requires to changes in the connection. Possible connection links can differ over time in terms of bandwidth, latency, and other parameters. This leads to the need for adaptive communication mechanisms that control and optimize the Car-to-Cloud communication for guaranteeing quality of service. Depending on the nature of the application, different quality of service levels are necessary and different parameters are important (cf. Table I). This adaptive communication approach raises different research questions and challenges:

1) Which end-to-end timing can be achieved?
2) What level of stability of a wireless connection can be assumed?
3) How to handle hand-over situations?

[2] The following text is a summary of applications presented in [6] and [7]. Due to space constraints, we do not provide the original references of these applications. For that, the interested reader is referred to [6] and [7].

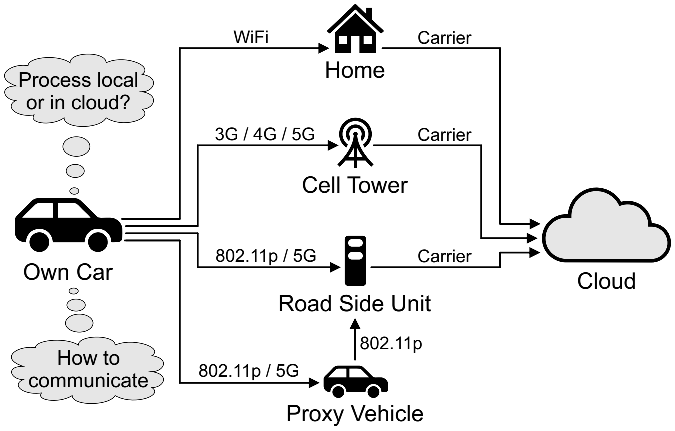

Fig. 1. Communication Types for Car-to-Cloud Transmission.

4) Which improvement of standards (and protocols) are necessary?
5) How to design an adaptive communication mechanism?

Adapting the communication mechanisms is one possibility while another is to adapt the application behavior. We want to illustrate this with a scenario for dynamic routing. In cases of a stable and fast network connection, the application would run in the cloud and use up-to-date traffic information to calculate the fastest route. However, if we assume to drive in the countryside with bad cellular coverage, it may be beneficial to download the map in advance and calculate the route onboard. This comes with the disadvantage of not having real-time traffic information to optimize the route. However, it is still possible to calculate the route even if no connection to the cloud service is possible. Accordingly, applications should have adaptive behavior to run either on the car or in the cloud. Thus, besides adaptive communication, we plan to investigate the possible application placement strategies.

This also includes functionality that is traditionally processed in the vehicle, such as the range estimation of electrical cars. Currently, the battery distance calculation for an electrical vehicle primary relies on the driving behavior. The knowledge about the remaining battery range of the vehicle can be important for the driver for planning if and when charging stops are needed. This knowledge can be leveraged to optimize the allocation of charging stations by booking an estimated time slot. The knowledge of the planned route of the vehicle provides an important factor for range estimation. The geographic topology and speed limits could also be extracted from offline maps. Beside the route, there are some additional factors that influence the power consumption of an electrical vehicle. Context information such as the outside temperature also has an impact on the range, as the batteries might have to be heated or cooled down. Without the knowledge of the outside temperature, the power consumption for achieving a proper battery temperature and therefore the range cannot be estimated well. Additionally, the wind influences the range of a vehicle as well. Headwind results in a higher power consumption while tailwind may positively affect the range of the vehicle. Such context information like the current temperature and wind cannot be provided by offline maps. The distance calculation for electrical vehicles would benefit if such information is accessible via the cloud. In smart cities, such data can be provided through an existing infrastructure of weather stations. Nevertheless, highways and rural areas would not be covered in the near future by such weather stations. Besides polling context information from the cloud, the vehicle can also be used as IoT sensor by reporting, e.g., the outdoor temperature into the cloud and making it available for other road users. Summing up, a rough range estimation is already possible with static maps. However, the cloud connectivity provides fine-grained live data to optimize arrival and battery estimations and for using the vehicles as IoT sensors.

In the context of the above applications, the following research questions and challenges arise:

1) Which amount of sensor and configuration data is required to be up-/downloaded for an application?
2) Which alternative function partitioning variants are reasonable?
3) Which calculation can be done onboard; which has to be done off-board?
4) Which applications need to run online (i.e., while driving), which can run offline?
5) When to process data?
6) How to ensure data protection and privacy?

V. Related Work

Pillmann et al. examine Car-to-Cloud communication from different perspectives [10], [11]. They integrate a Common Vehicle Information Model (CVIM) into a vehicle simulator and investigate the network traffic within Long Term Evolution (LTE) mobile radio cell. Through a newly generated Car-to-Cloud communication traffic model, the amount of data and data rate in different traffic situations like normal traffic flow or traffic jams can be modeled. Although the communication traffic model can be used to simulate the data traffic between cloud and vehicles, the adaptive aspect is not covered.

The V-Cloud architecture includes vehicular cyber-physical systems (VCPS), V2V network and V2I network layers [12]. It classifies the required communication by the communicating entities, i.e., if the communication is within a vehicle, from vehicle to vehicle, or from vehicle to infrastructure including clouds. Some use cases and benefits of CPS-enabled vehicles are discussed, but specific models and evaluations are not included in their work.

Montanaro et al. present a cloud-assisted control system architecture for platooning [13]. The functional architecture is structured in three layers, which are distributed between onboard vehicle systems, a roadside infrastructure, and a cloud. Their aim of offloading some functionality, in this case platooning coordination, into the cloud is similar to our approach. Besides the different scope, the work presents only a high-level concept.

Rémy et al. provide LTE4V2X, a framework for a centralized vehicular network organization using LTE [14], [15]. The authors evaluate the realistic traffic urban as well as highway scenarios using the ns-3 network simulator. Aspects of the investigation are overheads by control messages, packet losses, and handovers. Although the work has some similarities with our proposed vision, it is limited to LTE and the adaptive aspect is missing.

The work of Festag et al. focuses on Car-to-Car communication [16]. It is based on FleetNet, a Car-to-Car communication platform. Similar to our approach is the adaptiveness to network availability, which uses multi-hop packed forwarding over other cars if no direct connection is available.

Sallam et al. investigate the performance of routing protocols in vehicular ad-hoc networks (VANETs) through ns-3 network simulator [17]. Identical to our approach the performance is evaluated in driving scenarios generated by SUMO traffic simulator. However, this work focuses on V2V communication and covers only routing protocols.

An evaluation of Vehicle-2-Vehicle communication channel is provided by Grau et al. [18]. Similar to our planned testbed, real hardware from Cooperative Vehicle Infrastructure Systems (CVIS) has been used for measurement. This evaluation focuses on the communication channel between vehicles, and does not cover any higher protocol layers.

Bilstrup et al. investigate the medium access method of 802.11p with focus on real-time capabilities and concludes that the used CSMA/CA is not suitable to guarantee fixed transmission times [19]. Instead of this, they introduce a self-organizing time division multiple access method. Although the scope of this work is different from our approach, the communication channel usage, i.e., with several vehicles with real-time requirements is evaluated in a simulation environment.

Dressler et al. present an adaptive approach for guaranteeing latency and reliability for communication in platooning applications [20]. The approach integrates Tactile Internet concepts and interdisciplinary approaches from control theory, mechanical engineering, and communication protocol design. However, they only focus on V2V communication.

In contrast to existing works, we investigate which types of processing can be done on the vehicles and which require a connection to the cloud. Further, we integrate a mechanism for adaptive Car-to-Cloud communication.

VI. Status Quo and Future Work

Currently, we are developing a simulation environment for Car-to-Cloud communication. The environment integrates a vehicle simulator for simulation of driving and a network simulator for simulating the communication between vehicles and the cloud. We evaluated several alternatives and concluded with an in-depth analysis of Veins [21] – which integrates the SUMO traffic simulation with the OMNeT++ network simulator [22] – and the VSimRTI [23] simulation framework, which combines several traffic and network simulators using a Java-based API. For modeling and simulating the traffic in the cloud we consider using the Descartes Network Infrastructures Modeling Language (DNI) [24], which also interacts with OMNeT++. The simulation environment will be easily configurable to define the different parameters that characterize the Car-to-Cloud application (cf. Table I). This includes the configuration of various cycle times of traffic, data types (streaming of data versus single files), different velocity speeds of vehicles, switches between wireless connections and message protocols, adjusting message payloads, as well as the inclusion of communication errors and packet losses. It will cover inner-city communications as well as highway scenarios. Figure 2 shows the concept of our simulation environment.

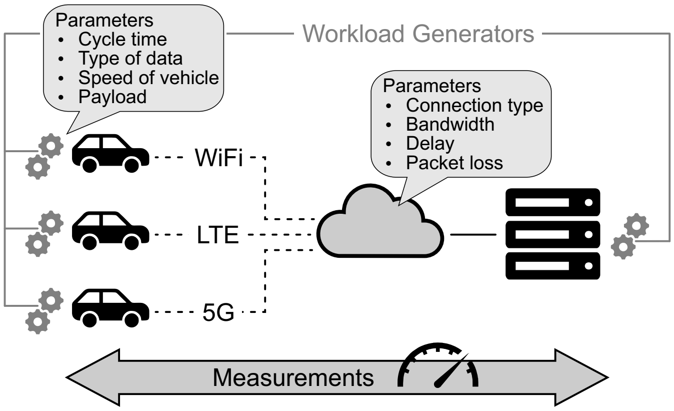

Fig. 2. Simulation Environment for Adaptive Car-to-Cloud Communication.

We will complement the simulation environment with a middleware for adaptive Car-to-Cloud communication, e.g., for choosing the best parameters and protocols for different use cases. This integrates self-adaptation [25] into the communication as well as the placement of applications.

In previous work, we already built a reconfiguration engine based on Dynamic Software Product Line (DSPL) modeling techniques for specifying the reconfiguration behavior of communication systems [26]. There we showed that the reconfiguration behavior of a communication system can be specified using modeling techniques from the DSPL domain. This allows to reuse the reconfiguration logic. Additionally, domain experts who may not be familiar with building reconfigurable systems are enabled to model the reconfiguration behavior of a communication system. Further, context-dependent reconfigurations can be easily specified. The controlled system – in the context of this paper this is the car – sends sensor information to the reconfiguration engine. Then, the reconfiguration logic analyzes the sensor data and plans a reconfiguration if needed. Finally, the reconfiguration is sent to the controlled system which then reconfigures itself accordingly.

Applying a central approach using a SAT solver in [26] limited its applicability to centralized boolean problems for turning features on or off. In recent work, we increased the capabilities of the reconfiguration logic by using a Mixed Integer Linear Programming solver for planning [27]. This directly enhanced the possibilities for specifying the reconfiguration behavior, as numeric value attributes can be used. Additionally, it allows us to address the issue of achieving certain quality goals of nonfunctional properties such as latency or bandwidth. Also, we used machine learning for learning the influence of system configurations on the values of nonfunctional goals. This allows to optimize the system configuration towards a nonfunctional goal. Additionally, multiple goals can be specified and weighted for balancing conflicting goals such as transmission speed and energy consumption individually.

For this use case, we require the integration of a middleware for adaptive communication. In [28], we describe the challenges for building a self-adaptive communication middleware. The challenges correlate to this paper's Car-to-Cloud use case:

1) How can the reconfiguration logic be distributed?
2) How should interfaces from the reconfiguration logic to the actual communication mechanisms look like?
3) How to distribute the reconfiguration knowledge in the system?

Currently, we address these challenges through applying our existing reconfiguration logic in the Car-to-Cloud use case.

In addition to the simulation environment, we plan to apply our approach in a testbed for benchmarking in cooperation with an industry partner. The aim of the testbed is the verification of the simulation results as well as executing tests on real hardware, including real driving scenarios. For the testbed, the involved components are implemented as software artifacts which can emulate the real hardware components. This allows to independently replace the emulated parts in the communication path by hardware, while other parts remain as software artifacts. Benefits of this approach are the isolated benchmarking of dedicated components such as different cloud providers, and the assumption of deterministic behavior of some components, e.g., the mobility network. The emulated components can be implemented on different granularity. For example, the car could either be implemented as one communicating entity or as different components, connected through Ethernet or bus systems like CAN, LIN, RlexRay, and MOST.

VII. Conclusion

In this paper, we outlined our vision of an adaptive Car-to-Cloud communication middleware. We presented an overview of possible applications and their characteristics. As these new applications have different communication requirements, we outlined the associated research challenges and related work in the field. Finally, we described our existing and current work on adaptive Car-to-Cloud communication for making these new applications possible using different existing communication infrastructures.

Acknowledgment

This work has been funded by the German Research Foundation (DFG) as part of project A4 within the Collaborative Research Center (CRC) 1053 – MAKI and under grant No. KO 3445/18-1.

References

[1] T. Willke, P. Tientrakool, and N. Maxemchuk, "A Survey of Inter-Vehicle Communication Protocols and Their Applications," *IEEE Comm. Surveys & Tutorials*, vol. 11, 2009.

[2] J. Barrachina, P. Garrido, M. Fogue, F. J. Martinez, J. Cano, C. T. Calafate, and P. Manzoni, "Road Side Unit Deployment A Density-Based Approach," *IEEE ITSM*, vol. 5, 2013.

[3] M. L. Sichitiu and M. Kihl, "Inter-Vehicle Communication Systems: A Survey," *IEEE Communications Surveys Tutorials*, vol. 10, 2008.

[4] C. Krupitzer, M. Breitbach, J. Saal, C. Becker, M. Segata, and R. Lo Cigno, "RoCoSys: A Framework for Coordination of Mobile IoT Devices," in *PerComW*, 2017.

[5] S. Tomforde, H. Prothmann, F. Rochner, J. Branke, J. Hahner, C. Muller-Schloer, and H. Schmeck, "Decentralised Progressive Signal Systems for Organic Traffic Control," in *SASO*, 2008.

[6] R. Coppola and M. Morisio, "Connected Car Technologies, Issues, Future Trends," *ACM Computing Surveys*, vol. 49, 2016.

[7] J. E. Siegel, D. C. Erb, and S. E. Sarma, "A Survey of the Connected Vehicle Landscape-Architectures, Enabling Technologies, Applications, and Development Areas," *IEEE Transactions on Intelligent Transportation Systems*, vol. 19, 2018.

[8] *eCall Time saved = lives saved*, accessed 2018-11-29. [Online]. Available: https://ec.europa.eu/digital-single-market/en/ecall-time-saved-lives-saved

[9] C. Krupitzer, M. Segata, M. Breitbach, S. S. El-Tawab, S. Tomforde, and C. Becker, "Towards Infrastructure-Aided Self-Organized Hybrid Platooning," in *IEEE GCIoT*, Alexandria, Egypt, 2018, to be published.

[10] J. Pillmann, D. Behnke, B. Sliwa, M. Priebe, and C. Wietfeld, "Efficient and Reliable Car-to-Cloud Data Transfer Empowered by BBR-enabled Network Coding," in *IEEE VTC Fall*, Chicago, USA, 2018, to be published.

[11] J. Pillmann, B. Sliwa, J. Schmutzler, C. Ide, and C. Wietfeld, "Car-to-Cloud Communication Traffic Analysis Based on the Common Vehicle Information Model," in *IEEE VTC Spring*, 2017.

[12] H. Abid, L. T. T. Phuong, J. Wang, S. Lee, and S. B. Qaisar, "V-Cloud vehicular cyber-physical systems and cloud computing," in *ISABEL*, 2011.

[13] U. Montanaro, S. Fallah, M. Dianati, D. Oxtoby, T. Mizutani, and A. Mouzakitis, "Cloud-Assisted Distributed Control System Architecture for Platooning," in *IEEE ITSC*, 2018, to be published.

[14] G. Remy, S.-M. Senouci, F. Jan, and Y. Gourhant, "LTE4V2X LTE for a centralized VANET organization," in *IEEE GLOBECOM*, 2011.

[15] G. Rémy, S. Senouci, F. Jan, and Y. Gourhant, "LTE4V2X-impact of high mobility in highway scenarios," in *IEEE GIIS*, 2011.

[16] A. Festag, H. Fussler, H. Hartenstein, A. Sarma, and R. Schmitz, "FleetNet: Bringing Car-to-Car Communication into the Real World," *Computer*, vol. 4, 2004.

[17] G. Sallam and A. Mahmoud, "Performance Evaluation of OLSR and AODV in VANET Cloud Computing Using Fading Model with SUMO and NS3," in *IEEE ICCC*, 2015.

[18] G. P. Grau, D. Pusceddu, S. Rea, O. Brickley, M. Koubek, and D. Pesch, "Vehicle-2-Vehicle Communication Channel Evaluation using the CVIS Platform," in *IEEE CSNDSP*, 2010.

[19] K. Bilstrup, E. Uhlemann, E. G. Strom, and U. Bilstrup, "On the ability of the 802.11 p MAC method and STDMA to support real-time vehicle-to-vehicle communication," *EURASIP Journal on Wireless Communications and Networking*, 2009.

[20] F. Dressler, F. Klingler, M. Segata, and R. L. Cigno, "Cooperative Driving and the Tactile Internet," *Proceedings of the IEEE*, 2018.

[21] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE TMC*, 2011.

[22] A. Varga and R. Hornig, "An Overview of the OMNeT++ Simulation Environment," in *SimuTools '08*, 2008.

[23] B. Schünemann, "V2X Simulation Runtime Infrastructure VSimRTI: An Assessment Tool to Design Smart Traffic Management Systems," *Computer Networks*, vol. 55, 2011.

[24] P. Rygielski, "Flexible Modeling of Data Center Networks for Capacity Management," Ph.D. dissertation, University of Wurzburg, Germany, March 2017. [Online]. Available: https://opus.bibliothek.uni-wuerzburg.defrontdoorindexindexdocId14623

[25] C. Krupitzer, F. M. Roth, S. VanSyckel, G. Schiele, and C. Becker, "A Survey on Engineering Approaches for Self-Adaptive Systems," *Pervasive and Mobile Computing Journal*, 2015.

[26] M. Pfannemuller, C. Krupitzer, M. Weckesser, and C. Becker, "A Dynamic Software Product Line Approach for Adaptation Planning in Autonomic Computing Systems," in *ICAC*, 2017.

[27] M. Weckesser, R. Kluge, M. Pfannemuller, M. Matth'e, A. Schurr, and C. Becker, "Optimal Reconfiguration of Dynamic Software Product Lines Based on Performance-Influence Models," in *SPLC*, 2018.

[28] M. Pfannemuller, "Self-Adaptive Middleware for Model-Based Network Adaptations," in *Percom*, 2018.