In: Proceedings of the 16th International IEEE Annual Conference on Intelligent Transportation Systems (ITSC 2013), The Hague, The Netherlands, October 6-9, 2013

Context Modeling for Dynamic Configuration of Automotive Functions

Gereon Weiss, Florian Grigoleit, and Peter Struss*

Abstract— **Current vehicles are usually equipped with an abundance of advanced driver assistant systems. Only a limited number of them can really be active permanently. The utility of the others depends on particular context scenarios. This motivates our goal of providing the car with the means necessary to dynamically adapt the set of active functions to its current requirements. Such a context-aware system has to construct a sound model of the actual context, based on available sources of information, such as sensors. In this paper, we present a generic context modeling approach suitable for dynamic configuration of automotive functions. The context model is divided into layers of different abstraction levels to enable the system to extract relevant context information. Data abstraction is accomplished by applying qualitative modeling techniques. The proposed method is sufficiently generic and enables an easy adjustment to specific system configurations and adaptation to new functions. The demonstration of the feasibility of the proposed solution and evaluation of its effectiveness was based on a simulated prototypical system configuration. Characteristics of ADAS functions were specified and their activation was measured during norm cycle test drives. The simulations yielded to a significant reduction in average function activity of an exemplary car system. Depending on the provided context parameters, a reduction of up to 24% was achieved.**

I. INTRODUCTION

Modern cars comprise more and more software-based functionality [1]. Besides the basic functionality, such as braking, steering, and engine control, the number and variety of so-called Advanced Driver Assistance Systems (ADAS) is increasing, which aim at improving divers' comfort and the safety of driving. ADAS range from features like supporting the actual driving over warning the driver in dangerous situations to active safety systems which take over in certain situations. Since the control units executing these functions and communication between them consume energy and cause additional cost, this trend is in conflict with another goal of vehicle development, which increasingly receives attention today: reduction of energy consumption and cost, which is especially relevant to electrical vehicles. In contrast to the basic functionality, ADAS often provide functionality relevant only in specific driving situations, such as a Parking Assistant Systems (PAS) and Lane Departure Warning (LDW). Hence, activating such systems only when they are needed or expected to be useful promises savings in energy, which can be significant if an entire Electronic Control Unit (ECU) can be shut down. The potential for reducing real world power consumption of ECUs through deactivation because of certain driving criteria is shown in [2]. Within their test scenario and duration of 1180 s considering five ECUs they could save 2.8 Wh, which equals a CO2 reduction of 219 mg/km. While the idea of context-based adaptation is straightforward, its implementation is not.

In order to realize a system that dynamically (de-)activates vehicle functions requires, beneath the actual configuration, identifying the features of the current driving situation of the vehicle that determine the need and utility of these functions. There are various sensors available that provide detailed raw data on the environment, which can range from simple radar information up to complex future Car-2-X information. This is already used for ADAS and advanced environment models (like virtual horizons). However, enabling the context-sensitive usage of functionality requires a high-level representation of the context, since the relevant context attributes are rather abstract and qualitative, such as weather condition (e.g. sunny, rainy or foggy) or driving state (e.g. stopped, slow or fast). The recognition of such important context features has to be robust enough to exploit data and information that is subject to noise, errors, and contradictions. Furthermore, the intended solution aims at a moving target – not only literally because of the automotive domain, but also because the set of system functions, sensors, and available information sources is rapidly changing and also varying from vehicle to vehicle. Hence, an approach to solving this problem tailored to a specific configuration and the relevant context features is not a solution. Therefore, our work aims at a generic context model and a flexible approach to context recognition, which allows for easy extension and adaptation to advancements in technology and a specific repository of functions.

The scope of the solution is determined by some basic requirements. Firstly, the activation and deactivation of features are only acceptable for driver supporting functionality, which does not provide mandatory safety-related features like an emergency brake. Secondly, the context features determining the (de-)activation have to be persistent to some extent, because too frequent

---

* This work was partially supported by the Programme Inter Carnot Fraunhofer from the German Federal Ministry of Education and Research (BMBF) (Grant 01SF0804) and the French L'Agence nationale de la recherché (ANR) within the project RT-DESCRIBE.

G. Weiß is head of the Automotive Software Group at the Fraunhofer Institute for Embedded Systems and Communication Technologies ESK, Munich, BY Germany, (e-mail: gereon.weiss@esk.fraunhofer.de).

F. Grigoleit is a Post-Graduate Student at the Technische Universität München (TUM), Munich, BY Germany (e-mail: florian.grigoleit@tum.de).

P. Struss is head of the Model Based Systems & Qualitative Reasoning Research Group at the Technische Universität München, Munich, BY Germany, (e-mail: struss@in.tum.de).

978-1-4799-2914-613/$31.00 ©2013 IEEE

reconfiguration of the car's functionality is likely to cause an increase, rather than reduction of energy-consumption.

After revisiting related work, the paper presents the generic solution in Section III and its application to a specific, but typical set of functions in section IV. Section V summarizes the results of an evaluation of this solution. It is based on a simulated vehicle with diverse ADAS functions, which indeed provides evidence for the potential energy savings of such an approach.

II. RELATED WORK

For enabling context-awareness in systems, context modeling has already been identified as interesting research area in several work [3][4][5]. Focusing on mobile and ubiquitous computing applications, such approaches do not consider the application-specific requirements of an automotive application as necessary in the previously described intelligent vehicle scenario.

In [6], a middleware-based architecture enabling context-awareness in highly equipped vehicles, so-called smart cars, is presented. It also includes a three-layered context recognition architecture but is designed for static configurations and focuses on fine-granular context information. Hence, it does not address the implications of dynamic configurations for increasing the energy-efficiency as does our approach. An ontology-based context model for vehicles is introduced in [7]. The purpose of the model is to enable context-aware functionality and to provide context information to applications. Hence, in this approach the context-awareness is used for enhancing applications not the dynamic configuration of system functionality. Fusion of sensor and Car-2-X data for improved context information is shown in [8] for situation-aware driver assistance systems. It again aims at more precise context information for ADAS systems. With LDMs [9] future in-vehicle implementations of multi-layered context with diverse dynamics and outside information will be realized. Our approach can be seen as a next step towards more aggregated and multi-layered context as it will be provided by LDMs.

As the context is aggregated from raw data such information implies a distinct uncertainty, mainly depending on the accuracy and reliability of the sensor data. An overview on handling such uncertainty is given in [10] from which we identified utilizing two factors (cf. Section 0) to be most suitable for our scenario. For enabling dynamic configuration of a vehicle, a formal coarse-grain representation of a car's environment is needed. Qualitative Reasoning as described in [11] provides sound methods for such reasoning and context representation. It enables the formalization of intuitive knowledge and / or abstract first principle knowledge and numerical models of the physical world, see [12][13].

As can be derived from the described work, several approaches are addressing context even for automotive systems, but do not consider context for the dynamic de-/activation of vehicles' functionality for saving runtime resources like energy.

III. CONTEXT MODEL FOR DYNAMIC CONFIGURATIONS

In this section we introduce a meta-model for context and a generic approach to context recognition based on this. Our approach comprises methodology for deriving the relevant context attributes for a certain set of functions.

A. *A Generic Approach to Context Modeling*

In a first step we formally define context in the scope of our work. Therefore, we consider context, as proposed in [14] as "*…[a] set of environment states and settings that either determine an application's behavior or in which an application event occurs and is interesting to the user*".

Features of the context will be called *context influences* if they

- are *reliably recognizable* by automotive data sources (e.g. Sensors),
- contribute to determining the *usability* or *utility* of at least one ADAS,
- occur frequently enough, so that they can distinctly contribute to the context model's aim, and
- have a changing frequency that is sufficiently low, so that the system can adapt itself to altered conditions (e.g. specific driving maneuvers change too fast to be useful for reducing any energy consumption).

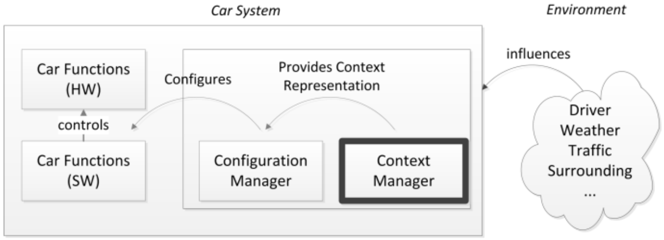


Fig. 1: Overview of context handling in a dynamic configuration scenario

Fig. 1 shows the commonly used architectural approach for dynamic configuration in a car. The *Context Manager* identifies the relevant context on which basis a *Configuration Manager* controls the activity of the vehicle functions. Within the scope of this work we focus on the *Context Manager* and its capability to aggregate and derive such context. Based on this definition, three major tasks are necessary for developing the context model. First, it has to be assessed which context features are relevant for this application. Second, we have to determine if and how these features can be observed by the system, and finally, a precise and efficient representation for the features has to be developed that allows an unambiguous configuration of the concerned car functions. To represent context in our work, we use a hybrid approach exploiting techniques from Ontology-Based Modeling [15] and Object-Oriented Modeling [16]. The determined context variables are represented as so-called *Context Attributes* with discrete domains. This way, the denoted context is divided into a number of complementary variables, each representing a distinct aspect of this context. The aggregation of all the *Context Attributes* constitutes a *Context Situation*, thus, a possible state of the car (e.g. the driving condition, cf. Table 1). Based on the *Context Situation*, a Configuration Manager (as shown in Fig. 1), is able to configure the car functions according to the recognized context.

B. *Qualitative Context Attributes*

Since runtime resources in embedded systems like automobiles are scarce, we aim for a reduced complexity of the resulting model. Therefore, we use qualitative modeling [11] to determine possible values of the *Context Attributes*.

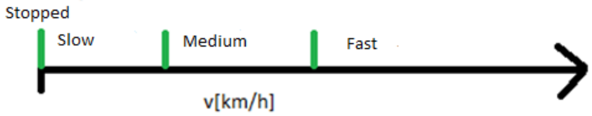


Fig. 2: Partitioning of continuous values like velocity

For this, we map the, usually, continuous values of the context influences into meaningful intervals. As explained in [14], this means portioning values into regions that affect the car system in different ways. Fig. 2 shows such an exemplary mapping of the vehicle speed *v* to interval values “Slow”, “Medium”, and “Fast”.

Too frequent changes of the recognized context model would result in constant changes of the car functions’ configuration. Thus, the number of intervals is kept as low as possible, while maintaining the desired functionality. The resulting intervals of our studies for the selected context influences are also shown in Table 1.

Additionally, if the value of a context influence is close to the threshold between two intervals, the value of the concerned attribute could change frequently. To prevent such an unstable value, the threshold between regions is modeled vaguely as shown in Fig. 3. A defined area close to the threshold allows both adjoining context values. The *Context Attribute* will not change its state until the continuous value has definitely traversed to the new region.

Enabling the integration of the described context model we introduce a three-layered architecture, as depicted in Fig. 4. This allows achieving a certain platform and manufacturer independence and easy extension and adaptation to different configurations of functions, sensors etc. The Data Layer contains low level data received from the sensor or data source. The Information Layer deals with more abstract data, called *Context Information* that is usually determined or computed from the data of the Data Layer. The Context Layer contains the *Context Attributes* obtained from the qualitative mapping described before.

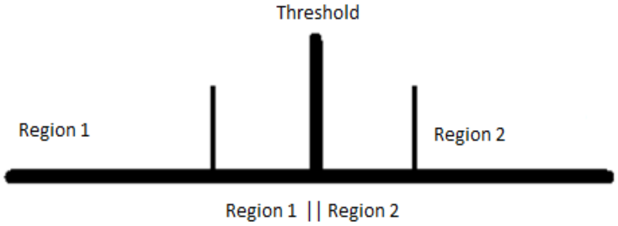
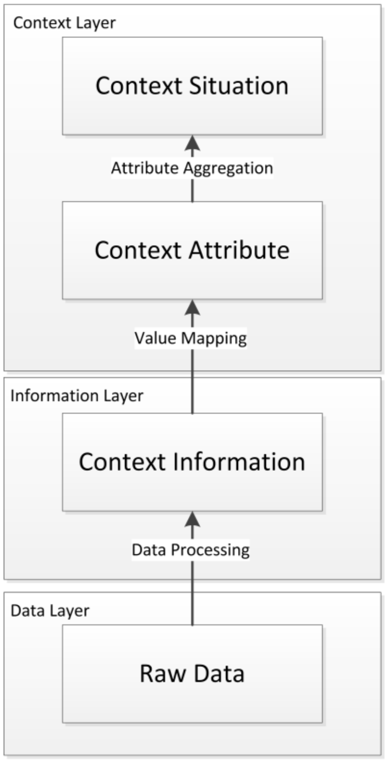


Fig. 3: Overlapping intervals

Fig. 4: Layered context model architecture



C. *Validation of Context Information*

To strengthen the confidence in the recognized context model and, thus, to ensure that the self-adaptation of car functions is always based on the correct context, we provide additional means to handle imprecise data and uncertainty. In our model, the procedure most prone to uncertainty based errors is the qualitative mapping from *Context Information* to *Context Attributes*. Naturally, wrong mapping decisions occur generally close to the interval thresholds. This is especially a problem in case of context information provided by more than one data source, without a sophisticated data fusion algorithm. Therefore, we propose a procedure based on two factors. The first, called *Tendency* is based on the observed tendency of the previous values. Here, the Context Manager has to prefer the value from the data source that correlates better with previous values. In the example given by Fig. 5 the *Sensor 2* value is preferred. It has the better tendency with the previous values X_{t-2} and X_{t-1} than the value of *Sensor 1*. The second factor is *Evidence*. *Evidence* is a factor that shows the confidence the system has into the precision and reliability of a data source. This can either be a static factor, based on parameters like precision of a sensor or error-proneness of an algorithm, or a dynamic factor. The

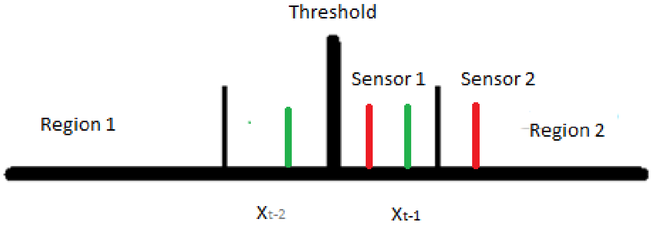


Fig. 5: Tendency-based decision

latter in addition to such a static part incorporates a dynamic part based on the current context. An example is a computer vision algorithm which works less reliable the darker the surrounding is. Utilizing the two proposed factors, it is possible for the system to easily and safely choose between two different sensor values. To avoid the forwarding of wrongly identified attribute values, we propose to additionally cross-check the values of different attributes against pre-defined rules. Therefore, the model includes a number of implications, which prohibit certain impossible or improbable combinations of attribute values. For instance, we included the rule that “Parking” and “Fast Driving” are context values that are mutually exclusive.

D. *A Methodology for Applying the Context Model*

Based on the previously introduced concepts we outline in the following the applicability of our approach. Therefore, we describe the methodology for building a context model for a specific set of functions. Additionally, the capability of our approach supporting an evolution of such a system is investigated by the exemplary use case of adding a new sensor and its related context to the given car.

As mentioned in the previous sections, the actual car system, and thus, the context model depend on a variety of factors and variations. The major differences are the applied technologies, e.g. using radar or Lidar for measuring distance, installed equipment and the individual characteristics of data sources and car functions (e.g. specifications of ADAS functions). These factors have to be taken into account when implementing the Context Manager for a specific car. Furthermore, it has to be expected that already implemented context models will be modified, either adding or removing car functions. For implementing the context model framework on a specific car system the following steps are required. First, the activity implications need to be adapted based on the specifications of the ADASs and their sensors. Second, the mapping functions based on the available context information have to be defined. Last, validation and decision algorithms have to be implemented if multiple sources allow the derivation of same information.

Since maintainability of automotive functions and thus of the context model as well is an important factor, we evaluate the integration of new car functions and sensor information. Generally, the integration of new components has to be differentiated between sensors and car functions. For new sensors, it is necessary to asses which context information and, thus, which *Context Attributes* are affected by the change and, if necessary to modify the according algorithm. For example, in the case radar is replaced by a Lidar system. Such a sensor offers higher precision and longer ranges, but is less reliable under weather conditions like fog. Hence, the sensor change results in different *Evidence* values, but not in a structural modification of the context model. This would only be the case if sensors necessary for distinct context recognition are removed. In this case it will be required to modify the context model as presented before.

If a new car function is integrated into an existing system or if the characteristics of an existing function are changed, the procedure is as follows. First, the implication of activity (s. Section III.A) has to be produced, on the basis of the functions’ specifications. If this implication can be created based on the existing context model, the new function can be integrated without further actions necessary. If this is not the case the context model has to be modified in a way that treats the modification as an exception so other functions are not affected. For example, a new ACC to be added operates only in speeds up to a limit, which is not captured by the present context value mapping. Hence, it is necessary to add a new context value (e.g. “very fast”) to the *Context Attribute “*Driving Condition”. If this context value is recognized the ACC would not be activated in conformance with its specification. When updating the intervals of a Context Attribute a trade-off has to be made between fine-granular value ranges and number of intervals. Additionally, to avoid further changes in the selection of other functions, e.g. like for the Blind Spot System, the Configuration Manager can simply react on new values for unaffected functions like for the already used value. For instance, “very fast” would be treated like “fast”.

IV. APPLICATION OF THE APPROACH

For evaluation purposes the presented context modeling and methodology have been applied for a specific intelligent vehicle.

TABLE 1: CONTEXT ATTRIBUTES AND VALUES

Context Attribute	**Characteristics**	
	Values	***Used Data Sources***
Weather	Clear	Camera
	Rain / Snow	Rain Sensor
	Fog	Weather Information
Scenario	Interstate	Geo Information System / Navigation
	Highway	
	City	Camera
	Parking	
Surrounding	Undisturbed	Camera
	Construction Site	Radar / Lidar
	Accident / Obstacle	Geo Information System
Traffic	Free	Traffic Information Channel
	Dense	Radar / Lidar
	Congestion	Camera
Visibility	Free	Camera
	Dark	
	Limited	Brightness Sensor
	Direkt Sun / Too Bright	
Driving Condition	Fast	Tachometer
	Medium	
	Slow	Navigation
	Stopped	

To this end, we analyzed context variables that are observable by available automotive sensors and, which, are also considered relevant for context-based self-adaptation of the configuration. The scenario assumed the 8 ADAS, which are typical of upper class cars and listed in Table 2 along with their activation conditions. The context attributes, which occur in these conditions, are presented in Table 1, which also shows the potential sources of data and information for determining the context attributes.

V. SIMULATION AND EVALUATION

A. *Simulation Setting*

In this section, we present the simulation and evaluation of the sample car and context model introduced in Section

IV. For more detailed descriptions of the simulation setup we refer to [17]. To exploit our concept and to analyze the possibility of optimizing the number of active functions, thus, reducing the energy consumption, we created an implementation within an automotive-specific SystemC-based simulation framework [18].

TABLE 2: ADVANCED DRIVER ASSISTANCE SYSTEMS

System	**Characteristics**
	Definition
Adaptive Cruise Control	$Scenario: Highway \lor Scenario: Interstate) \land (DrivingCondition: Fast) \land (Surrounding: \mathrm{Undisturbed}) \land (\neg Traffic: Congestion) \land (\neg Weather: Rain \lor \neg Weather: Fog) \Rightarrow Operation(ACC)$
Blind Spot System	$(Scenario: Highway \lor Scenario: Interstate) \land (DrivingCondition: Fast) \land (Surrounding: \mathrm{Undisturbed}) \land (\neg Traffic: Congestion) \land (\neg Weather: Rain \lor \neg Weather: Fog) \Rightarrow Operation(BSS)$
Light Control System	$(\neg Scenario: Parking) \land (\neg Visibility: Clear) \Rightarrow Operation(LCS)$
Lane Departure Warning	$(Scenario: Highway \lor Scenario: Interstate) \land (\neg DrivingCondition: Stopped) \land (Surrounding: \mathrm{Undisturbed}) \land (\neg Traffic: Congestion) \land (\neg Visibility: Limited) \land (\neg Weather: Rain \lor \neg Weather: Fog) \Rightarrow Operation(LDW)$
Junction Assistant System	$(Scenario: City \lor Scenario: Highway) \land (\neg DrivingCondition: Fast) \land (\neg Visibility: Limited) \land (Weather: Clear) \Rightarrow Operation(JAS))$
Night Vision System	$(Scenario: Highway \lor Scenario: Interstate) \land (Surrounding: Undisturbed) \land (Traffic: Free) \land (Visibility: Dark) \land (Weather: Clear) \Rightarrow Operation(NVS)$
Parking Assistant System	$(Scenario: Parking) \land (DrivingCondition: Stopped \lor DrivingCondition: Slow) \Rightarrow Operation(PAS)$
Traffic Sign Recogntion	$(\neg Visibility: Limited) \land (\neg Weather: Rain \lor \neg Weather: Fog) \Rightarrow Operation(TSR)$

The main focus of this evaluation is the context modeling. Hence, for the adaptation of the configuration of software functions we use a straightforward configuration system which activates and deactivates functions depending on their usability. To determine the usability of a car function, we implemented the usage implications as described in Table 2 which reflects the availability based on common ADAS specifications. The configuration and, thus, the result of the simulations, largely depend on the context of the car. Similar to the traditional *Norm Cycle* as test scenario our simulation environment consists of 33% city, 33% highway, and 33% interstate. Other parameters, such as average velocity or probability of rain, were increased stepwise over a series of otherwise identical simulations. Through this, we can identify the influences of *Context Attributes* and values on the number of active functions. In a second step we evaluated our context modeling approach with respect to its potential of reducing the number of active functions. For this purpose we simulated systems without any context model, one with simple location-awareness, and with our introduced context modeling approach.

B. *Simulation Results*

Fig. 6 shows the results of simulation runs with an increasing likelihood of rain on the horizontal axis and percentage of function activity / number of active functions during the simulation on the vertical axis. The three graphs show the activity of an ACC (Adaptive Cruise Control), LCS (Light Control System) and the overall activity of all simulated ADAS functions. Here, it can be derived that the activity of the ACC and the average number of active functions decreases with an increased likelihood of rain. The average number of active functions decreases significantly with more than 30% likelihood of rain, as many of the simulated ADAS functions do not work with respect to their specification under the simulated conditions. Such an influence of a single context value on the overall function activity does not need to have the same effect on single ADAS functions. For the depicted activity of the LCS a rise in activity with the likelihood of rain can be seen, as more often an exterior light is needed in rainy weather conditions.

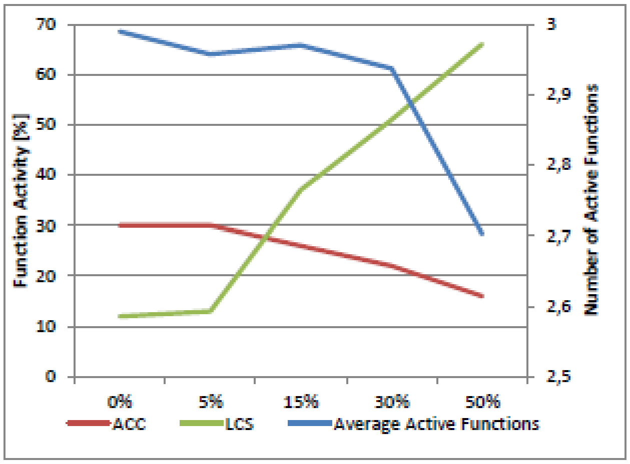


Fig. 6: Influence of *Context Value* “Rain”

Since the average number of active functions decreases depending on the *Context Value* “Rain” we analyze the effectiveness of the context recognition in a subsequent step. To this end, we compare the function activity of three different types of dynamic configurations. First, we used a conservative approach, in which the function activity is based on a single variable, e.g. the ACC depending on the car’s velocity. Second, we extended the first system with a very basic context model consisting of the *Context Attributes* “Velocity” and “Scenario” (cf. Table 1) only, and, finally, we simulated the introduced context approach, as presented in this paper. For these simulations we used the frequencies of occurrence based on the simulations shown in [17].

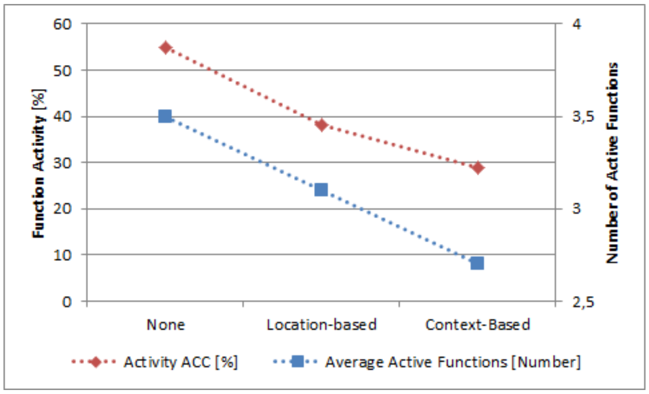


Fig. 7: Influence of context on the average number of active functions

The results depicted in Fig. 7 clearly show that the activity rate of all simulated car functions is decreased with the use of our introduced context-based approach. The same effect, but with a lower magnitude, can be observed for the purely location-based dynamic configuration. All in all, within our synthetic scenarios we measured a 12% reduction compared to the solely location-based configuration, and a 24% reduction of function activity compared to a static traditional system.

C. Discussion

A context-based dynamic configuration enables a car system to reduce the average number of active car functions. As expected and depicted in Fig. 7, the quantitative reduction is largely dependent on the simulation parameters. The actual reduction depends on the simulated environment and specification of the ADAS functions. It can also be derived that the resulting reduction of function activity increases for lower frequencies of occurrence until it approaches function activity of a purely location-based system. Within the scope of this study we could not simulate the effectiveness of the concept in a concrete car, nevertheless, we could highlight that our context-based approach is capable of reducing the number of active functions effectively, even more than just by a location-based activation. With system architectures allowing the shutting down of ECUs and our introduced context model, the in-vehicle energy consumption could be reduced, as shown in [2]. The impact of our approach in future centralized architectures with sensor data fusion poses an interesting field for future studies.

VI. CONCLUSION AND FUTURE WORK

Enabling context-aware configuration of automotive functions in intelligent vehicles holds great potential. In this paper we introduced a novel context modeling approach that allows the context-based activation and deactivation of automotive functions. It incorporates a layered approach which can be tailored to different car variants and allows validating identified contexts. This model was applied to a sample today’s intelligent vehicle system equipped with diverse Advanced Driver Assistance systems. In different simulation scenarios we could demonstrate that our context modeling is feasible for dynamic configuration of automotive functions. Moreover, we could show that our context-based configuration allows a reduction of up to 24% of function activity in the simulated scenarios. Such potential deactivation of unused functions can further result in higher energy-efficiency of the in-vehicle electronics.

While we were able to show the feasibility and theoretical worth of the concept, an implementation and evaluation in a real car system is expected to give in-depth results on the energy-saving potential through our context-based dynamic configuration approach. Furthermore, our approach may also be integrated with other context-aggregation approaches, like Local Dynamic Maps (LDMs) [9] in future Intelligent Transportation Systems.

REFERENCES

[1] K. Venkatesh Prasad, M. Broy, and I. Krueger, "Scanning Advances in Aerospace & Automobile Software Technology," Proceedings of the IEEE , vol.98, no.4, pp.510,514, April 2010

[2] C. Schmutzler, M. Simons, and J. Becker, "On demand dependent deactivation of automotive ECUs," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2012 , vol., no., pp.69,74, 12-16 March 2012

[3] T. Strang and C. Linnhoff-Popien, “A Context Modeling Survey” UbiComp, 2004

[4] C. Bettini et al., ”A survey of context modeling and reasoning techniques,” Pervasive and Mobile Computing, Volume 6, Issue 2, April 2010

[5] P. Makris, D. N. Skoutas, and C. Skianis, "A Survey on Context-Aware Mobile and Wireless Networking: On Networking and Computing Environments' Integration," Communications Surveys & Tutorials, IEEE , vol.15, no.1, pp.362,386, First Quarter 2013

[6] J. Sun, Y. Zhang, and K. He, "Providing Context-awareness in the Smart Car Environment," Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on , vol., no., pp.13,19, June 29 2010-July 1 2010

[7] M. Madkour and A. Maach, “Ontology-based context modeling for Vehicle Context Aware Services,” Journal of Theoretical and Applied Information Technogloy, 2011

[8] M. Rockl et al., "Hybrid fusion approach combining autonomous and cooperative detection and ranging methods for situation-aware driver assistance systems," Personal, Indoor and Mobile Radio Communications, PIMRC 2008. , vol., no., pp.1,5, 15-18 Sept. 2008

[9] Z. Papp, C. Brown, and C. Bartels, "World modeling for cooperative intelligent vehicles," Intelligent Vehicles Symposium, 2008 IEEE , vol., no., pp.1050,1055, 4-6 June 2008

[10] A. Ranganathan, “Reasoning about Uncertain Contexts in Pervasive Computing Environments,” Pervasive Computing, 2004

[11] K. D. Forbus, “An Introduction to Qualitative Modeling,” Northwestern University, 2003

[12] B. Bredeweg et al., “Towards a structured approach to building qualitative reasoning models and simulations,” Ecological Informatics 3, 2008

[13] P. Struss and M. Sachenbacher, “Significant Distinctions Only: Context-dependent Automated Qualitative Modeling,” 13th International Workshop on Qualitative Reasoning, 1999

[14] G. Chen and D. Kotz, “A Survey of Context-Aware Mobile Computing Research” IEEE, 2000

[15] R. Krummenacher and T. Strang, “Ontologiy-Based Context Modeling,” Workshop on Context-Aware Proactive Systems, 2007

[16] B. Bouzy and T. Cazenave, “Using the Object Oriented Paradigm to Model Context in Computer Go,” First International and Interdisciplinary Conference on Modeling and Using Context

[17] F. Grigoleit, “Context Modeling for Dynamic Configuration of Automotive Functions. Master's Thesis,” Tech. Univ. of Munich, Comp.Sci. Dept. 2012

[18] M. Zeller et al., “Co-simulation of adaptive automotive embedded systems,” IEEE / IFIP International Conference on Embedded and Ubiquitous Computing (EUC), 11.-13.12.2010, Hong Kong