Control-over-the-air (COTA) for automotive comfort functions

Martin Sommer, Luca Seidel and Eric Sax
Institut fuer Technik der Informationsverarbeitung (ITIV)
Karlsruhe Institute of Technology (KIT)
Engesserstr. 5, 76131 Karlsruhe, Germany
E-Mail: {ma.sommer, luca.seidel, eric.sax}@kit.edu

***Abstract*—The switch to service-oriented architectures in the automotive sector offers the opportunity to address key industry challenges, such as the reusability of hardware and software components, and the more efficient integration and maintainability of digital functions. The service-oriented architecture also provides the basis for the implementation of cloud-based vehicle software components. Offloading to the cloud is an upcoming option for compute-intensive functions to satisfy high resource demands. Advanced control mechanisms, such as model predictive control represent a particular component that requires high computing resources. Against this background, the overarching aim of this study is to investigate the control-over-the-air approach using the example of heating,ventilation, and air-conditioning control of battery electric buses. For this purpose, a hardware setup was used to assess the suitability of cloud-based control and investigate the need for a fallback in the vehicle and potential energy savings through offloading.**

***Index Terms*—Service-oriented architecture, Cloud, Offloading**

I. Introduction

A modern premium vehicle has 150 electronic control units (ECUs) [1] that monitor driving operations and provide various features of comfort and driver assistance.

These control units are installed in vehicles with lifecycles of up to 35 years for buses and coaches [2]. This means that statically installed computing and memory resources in vehicles will have to satisfy future software requirements for several decades. Considering the rapidly increasing scope of vehicle technology driven by software features, it can be assumed that this assumption is not tenable. Instead, as in the smartphone sector, software updates are no longer offered at all or in a reduced form from a defined version onward because compatibility is not guaranteed. An approach to maintaining hardware and software consistently up-to-date throughout the lifecycle is the '(re-)configurable vehicle architecture' [2], in which hardware/software components can be replaced or added throughout the lifecycle.

Another option is to extend electrical/electronic (E/E) architecture to the cloud. This extension involves using the cloud's scalable resources to execute software components (SWCs), the output of which can in turn be used in the vehicle. The feasibility of offloading a SWC to the cloud depends in particular on safety, real-time constraints, and reliability. Thus, SWCs with automotive safety integrity level (ASIL) assigned are not feasible for offloading, as described in a previous work [3], but could become feasible in the future with ever-improving mobile radio technologies.

The heating, ventilation, and air conditioning (HVAC) control, especially in battery electric bus (BEB), proves to be a suitable SWC in the aforementioned work. This is particularly evident because it has a very high contribution to the total energy consumption of the vehicle. Rösch et al. [4] identified HVAC as the second-largest consumer in BEB, with an average contribution of 37 % to total energy consumption. If control-over-the-air (COTA) makes it possible to save energy at this point, then this has a direct effect on the range and can thus increase the area of application of BEB. In addition to reducing energy consumption in the vehicle, cloud offloading can also shorten the execution time and save hardware costs. Partially quantifiable advantages include a high degree of flexibility in software adaptation and release management in the cloud, because the computing resources of individual vehicles do not have to be considered. In addition, the central cloud approach can implement learning systems across fleets. The following research questions regarding COTA for HVAC, which to the best of our knowledge, have not been addressed yet, arise:

RQ-1 Is it necessary to have a fallback controller in the vehicle or can the corresponding hardware and software costs be saved?

RQ-2 Can the COTA approach contribute to energy savings both at ECU level and by leveraging cloud resources to execute more advanced control algorithms than those feasible on in-vehicle hardware?

II. Background

A. Service-oriented architecture (SOA)

Automotive E/E architectures represent systems of systems that incorporate software, hardware, and mechanical components. Traditional distributed architectures are characterized by tightly coupled hardware and software, with numerous ECUs dedicated to specific functionalities [5]. However, the emergence of SOA introduced a paradigm shift by decoupling application software from the underlying hardware, thereby enabling greater scalability and flexibility.

In this new paradigm, the roles of middleware and operating systems (OSs) have become increasingly critical. Middleware, traditionally limited to facilitating communication between

ECUs, now serves as the backbone for abstracted and virtualized communication across a vehicle's architecture and with external cloud and backend systems [6].

B. HVAC in electric city buses

Table I presents the most common heating, ventilation, and air conditioning (HVAC) system combinations. Roof-mounted heat pumps can switch between the heating and cooling modes by reversing the refrigerant circuit or implementing air reversal. The coefficient of performance (COP) of heat pumps is greater than one, such that their electric energy needs are less than those of electric resistance heaters (COP $\approx$ 0.9). This makes them especially suitable for battery electric buses (BEBs), which cannot utilize residual heat from an internal combustion engine for cabin heating.
The range of BEB equipped with a heat pump, is still significantly affected by HVAC. Consequently, the range of a BEB is reduced by $22\,\%$ in the heating mode at $-15\,°\mathrm{C}$ ambient temperature compared to a diesel bus in the same scenario [7].

TABLE I: Common HVAC systems in electric buses (based on [7])

Variant	Cooling	Heating
1	Rooftop air conditioner	Electric resistance heater
2	Rooftop air conditioner	Auxiliary heater
3	Roof-mounted heat pump with cooling and heating mode (with auxiliary heater if necessary)	

C. HVAC control

The currently employed HVAC solutions are set as cascade control loops, where the outer control loop determines the control variables for the HVAC system, which, in turn, uses its internal control loop to achieve the setpoints requested by the outer loop. Classic controllers, such as two-point control and PID control, are typically used in both outer and inner control loops.

Advanced control methods such as model predictive control (MPC) are increasingly being applied in the field of building technology [8], [9]. Prototypes have also been presented in the automotive sector, and the suitability and benefits of this control have been demonstrated [10] which is why this controller has also been used for COTA.

III. Related work

Esen et al. introduced the 'Control as a Service (CaaS)' software architecture, exploring the feasibility of leveraging cloud computing for control functions in automotive systems. The proposed architecture is signal oriented and based on controller area network (CAN) messages within the concept vehicle and WiFi connection to the cloud. The implemented use case is a cloud-based throttle limitation, where the focus was placed on highlighting architectural design and the handling of network-related challenges, such as variable communication delays. Despite the inherent constraints of network reliability and latency, the authors demonstrated the potential of cloud-based control.
The work on cloud-based driver assistance by Wang et al. [11] describes a system that provides suggestions for setting actuators, such as the gas pedal for the driver, using a driver–vehicle interface (DVI). The settings are not executed automatically (e.g., by writing the values directly to the CAN bus).
A similar use case is described by Chao Yang et al. to optimize energy (for city buses) using a cloud-based framework [12]. The framework collects position and speed data from city buses with plug-in hybrid drives and initially attempts to categorize them into different route profiles (clusters). Using the route profile and real-time vehicle braking and acceleration data, a cloud-based algorithm calculates an energy-optimized strategy for distributing the torque between the electric motor and combustion engine for the remainder of the bus route.

Many other authors have proposed the use of cloud computing to optimize energy consumption [13] and/or battery life [14] using algorithms that are not suitable for vehicle deployment, such as dynamic programming. However, research has shown that the control-over-the-air (COTA) approach has only been addressed in Esen's software architecture in the throttle control use case. A service-oriented COTA approach to a comfort function has not yet been investigated.

IV. Concept & Design of COTA for HVAC

Previous work has demonstrated the potential of cloud-based control but focused mainly on specific use cases. A service-oriented COTA approach for comfort functions, such as HVAC, has not yet been explored. To enable the design and implementation of such a system, the following section defines the essential system requirements and key design aspects that must be fulfilled to ensure reliable and efficient operation.

A. System requirements

Req-1 **Error handling and recovery** The system must contain error handling mechanisms to deal with service failures and network connection failures, data errors (inconsistent data), and delays in data provision. In addition, the system must be able to restore the operation of services or entire applications after an error.

Req-2 **Time requirements and data processing** The cloud controller must calculate manipulated variables in a time window smaller than the sampling rate. Furthermore, cloud execution must provide temporal advantages over on-board execution. Owing to execution in the cloud, the time for data transmission between the cloud and the vehicle must also be considered.

Req-3 **Maintainability and extensibility** The system should have a modular structure so that the applications and their services in particular can be easily updated. The architecture should also be designed so that future extensions or adjustments can be made without refactoring.

Req-4 **Reliability** Once the controller is integrated, the system must ensure reliable climate control under varying conditions. The metrics of deviation from the setpoint temperature, as proposed by the Association of German Transport Companies in regulation 236 (VDV-236) [15] and predicted mean vote (PMV)[1], must be applied.

B. Components overview

1) Service-oriented architecture and the software orchestrator: The service-oriented architecture (SOA) is built using ROS 2[2] middleware, with a software orchestrator as the central component in the vehicle that manages services and networks. The orchestrator was published as a result of a previous study [17].
2) HVAC cloud controller: A MPC is used for the outer control loop of the HVAC system. The controller depends on the same prediction models as those used in the simulation, with the difference that Gaussian noise was added to the simulation models to represent the real conditions. All variables of the MPC scheme are listed in Table II.
3) HVAC in-vehicle fallback controller: A local PID controller is used to establish comparability in the evaluation of the COTA approach. Additionally, it was used to identify the need for a fallback controller in the vehicle (**RQ-1**).
4) Simulation models of HVAC system and bus cabin: The HVAC system consists of a roof-mounted heat pump for heating and cooling, and therefore corresponds to variant 3 of Table I. The operating state of the heat pump is modeled using a COP characteristic analogous to the work of Rösch [18] and parameterized using the data sheet of a standard heat pump[3] installed in BEB. The bus cabin was modeled based on a thermal white box model. The cabin model describes the influence of heat flows, such as fresh air entering the cabin and heat produced by passengers, on the temperature T_{Cabin}. The parametrization was carried out using the Mercedes Benz eCitaro solo bus with 3 doors[4].
5) Hardware in the loop (HiL) test-bench (s. Figure 1): The test-bench includes a backend PC (AMD Ryzen 5 3500U and 16 GB RAM) ①, a simplified E/E architecture with an UpXtreme i12[5] as a high-performance computer (HPC) ② and a Raspberry Pi 4B as an automotive ECU ③. The 3D simulator CARLA [19] was executed on a performance simulation PC ④.

V. Evaluation

The evaluation of the COTA approach is based on the requirements of Section IV and attempts to answer the research questions. The proposed reduction in execution time is achieved

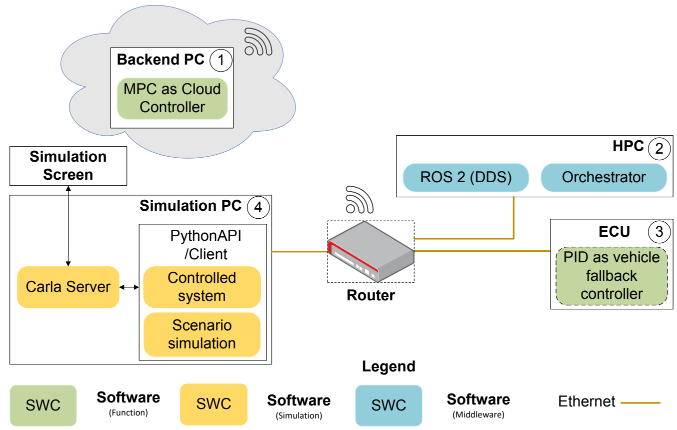


Fig. 1: HiL test-bench for COTA

TABLE II: Definition of the MPC-variables

Variable	Description	Unit
Reference variable		
$T_{\text{Cabin,set}}$	Setpoint of the cabin temp.	$^\circ C$
Manipulated variables		
$\dot{Q}_{\text{set}}$	Requested heat flow of HP	kW
$\dot{Q}_{\text{HP}}$	Measured (provided) heat flow of HP	kW
Control variable		
T_{Cabin}	Actual cabin temp.	$^\circ C$
Context data		
$N_{\text{Passenger}}$	Number of passengers	$-$
$\dot{Q}_{\text{Solar}}$	Solar radiation	kW
$N_{\text{Doors,open}}$	Number of opened doors	$-$
T_{Ambient}	Ambient temp.	$^\circ C$
V_{Vehicle}	Vehicle speed	km/h
$\dot{m}_{\text{Freshair}}$	Fresh air supply	kg/s
ϕ_{Ambient}	Relative ambient humidity	$\%$
Restrictions		
$\dot{Q}_{\text{min}}$	minimal heat flow of HP	kW
$\dot{Q}_{\text{max}}$	maximum heat flow of HP	kW

if the local execution of a function is longer than the cloud execution plus the round-trip time (RTT) from the vehicle to the cloud and back (s. Equation 1). The same applies to the energy savings at ECU level in Equation 2, which are calculated using the utilization (L_{CPU}) and thermal design power (TDP) of the CPU.

$$t_{\text{exec,local}} > t_{\text{exec,cloud}} + t_{\text{RTT}} \tag{1}$$

$$P_{\text{exec,local}} \cdot t_{\text{exec,local}} > P_{\text{idle,local}} \cdot t_{\text{exec,cloud}} + P_{\text{transfer}} \cdot t_{\text{RTT}} \tag{2}$$

$$P = \frac{L_{\text{CPU}}}{100} \cdot \text{TDP} \tag{3}$$

The evaluation scenario describes a bus route through a city in the CARLA simulation environment. The route includes four bus stops where a stopover lasts in average 16.5 s, with

[1] Predicted Mean Vote: s. ISO Standard 7730 [16]
[2] https://docs.ros.org/en/humble/index.html
[3] https://www.konvekta.de/fileadmin/user_upload/docs/datenblaetter/busse/UltraLight_500-700EM_CO2_HP_dt_engl_0722.pdf
[4] https://www.mercedes-benz-bus.com/de_LU/models/ecitaro/facts/facts-ecitaro.html
[5] https://up-shop.org/DE/up-xtreme-i12-core-i7-1270pe-16gb-ram.html

a standard deviation of 3.3 s. The change in the number of passengers and number of open doors was assumed to be a uniformly distributed random drawing.

A. Fulfillment of the requirements

Req-1 is assessed with a simulation scenario where the vehicle's connection to the cloud is unstable and keeps breaking off. Furthermore, this scenario was performed using two different COTA application models (s. also [20]). The first fallback (FB) application model has the option of switching to the PID controller in the vehicle if cloud service is not available (s. ECU ③ in Figure 1). The second is called the Only Cloud (OC) application model, which does not rely on an extra controller in the vehicle, as it switches to the open-loop mode once the cloud service is unavailable. In this case, it can profit from buffered manipulated variables (BMV) of the prediction horizon of the MPC, and thus has access to the calculated values for at least this period. Figure 2c shows two connection downtimes that the system handles by switching either to the fallback PID or to the BMV in the case of the OC application model. For comparative purposes, the same scenario was executed with on-board PID control and the OC model without cloud connectivity interruptions, named $OC_{\mathrm{NoInterruption}}$. The results in terms of cabin temperature and comfort measured in PMV are shown in Figure 2a and Figure 2b. The OC application model uses a prediction horizon of 10. Because the network cutoff was longer than the prediction horizon, BMV expired. This results in constant heat flow, which leads to an overshoot in the cabin temperature T_{Cabin}. By contrast, the FB application model tends to oscillate. When switching, the PID controller must first overcome the transient phase. This resulted in a significant drop below the set-point temperature.
Overall, the requirement can be considered fulfilled because connectivity problems are handled according to the strategies of the system.
Req-2: To analyze the temporal behavior, the above scenario is evaluated by varying the prediction horizon of the MPC. Table III lists the execution times on the local ECU and on the cloud, including the measured RTT. Based on equation 1, this corresponds to a buffer of 7.96 s for cloud-based MPC, assuming a prediction horizon of 20 and a controller sampling rate of 0.1 Hz. As illustrated in Figure 2, sufficient control performance was achieved with a prediction horizon of 10, whereby **Req-2** was fulfilled. The maximum prediction horizon for cloud-based model predictive HVAC control with Powell [21] optimization that still satisfies the sampling rate of the controller can reach $N_{\mathrm{P}} = 60$. By contrast, a local implementation reveals that the ECU employed in the HiL setup fails to satisfy Equation 1 at a prediction horizon of 30. The comparatively insignificant RTT highlights the fact that computationally intensive optimization of MPC benefits from a cloud-based implementation. Despite the comparatively strong ECU and existing transmission and latency delays, the backend execution is more than three times faster.
Req-3: The SOA facilitates a dynamically adaptable and scalable framework, allowing for runtime and post-deployment system extensions. A service-based modular system was established using an orchestrator in conjunction with over the air (OTA) update capabilities. The experimental setup was defined using the feature modeling represented in a *YAML* file. This feature model serves as the input for a variant-aware simulation configurator, which generates a simulation environment tailored to the current system configuration comprising various services [22]. The resulting simulation setup consists of a CARLA environment, a data distribution service (DDS)-based communication layer for service interaction, and a ROS 2-based service implementation. Thus, the requirement can be regarded as fulfilled.
Req-4 is ensured by **Req-1** (error handling) and **Req-2** (time requirements). The reliability of the control also depends on the accuracy of the prediction model and consideration of constraints. Consequently, reliability was assessed for different driving scenarios and operating conditions, some of which are listed in Table V. The implemented operating conditions include varying weather conditions, bus stops, and numbers of passengers.

TABLE III: Measured values of the MPC execution time and the required power on the HiL test-bench (N_{p}=20)

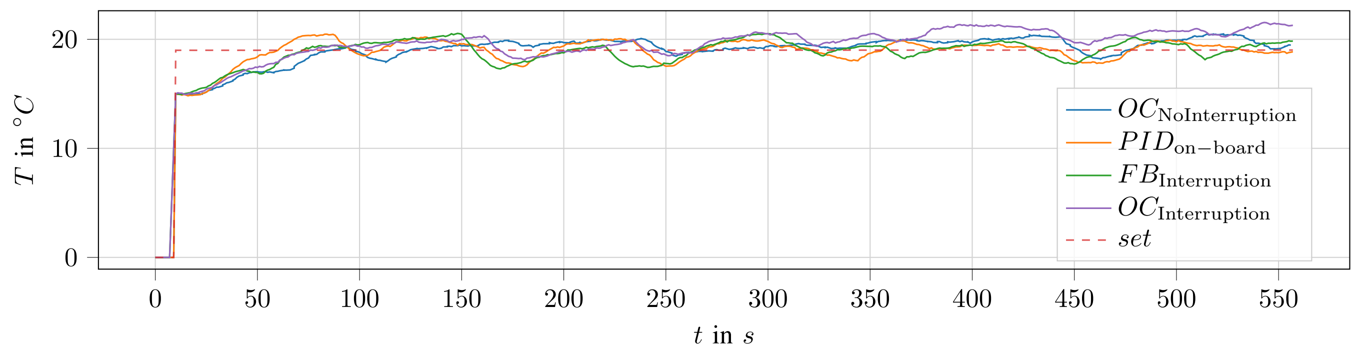
Parameter	Value
$t_{\mathrm{exec,cloud}}$	1.87 s
$t_{\mathrm{exec,local}}$	6.3 s
t_{RTT}	173.00 ms
$L_{\mathrm{exec,local}}$	26.13 %
$L_{\mathrm{idle,local}}$	0.5 %
L_{transfer}	11.17 %
$P_{\mathrm{exec,local}}$	1.96 W
$P_{\mathrm{idle,local}}$	0.04 W
P_{transfer}	0.84 W

B. Statements on the research questions

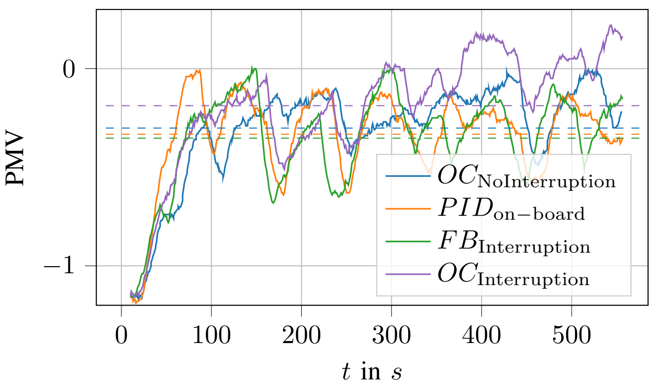
RQ-1: The need for a fallback application model with an on-board PID controller depends largely on passenger comfort. Considering the measurements in Table IV, the OC application model achieves the best value in terms of PMV.
The average control error (deviation from the setpoint temperature) for the OC application model is 1.24 K compared with 0.76 K for the FB model. The scenario with very long connection interruptions of 100 s and a connection loss of a third of the overall time window can be classified as unlikely,

TABLE IV: Measured values of the electrical energy demand and PMV of the heat pump in the scenario described in Figure 2

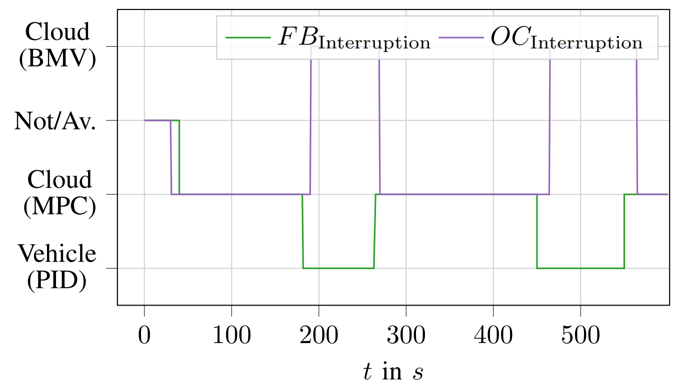
Variant	**Electrical Energy** (E)	**PMV**	**Energy savings in %**
$OC_{\mathrm{NoInterruption}}$	1238 kW s	-0.31	-12.42
$PID_{\mathrm{on-board}}$	1392 kW s	-0.33	Baseline
$FB_{\mathrm{Interruption}}$	1509 kW s	-0.35	8.42
$OC_{\mathrm{Interruption}}$	1760 kW s	-0.19	26.40



(a) Comparison of the measured cabin temperature



(b) Comparison of the Predicted Mean Vote



(c) Comparison of execution platforms

Fig. 2: Comparison of different control strategies and their results during the city bus route simulation at $\overline{T}_{\text{Ambient}} = 9.58\,°\text{C}$.

but still interesting as an extreme case. Despite harsh conditions with long interruptions of the cloud connection, the comfort requirements of VDV-236 are still met because a deviation of $2\,\text{K}$ is allowed. Therefore, the OC application model is sufficient for the COTA approach to comfort functions.

RQ-2: Energetic optimization is divided into analysis at the ECU level and the potentials that arise through optimized control.

ECU level energy savings are achieved if equation 2 is fulfilled. The TDP of the Raspberry Pi as local ECU is determined at $7.5\,\text{W}$[6]. According to the measured values in Table III savings of $12.17\,\text{W}\,\text{s}$ per control step can be realized. Based on a sampling rate of $0.1\,\text{Hz}$ and an operating time of $9.35\,\text{h}$ (extracted from [23]) of the BEB, savings of $11.62\,\text{W}\,\text{h}$ are achieved. These are negligible compared with the size of a typical traction battery in a city bus of up to $588\,\text{kW}\,\text{h}$ (s. Footnote 4).

Regarding energy savings through advanced control algorithms in the cloud reference can be made to Table IV. The OC application model without connectivity interruptions had the lowest energy consumption. This is proven in other simulations with varying ambient temperatures when comparing the COTA approach without connectivity interruptions $OC_{\text{NoInterruption}}$ to an on-board PID controller, as displayed in Table V. Simulation 1 in Table V showed that an increase of N_{P} reduced the energy consumption of HVAC, which revealed the benefits of the cloud, as this increase also entailed an increase in computing requirements which are not practicable with static ECUs in the vehicle. Again, assuming a bus operating time of $9.35\,\text{h}$ and the presented savings of $12.42\,\%$ from Table IV, $2.54\,\text{kW}\,\text{h}$ can be saved. These savings can be considered significant in terms of the battery size of the proposed BEB, which is why **RQ-2** can be positively evaluated.

VI. Discussion and Outlook

In conclusion, this study has proven the suitability of the COTA approach to the non-safety-critical HVAC control. The implementation includes cloud-based model predictive HVAC control of an electric city bus. The evaluation of the system was carried out on a HiL test-bench in the CARLA simulation environment. The microcontrollers used in the test-bench reflect a simplified modern SOA architecture with central HPC architecture. The results showed that a system with low dynamics can be controlled via the cloud. Having a fallback level in the form of an additional controller in the vehicle was not required. This can potentially eliminate the need for hardware and therefore lead to the proposed reduction of hardware cost in the vehicle if the control was previously carried out on a dedicated ECU, whereby the running costs of the cloud service provider (CSP) cannot be neglected. Energy-saving potentials do not arise at ECU level, but come with

[6] https://www.cpu-monkey.com/en/cpu-raspberry_pi_4_b_broadcom_bcm2711

TABLE V: Comparison of $PID_{\text{on-board}}$ and $OC_{\text{NoInterruptions}}$ in various simulations; Simulation time: 580 s

$\overline{\mathbf{T}}_{\text{Amb.}}$	**Metric**	$\mathbf{PID}_{\text{on-board}}$	$\mathbf{OC}_{\text{NoInterruption}}$
9.58 °C	$\mathbf{N}_{\text{P}} = \mathbf{N}_{\text{C}}$	-	10
	$\overline{\Delta T}_{\text{set}}$	0.75 K	0.65 K
	E	1392 kW s	1238 kW s
	PMV	-0.33	-0.31
9.58 °C	$\mathbf{N}_{\text{P}} = \mathbf{N}_{\text{C}}$	-	20
	$\overline{\Delta T}_{\text{set}}$	0.75 K	1.13 K
	E	1392 kW s	1080 kW s
	PMV	-0.33	-0.57
9.58 °C	$\mathbf{N}_{\text{P}} = \mathbf{N}_{\text{C}}$	-	30
	$\overline{\Delta T}_{\text{set}}$	0.75 K	0.98 K
	E	1392 kW s	1062 kW s
	PMV	-0.33	-0.57
0.94 °C	$\mathbf{N}_{\text{P}} = \mathbf{N}_{\text{C}}$	-	10
	$\overline{\Delta T}_{\text{set}}$	1.34 K	1.47 K
	E	2834 kW s	2672 kW s
	PMV	-0.06	-0.08
16.44 °C	$\mathbf{N}_{\text{P}} = \mathbf{N}_{\text{C}}$	-	10
	$\overline{\Delta T}_{\text{set}}$	0.59 K	0.54 K
	E	1046 kW s	1130 kW s
	PMV	-0.33	-0.32

the application of intelligent control mechanisms in the cloud. Future studies should focus on the scalability of cloud control. The question must be answered as to whether a large number of vehicles can be operated using a single cloud-based controller, thus limiting CSP costs.

Acknowledgment

This publication is based on the research project SofDCar (19S21002), which is funded by the German Federal Ministry for Economic Affairs and Climate Action. The authors would also like to thank student Laurin Koch for his enthusiastic work on this topic.

References

[1] Motortrend, “Coding the car - the leaders driving the software defined vehicle,” 2023. [Online]. Available: www.motortrendgroup.com/wp-content/uploads/2023/09/UPDATED-Coding-The-Car-EBook-Download.pdf

[2] H. F. Stoll, “Die (re-)konfigurierbare Fahrzeugarchitektur,” Ph.D. dissertation, Karlsruher Institut für Technologie (KIT), 2021.

[3] M. Sommer, D. Baumann, T. Rösch, F. Dettinger, E. Sax, and M. Weyrich, “Process for the identification of vehicle functions for cloud offloading,” in *Intelligent Computing*, K. Arai, Ed. Cham: Springer Nature Switzerland, 2024, pp. 596–608.

[4] T. Rösch, S. Raghuraman, M. Sommer, C. Junk, D. Baumann, and E. Sax, “Multi-layer approach for energy consumption optimization in electric buses,” in *2023 IEEE 97th Vehicular Technology Conference (VTC2023-Spring)*, 2023, pp. 1–6.

[5] A. Vetter, P. Obergfell, H. Guissouma, D. Grimm, M. Rumez, and E. Sax, “Development processes in automotive service-oriented architectures,” in *2020 9th Mediterranean Conference on Embedded Computing (MECO)*, 2020, pp. 1–7.

[6] O. Burkacky, J. Deichmann, G. Doll, and C. Knochenhauer, “Rethinking car software and electronics architecture,” *McKinsey & Company*, vol. 11, 2018.

[7] D. Jefferies, A. K. Tu-Anh Ly, and D. Göhlich, “Energiebedarf verschiedener Klimatisierungssysteme für Elektro-Linienbusse,” in *Deutsche Kälte- und Klimatagung*, 2015.

[8] M. Killian, B. Mayer, A. Schirrer, and M. Kozek, “Verteilte kooperative modellprädiktive Temperaturregelung für komplexe Gebäude,” *e & i Elektrotechnik und Informationstechnik*, vol. 8, no. 132, pp. 474–480, 2015.

[9] S. Freund, “Modellbasierte prädiktive Regelung komplexer gebäudetechnischer Anlagen zur Optimierung der Energieeffizienz und des Komforts,” Ph.D. dissertation, Technische Universität Hamburg, 2023.

[10] Y. Xie, Z. Liu, K. Li, J. Liu, Y. Zhang, D. Dan, C. Wu, P. Wang, and X. Wang, “An improved intelligent model predictive controller for cooling system of electric vehicle,” *Applied Thermal Engineering*, vol. 182, p. 116084, 2021.

[11] Z. Wang, X. Liao, X. Zhao, K. Han, P. Tiwari, M. J. Barth, and G. Wu, “A digital twin paradigm: Vehicle-to-cloud based advanced driver assistance systems,” in *2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring)*. IEEE, 2020, pp. 1–6.

[12] C. Yang, L. Li, S. You, B. Yan, and X. Du, “Cloud computing-based energy optimization control framework for plug-in hybrid electric bus,” *Energy*, vol. 125, pp. 11–26, 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544217302840

[13] J. Shi, B. Xu, X. Zhou, and J. Hou, “A cloud-based energy management strategy for hybrid electric city bus considering real-time passenger load prediction,” *Journal of Energy Storage*, vol. 45, p. 103749, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S2352152X21014237

[14] J. Hou and Z. Song, “A hierarchical energy management strategy for hybrid energy storage via vehicle-to-cloud connectivity,” *Applied Energy*, vol. 257, p. 113900, 2020. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0306261919315879

[15] H. Birkel, P. Bronnenberg, E. Krawinkel, R. Roch, and K. Walter, “Klimatisierung von Linienbussen der Zulassungsklassen I (Stadtbus) und II (Überlandbus), für konventionell angetriebene Diesel- und Gasbusse als auch für Hybrid-, Brennstoffzellen- und Elektrobusse,” VDV-Schrift 236, Nov. 2018.

[16] International Organization for Standardization, *ISO 7730: Ergonomics of the thermal environment — Analytical determination and interpretation of thermal comfort using calculation of the PMV and PPD indices and local thermal comfort criteria*, Std., 2005, ISO 7730:2005.

[17] M. Sommer, H. Guissouma, M. Schindewolf, and E. Sax, “An Orchestrator for the Dynamic Extension of Automotive E/E Architectures to the Cloud,” in *Service-Oriented Computing*, M. Aiello, J. Barzen, S. Dustdar, and F. Leymann, Eds. Cham: Springer Nature Switzerland, 2025, pp. 24–41.

[18] T. Rösch, “Optimierung der Wärmebereitstellung in Thermomanagementsystemen elektrisch betriebener Stadtbusse,” Ph.D. dissertation, Karlsruher Institut für Technologie (KIT), 2024.

[19] A. Dosovitskiy, G. Ros, F. Codevilla, A. Lopez, and V. Koltun, “CARLA: An Open Urban Driving Simulator,” in *Proceedings of the 1st Annual Conference on Robot Learning*, 2017, pp. 1–16.

[20] J. Ruhnau, M. Sommer, F. Dettinger, D. Kraus, S. Becker, E. Sax, and M. Weyrich, “Vehicle Function Offloading: Finding the Optimal Cloud/Edge Application Model (OptiCAM),” in *Proceedings of the 2025 Future of Information and Communication Conference (FICC)*, 2025.

[21] M. J. D. Powell, “An efficient method for finding the minimum of a function of several variables without calculating derivatives,” *The Computer Journal*, vol. 7, no. 2, pp. 155–162, Jan. 1964. [Online]. Available: https://doi.org/10.1093/comjnl/7.2.155

[22] L. Seidel, H. Guissouma, A. Schmid, and E. Sax, “Variant-aware reconfiguration of automotive virtual test environments,” in *Vol.9 - Driving Simulation Conference Europe 2024 VR (DSC 2024), Driving Simulation and Virtual Reality Conference and Exhibition, Straßburg, 18th-20th September 2024*, 2024, pp. 221 – 226.

[23] F. Widmer, A. Ritter, and C. H. Onder, “Ztbus: A large dataset of time-resolved city bus driving missions,” *Scientific Data*, vol. 10, no. 1, p. 687, 2023.