User-Centric Comfort Optimization With AI-Supported Methods

Daniel Braun, Andreas Raab

DZS
AVL Deutschland GmbH
Peter-Sander-Str. 32

55252 Mainz-Kastel

Daniel.Braun@avl.com
Andreas.Raab@avl.com

Abstract: The EU-funded SmartCorners project explores user-centered comfort in vehicles through AI-driven climate control. A method is developed to train a climate control algorithm with the help of AI and is trained in a comprehensive vehicle simulation model. Beyond personalization, AI is leveraged to optimize energy efficiency. The project also addresses the application of the virtually developed algorithm in a real-world vehicle. Initial simulation results are discussed in this paper with an outlook of the upcoming vehicle studies to validate the simulation results.

1 Introduction

Vehicle cabin comfort is a multifaceted concept influenced by thermal conditions, noise, vibration and harshness (NVH) levels, as well as air quality. Importantly, comfort is highly subjective – conditions that are acceptable for one occupant may be uncomfortable for another. Thermal comfort is typically managed through parameters such as zonal temperature settings, blower speed, air distribution, and the position of the recirculation flap. Zonal climate control systems have enabled personalized thermal environments for different seating areas, allowing occupants to tailor conditions to their preferences. However, accommodating these varying and dynamic preferences presents significant challenges for both the HVAC system and its control strategies.

Contribution: 2025 FKFS Conference on Vehicle Aerodynamics and Thermal Management
15 – 16 October 2025 | Leinfelden-Echterdingen

Traditional control strategies are often not optimized to balance user comfort with energy efficiency. By integrating artificial intelligence (AI) into simulation models and training it across diverse virtual scenarios, it is possible to develop adaptive solutions that significantly reduce the effort required for software development and calibration. Furthermore, AI systems can continue to learn post-deployment in real-world vehicles, enhancing their decision-making capabilities over time. This continuous learning approach simplifies the development of intelligent controllers, eliminating the need for multiples specialized control algorithms tailored to specific objectives such as individual comfort, overall cabin comfort, or energy efficiency (e.g., maximizing driving range).

2 Methodology

The development of AI-based control systems necessitates training with high-quality data to enable reliable decision-making. Acquiring such data using physical test benches or vehicles is time-consuming and costly. This challenge can be effectively addressed by using high-fidelity simulation models. By leveraging virtual environments, it becomes possible to generate large volumes of representative data for training AI algorithms, significantly reducing the need for physical testing.

This section provides a detailed overview of the proposed methodology, including the simulation setup, data generation process, and training pipeline for the AI controller.

2.1 Plant model

The plant model represents a comprehensive simulation model of the physical system under study. A highly accurate plant model ensures that the behavior of the simulated system closely mirrors real-world dynamics, thereby enhancing the reliability of AI training and control development. Any discrepancies in model accuracy can directly impact the performance and robustness of the resulting control strategies. Therefore, special emphasis is placed on the validation and calibration of the plant model to ensure it serves as a trustworthy foundation for virtual experimentation and AI-based control design.

The simulation environment incorporates detailed models of the vehicle, HVAC system, cabin, controller, driver and passenger comfort. Each model is designed with the necessary inputs and outputs to enable seamless interaction with the AI-based control system. Figure 1 provides a comprehensive illustration of plant model and the according interfaces. This section provides an in-depth description of the thermal and HVAC systems, as well as the cabin and comfort models, as the primary focus is on optimizing energy consumption of these systems while maximizing passenger comfort.

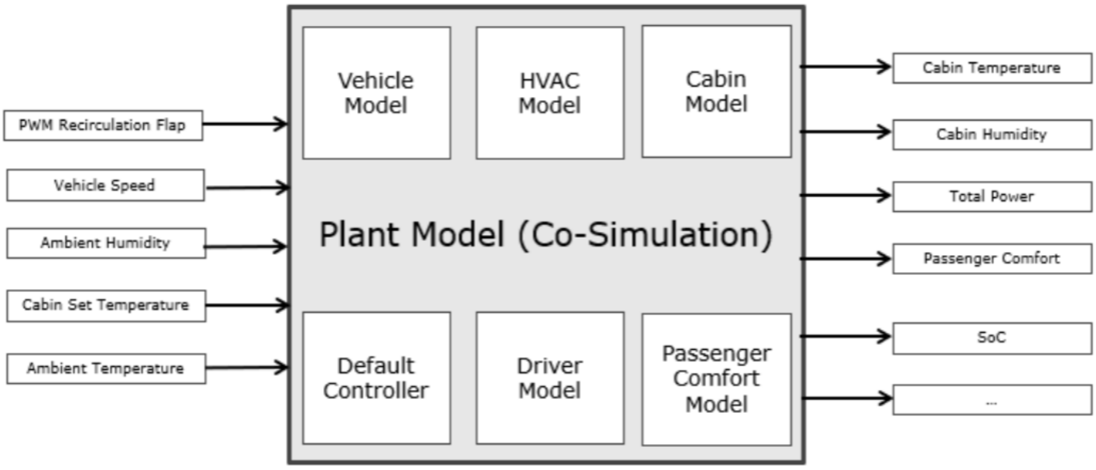

Figure 1: Overview plant model

Vehicle Thermal Management System (VTMS)

The thermal and HVAC system of the demo vehicle has a R290 and a R1234YF based refrigerant circuit and a coolant circuit. The vehicle can operate with both refrigerant circuits but in SmartCorners only R1234YF refrigerant circuit is used. Figure 2 shows the thermal management architecture consisting of the refrigerant circuit indicated in green color, the low-temperature circuit (LT-Circuit) indicated in blue color, the medium-temperature circuit (MT-Circuit) indicated in yellow color, the high-temperature circuit (HT-Circuit) indicated in red color, and the battery circuit (Bat-Circuit) indicated in grey color. The different operating temperatures of the circuit are determined based on the components placed in the circuits and their thermal requirements.

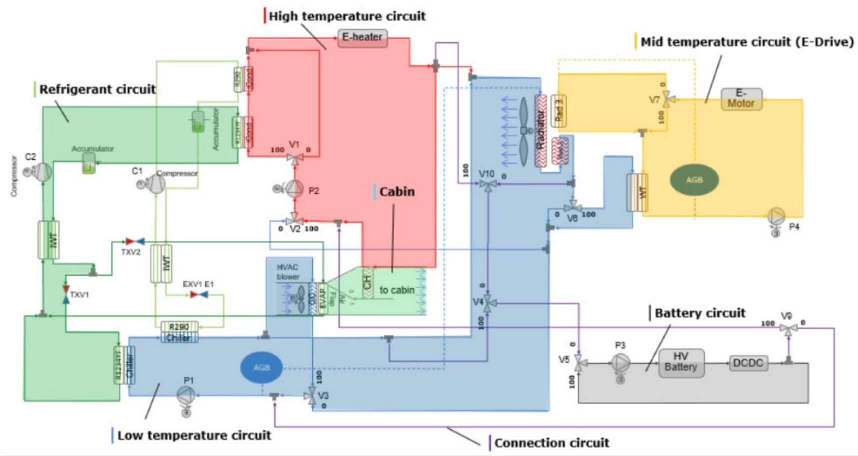

Figure 2: Layout coolant and refrigerant circuit

The plant model of the system is developed in AVL CRUISE^{TM}M, a system development software from AVL. The components are calibrated with the help of the measurement data gathered from the demo vehicle. The whole model is then validated with measurements from different operating conditions of the complete system.

Vehicle Cabin and Comfort Model

To accurately represent passenger comfort and cabin thermal dynamics, two complementary cabin models are developed:

Thermal cabin model:

A high-fidelity computational fluid dynamics (CFD) model is used to simulate airflow, heat transfer, and passenger comfort within the cabin. The model incorporates detailed geometry of the demonstrator vehicle (Mercedes-Benz B-Class), material properties of interior laminates, and additional heating devices such as seat heaters, steering wheel heating, and infrared panels. Environmental factors such as solar load and ambient conditions are also included. This model is primarily used for generating a 1D Matlab/Simulink model.

The 1D cabin model is a reduced-order representation derived from the high-fidelity 3D CFD cabin model. Its primary purpose is to provide a computationally efficient simulation environment for control development and real-time applications. The model uses aggregated thermal properties and response characteristics obtained from the 3D model. These derived parameters include heat transfer coefficients, thermal capacities, and airflow distribution characteristics.

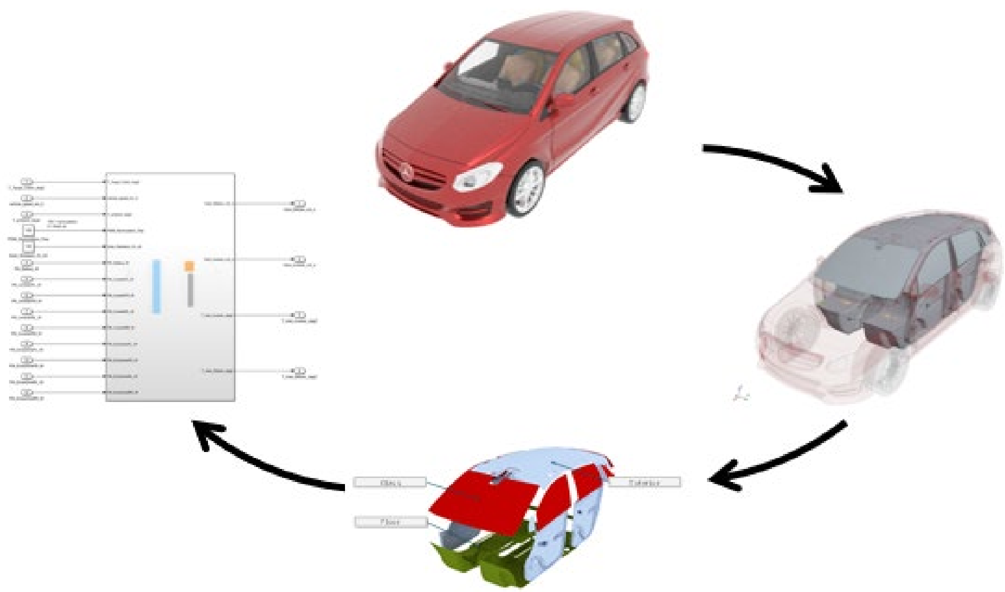

Figure 3: Body and clothing segments for EHT model

Cabin Comfort Modeling and Surrogate Model Development

To model thermal comfort in the cabin, the previously developed 3D CFD cabin model is utilized. This high-fidelity model includes detailed geometry of the demonstrator vehicle and incorporates thermal manikins to represent passengers. Each manikin is segmented into 17 body parts, enabling the calculation of local surface temperatures and heat fluxes under various operating conditions. The exact body parts and clothing segments are shown in Figure 4.

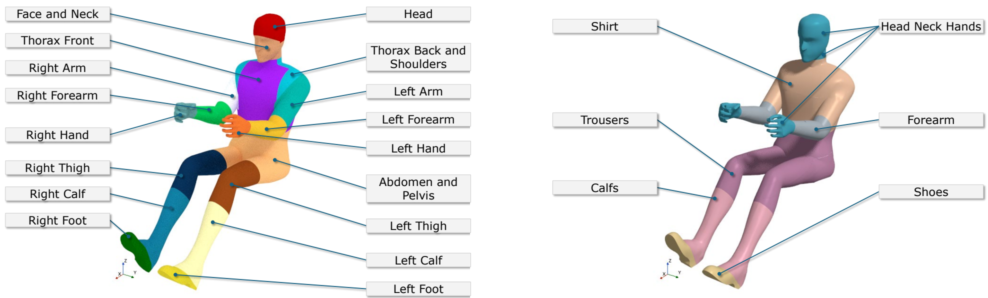

Figure 4: Body and clothing segments for EHT model

For comfort evaluation, the Equivalent Homogeneous Temperature (EHT) metric is applied. The EHT combines air temperature, mean radiant temperature, and air velocity into a single value, providing a comprehensive measure of thermal comfort in non-uniform environments such as vehicle cabins. Conceptually, EHT represents the wall temperature of a uniformly conditioned space under calibrated conditions, assuming negligible air velocity and equal mean radiant and air temperatures. Higher EHT values indicate reduced heat loss, while lower values correspond to increased heat loss. The EHT is computed from the manikin heat flux using the following relationships:

$$T_{eq} = T_S + \frac{x_0}{2x_1} - \frac{\sqrt{x_0^2 + 4x_1\dot{Q}}}{2x_1} \qquad \text{if } \dot{Q} > 0$$

$$T_{eq} = T_S - \frac{x_0}{2x_1} + \frac{\sqrt{x_0^2 + 4x_1\dot{Q}}}{2x_1} \qquad \text{if } \dot{Q} < 0$$

where $\dot{Q}$ is the heat flux on the manikin surface, T_s is the skin temperature, and x_0 and x_1 are calibration constants derived from regression analysis. Although the 3D CFD model provides high accuracy, its computational cost makes it unsuitable for real-time control development. To overcome this limitation, a fast-running surrogate model (FRM) is derived. A Design of Experiments (DoE) study is conducted to systematically vary key parameters such as HVAC settings, ambient conditions, and solar load. The resulting dataset is processed using AVL CAMEO, which generates mathematical response models for comfort indices and thermal states.

In Figure 5 plots of the model qualities are shown. The green point in the graphs represent the validation experiments which were not used for the mathematical model training. The shaded areas indicate the accuracy of the 3D model, ±2K.

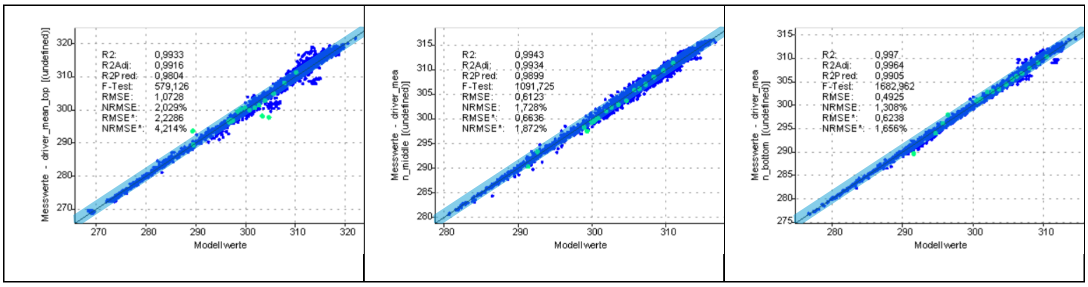

Figure 5: Example results for FRM

These models are packaged as a Functional Mock-up Unit (FMU) and integrated into the plant model, enabling real-time simulation for reinforcement learning (RL) training and hardware-in-the-loop (HiL) applications.

Virtual Driver Model

A virtual driver model is integrated to emulate human interactions with the climate control system. It adjusts parameters such as target cabin temperature, blower speed, and air distribution modes based on comfort feedback. This enables realistic training scenarios for the AI controller and facilitates user-specific adaptation.

2.2 RL Framework

This section gives a brief introduction to the most important concepts of RL and provides the context under which RL can provide solutions for the problem of optimal thermal control of the SmartCorners project.

Introduction to RL

An RL agent is the entity that interacts with an environment to learn how to achieve a goal by maximizing a reward it receives from the interaction. The training environment is the simulated context under which the agent learns how to act in the real world. The schema is shown in Figure 6. The training environment contains states, as well as a set of rules or transition dynamics under which the states transition from one to another. The agent interacts with the environment by

- observing certain states exposed by the environment
- setting actions within the environment, and
- receiving rewards for its actions.

The agent designs a control policy, which is a set of rules that predict the optimal actions an agent should apply given a set of observations. RL training refers to the process of continuous interaction between the agent and the environment during which the agent optimizes the control policy by maximizing the reward it receives after applying an action.

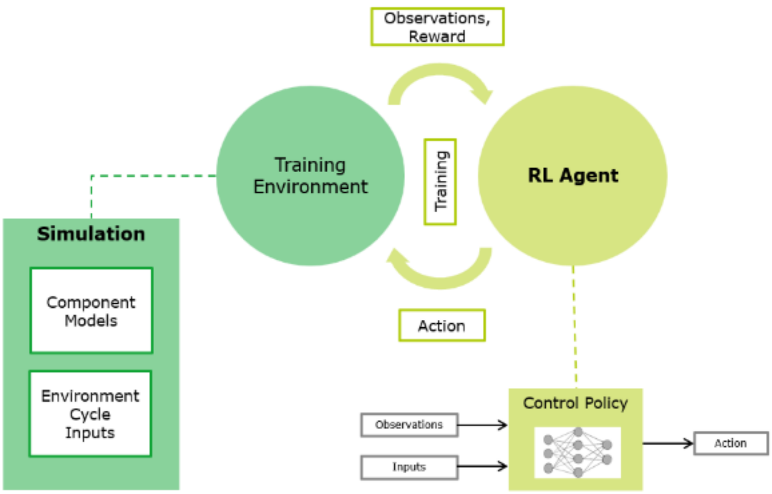

Figure 667: Scheme of the RL training process. The agent trains a control policy to optimally interact with the training environment. It receives an observation and a reward after each action. The optimal policy maximizes the agent's expected reward.

In the context of SmartCorners, the training environment contains all information relevant to the optimization of the thermal controls of the thermal control unit. Specifically, this includes

- an interface to the high-fidelity plant model described in Section 2.1
- a model for all relevant ambient conditions
- a model for driving cycles

The agent interacts in a stepwise and discrete fashion with the environment. That is, at each given point in time, the agent observes some states at a discrete time, determines an appropriate action based on its observations, and applies the action before the environment transitions to the next discrete time step with a new state. During the project we used the interface described in OpenAI's Gym library *Gymnasium*. This interface is simple to use as it only consists of two methods, the initialize method and the step method, which transitions the environment to the next state given an action and returns the current observation and reward.

Ambient Model & Driving Cycles

The training process must contain a model of the ambient environment under which the electric vehicle (EV) is operated. In the RL framework, the ambient model is intrinsically embedded in the training environment. The ambient model must reflect the real-world conditions under which the EV is operated. Moreover, for the sake of robust calibration, the model should contain as many so-called corner cases as possible, i.e. special cases under which either the simulation model or the control policy will respond in extreme ways.

The ambient model contains all necessary information for all model inputs and parameters that are not under direct control of the agent. This information may be given in the form of input distributions, e.g. for ambient weather conditions, or specific driving cycles, such as Worldwide Harmonized Light-duty vehicles Test Cycle (WLTC) for vehicle speed.

In the context of RL, the combination of a single driving cycle with a set of ambient conditions constitutes a so-called episode. During training the agent should be trained on as many representative episodes as possible. For this reason, the training environment was implemented in such a way that random episodes could be generated during training, i.e. the agent can be trained on a theoretically infinite number of realistic episodes.

RL Reward, Targets & Constraints

In RL, the agent will tune its control policy in such a way to maximize the reward it receives from the environment for its actions. In this sense, the reward represents the target of the underlying optimization problem, and it must be carefully designed to respect and trade-off several conflicting goals with each other.

For the SmartCorners project, the reward must therefore reflect all goals of thermal control as well as measures for user comfort based on individual preferences.

For this reason, the RL framework allows the definition of multiple types of control targets, and it allows the definition of multiple targets at the same time. These following types of control targets have been implemented so far:

- **Optimization Target**: allows to minimize/maximize a system output channel.
- **Control Target**: allows to control a system output channel towards a given demand channel. The demand channel must be contained in the ambient data model.
- **Constraints**: allows to force a system output within a specified range of values.

The agents' reward is a combination of all defined target functions. For the SmartCorners project, the following targets have been considered:

- Cabin humidity must be controlled towards a target of 40%
- User comfort must be maximized.

Boundary Conditions / Trainings Setup

The training setup defines the temporal resolution of the simulation, the initialization of each episode, and the physical and operational constraints that ensure safe and meaningful learning.

Each training episode represents a two-hour driving scenario with a total duration of 7200 seconds, with simulation steps of 1 second. The action is updated every 5 seconds.

At the start of every episode, the ambient temperature and humidity are sampled randomly to promote generalization. The ambient parameters continue to vary during the episode to reflect changing driving conditions.

- Ambient temperature: uniformly sampled from 10 °C to 30 °C
- **Relative ambient humidity**: uniformly sampled from 0% to 100%

To maintain physical plausibility and passenger safety, multiple constraint mechanisms are applied:

- **Hard and soft clipping**: Cabin temperature and humidity are subject to both hard limits (enforced at every step) and soft limits that introduce penalties when approached
- **Actuator normalization**: The controllable actuators are scaled to a normalized action range [-1, 1] and then rescaled to their allowed range. This ensures that the actuators stay inside their allowed ranges

The reward function combines two competing objectives:

- Thermal comfort (primary objective)
- Humidity regulation

Additional penalties are introduced when the observations reach a soft limit. The rewards are not scaled.

The training is carried out on GPU hardware when beneficial for algorithm speed, but may fall back to CPU execution if faster for specific algorithms (e.g. PPO). The training time is approximately 5 hours.

3 Results & Conclusion

After training several different RL algorithms, the best candidate is chosen. The chosen agent is based on the AWAC algorithm. This type of algorithm allows training on both offline and online data. Offline data is used to direct the agent in a good starting direction for online training in the real environment.

The final agent uses four environment observations: Ambient temperature, Relative ambient humidity, Cabin temperature, Relative cabin humidity

To demonstrate the agent in action, Figure 7 shows the agent's actions in a driving simulation. The first column of plots shows the observations of the algorithm, the second column shows the actions and the third column shows the resulting reward and comfort.

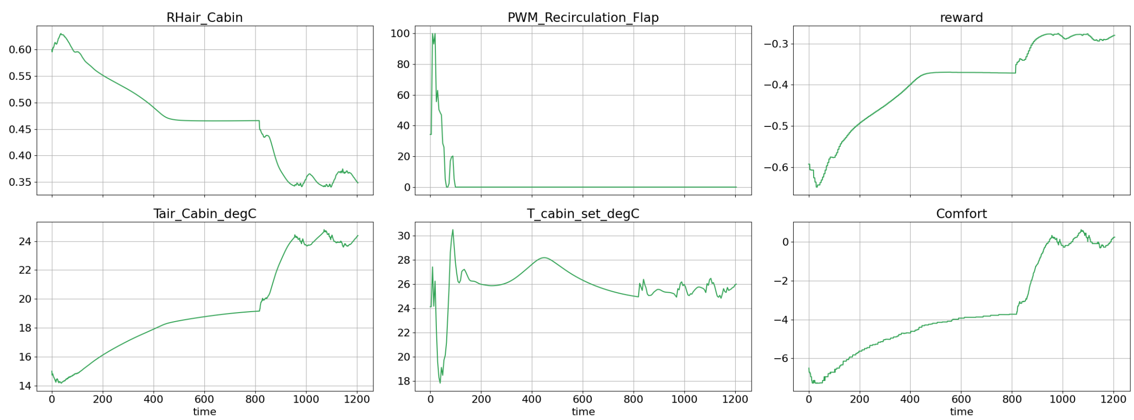

Figure 7: Driving simulation

It can be observed that the trained algorithm selects a cabin set temperature that results in a comfortable thermal sensation at the end of the driving cycle. Moreover, the cabin humidity is close to its target value of 40%. However, greater air recirculation would have been possible in this case.

Nevertheless, this example demonstrates that the training of the algorithm was successful and can be extended to include additional influencing factors such as solar radiation and vehicle speed. The results also indicate that it is particularly challenging for the AI to learn an appropriate set temperature during the heat-up phase. Especially within the first 200 seconds, the set temperature fluctuates significantly over short time intervals. This behavior is likely due to the limited influence of the set temperature during this phase, as the actual cabin temperature is still far from the target value. Such conditions may encourage random set temperature selections during training, leading to these pronounced variations.

4 Outlook

The subsequent steps in the reinforcement learning (RL) training process involve extending both the observation space and the action space of the RL agent. Regarding observations, the model will be augmented to incorporate indicators of user comfort, such as perceived air quality based on CO_2 concentration, and the likelihood of windshield misting due to humidity levels. These parameters must be maintained within predefined thresholds while minimizing electrical energy consumption.

Furthermore, the learning process will be expanded beyond a generic user profile to include adaptation to individual user preferences. This personalization will be achieved by analyzing user interactions and manual adjustments to system settings, which will serve as feedback signals for the learning algorithm.

Following successful demonstration of user-specific learning behavior in a simulation environment, the approach will be validated in a real-world test vehicle. Initially, the baseline algorithm trained on the standard user profile will be deployed. As the vehicle is operated by individual users over time, it is expected that the frequency of manual interventions will decrease, thereby enhancing the overall user experience through a more intuitive and personalized system response.