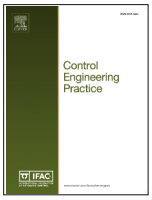
Control Engineering Practice 172 (2026) 106944

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac



Experimental demonstration of time-efficient auto-Calibration of a vehicle thermal management system using safe reinforcement learning

Prasoon Garg [a,*], Emilia Silvas [a,b], Frank Willems [a,b]

[a] Department of Mechanical Engineering, Eindhoven University of Technology, Control Systems Technology, 5600 MB, Eindhoven, The Netherlands
[b] TNO Mobility and Built Environment, 5708 JZ, Helmond, The Netherlands

ARTICLE INFO

Keywords:
Self learning control
Reinforcement learning
Vehicle thermal management
Automotive control

ABSTRACT

Future automotive powertrains become increasingly complex to achieve optimal and robust performance in real-world conditions. This is accompanied by rapidly escalating time and cost demands for calibrating powertrain control systems using existing map- and model-based methods. This challenge highlights the urgent need for self learning control strategies, which autonomously learn optimal control settings on the road. This work demonstrates the potential of Reinforcement Learning-based self learning control for a battery electric vehicle thermal system with safety constraints. To realize Reinforcement Learning on the tested vehicle, a novel exploration method is implemented, which explicitly deals with system safety and minimizes experiment time. By combining an online-learned Gaussian Progress Regression model and a reciprocal Control Barrier Function, the optimal direction and step size for information-rich actions is determined during exploration. Validated on a vehicle test-bench, the proposed method calibrates the reference generator to optimize steady-state operation of the heat pump system. Safe and robust performance is demonstrated for varying ambient temperature and humidity, achieving a relative error in heat pump efficiency within 2% of the true optimum across all validation points. Compared to conventional map-based control, the Reinforcement Learning-based approach reduces calibration time by 69%.

1. Introduction

1.1. Powertrain control is facing a turning point.

Strict regulations on greenhouse gas and real-world pollutant emissions require future automotive powertrains to perform optimally and achieve robust performance in real-world operating conditions. These conditions include a wide range of external disturbances (i.e., ambient and driving conditions) and system uncertainties (i.e., system aging and manufacturing tolerances). To realize optimal and robust performance, numerous hardware and software systems are being added to the existing powertrains, causing an increase in system complexity. Recent developments in automotive powertrains include advanced combustion concepts running on hydrogen and biofuels, increasing levels of electrification, hydrogen fuel cells and waste heat recovery systems (Pereirinha et al., 2018; Sanguesa et al., 2021). As a result, the control calibration effort will become unacceptably large for increasingly complex systems with the application of the traditional map-based control approach (Atkinson, 2014). Moreover, it is challenging to capture a wide range of operating conditions to achieve robust performance with the map-based approach (Willems, 2017). Consequently, there is a need for control methods that combine robust performance and acceptable calibration time and costs, as illustrated in Fig. 1.

1.2. Methods to reduce calibration effort

State-of-the-art model-based control (MBC) methods, such as model predictive control (Karlsson et al., 2010) and model embedded control, reduce calibration effort and improve robustness compared to map-based approaches. By enabling off-line calibration and model-embedded control, these methods decrease the need for vehicle test experiments. However, off-line model-assisted calibration offers limited robustness against real-world disturbances and requires significant expert effort, while model-embedded control faces challenges from high computational demands and modeling complexity, since systems grow more complex.

Further reduction in calibration effort can be achieved by automating the calibration process in the off-line environment, as illustrated in Fig. 1. This approach can significantly reduce the expert effort required in the calibration process. In our previous work (Garg et al., 2023), we achieved a significant reduction of 77% in the expert effort by using Reinforcement Learning to calibrate an automotive thermal system using

* Corresponding author.
E-mail address: prasoon.garg08@gmail.com (P. Garg).

https://doi.org/10.1016/j.conengprac.2026.106944
Received 29 October 2025; Received in revised form 24 February 2026; Accepted 15 March 2026
Available online 24 March 2026

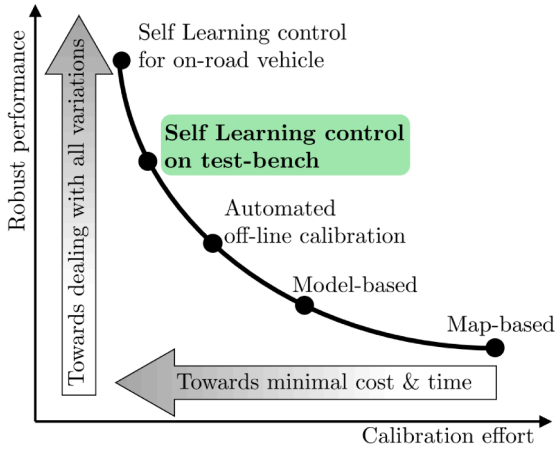


Fig. 1. Illustration of robust performance and calibration effort with different control calibration approaches. The green box highlights the focus of this work. This figure is adapted from Garg et al. (2021).

a high-fidelity simulation model. The method could deal with an added control complexity using deep neural networks, which offers improved robustness to known disturbances. Nonetheless, a significant effort is still required to generate a high-fidelity system model, and limited robustness can only be achieved against operating conditions outside the tested operation envelope.

Self Learning control (SLC) has the potential to further reduce the control calibration time and realize robust performance of future automotive powertrains (Willems, 2017). SLC refers to the control method that autonomously determines optimal control parameter values by interacting with the system on-line (i.e., on a vehicle test-bench or during on-road operation) and by using the available (virtual) sensor information. Fig. 1 illustrates the potential benefits of SLC in reducing calibration effort and enhancing robust performance in real-world conditions. SLC can significantly reduce the control calibration effort in two ways: i) Minimizing modeling effort because the controller can directly learn from the (virtual) sensor information, and ii) Reducing expert effort due to the autonomy of the SLC method. To realize SLC, multiple studies have investigated Extremum-Seeking Control (ESC), see e.g. Ramos et al. (2017), van der Weijst et al. (2019). ESC relies on data-driven and model-free optimization to determine online optimal control settings. Moreover, it can deal with constraints to guarantee safe system operation (van der Weijst et al., 2019). However, the assumption that the inputs to the cost function should possess a quasi-convex mapping limits its generalizability to cost functions with multiple maxima, which are typically found in most powertrain optimization problems.

On-line Reinforcement Learning (RL) offers an effective way to realize SLC on a vehicle test-bench (Garg et al., 2025). These algorithms learn the action-values of different actions (i.e., control settings) from a data sequence without requiring a parametric model of the system (Sutton & Barto, 2018). This data sequence is generated by the agent-environment interactions using exploration and exploitation strategies. Using the information from this data sequence, the agent converges to the optimal actions that maximize the reward function (i.e., minimize the cost function). Having said that, exploring different actions in interaction with the real vehicle is not straightforward due to hardware limits. A growing body of literature has explored RL-based control for engineering systems in the simulation environment, for example, engine control development (Koch et al., 2023), robotics (Kormushev et al., 2013), process control (Nian et al., 2020), automotive powertrain control (Norouzi et al., 2023), vehicle energy management systems (Lei et al., 2025; Qi et al., 2019; Zhang et al., 2025) and autonomous vehicles (Aradi, 2022). However, real-world applications of reinforcement learning in these systems remain limited due to challenges in ensuring safety, minimizing exploration time and requiring significant a-prior system knowledge in form of models. To overcome this challenge, we developed a novel Safe and Information-seeking exploration (Safe-ISE) method in previous work (Garg et al., 2025). This exploration method can realize SLC by ensuring safe system operation while minimizing the number of experiments required during the exploration. Moreover, the proposed method improves robustness in real-world operations by learning directly from the measurement data.

1.3. Research objective and main contributions

The objective of this work is to experimentally demonstrate the potential of the novel Safe-ISE method to drastically reduce calibration effort and improve robust performance in real-world operating conditions.

Compared to previous work (Garg et al., 2025), which introduced the novel exploration method, the main contributions of this work are:

1. Detailed information about the successful implementation of the novel Safe-ISE method on a vehicle;
2. Experimental results demonstrating the functionality of the novel method for calibrating control settings in a vehicle thermal system for different safety bounds;
3. Benchmarking the robust performance and calibration effort of the novel method with a coventional map-based approach under varying ambient temperature and humidity.

2. Heat pump control system

The thermal system of a battery electric vehicle is responsible for ensuring thermal comfort of the passengers in the cabin and temperature control of the powertrain components such as electric machines and batteries. Within the scope of this work, we focus on a heat-pump-based thermal system, which can operate in multiple modes, such as cabin cooling, heating, and dehumidification mode, battery cooling, and heating mode as a function of ambient conditions. Calibrating the corresponding control system across all operating modes of this complex multi-variable system is challenging and requires substantial effort. Determining optimal control settings is key, since the thermal system can consume up to 40% of the total battery energy depending on ambient conditions, which can significantly impact the vehicle range (Lahlou et al., 2020). Therefore, a time-efficient control calibration approach that can optimize the heat pump system's performance in all operation modes is crucial. For the demonstration of the proposed control method, we focus on the heat pump system's cabin cooling mode, which is one of its most used operation modes.

2.1. System description

The hardware schematic of the studied heat pump system in the cabin cooling mode is shown in Fig. 2. The system consists of four actuators (i.e., compressor u_{cmp}, expansion valve u_{exv}, blower u_{blwr} and fan u_{fan}), an accumulator (acc) and two heat exchangers: the outer heat exchanger (ohx) and the evaporator (eva).

The heat pump operation in the cooling mode can be explained from the pressure-enthalpy diagram, also called the Mollier diagram, shown in Fig. 3. In the cooling mode, the heat energy is extracted from the cabin by the heat pump system, and it is released to the environment to meet the demand of cabin air temperature. The air flowing over the outer heat exchanger at the vehicle front extracts the heat from the refrigerant flowing across it. This results in the condensation of the refrigerant, where its phase changes to two-phase and then liquid. If the refrigerant temperature is below its saturation temperature at a given pressure p_3, then the refrigerant is in a subcooled liquid state. The corresponding subcool temperature T_{sc} at a given working fluid pressure is defined as the temperature difference corresponding to $h_{3'}(p_3) - h_3(p_3)$, where h is

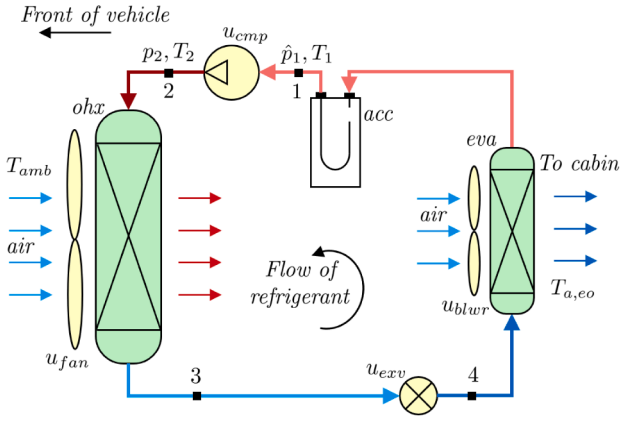


Fig. 2. Schematic of the studied heat pump system in the cabin cooling mode. Symbols T, p represent the measured values for temperature and pressure signals, respectively, for both refrigerant and air. $\hat{p}_1$ represents a virtual pressure sensor. Subscripts *amb* represents the ambient air and a, eo represents the air at evaporator outlet. Components marked in yellow are the actuators, and the heat exchangers are marked in green. Blue arrows represent colder temperatures of fluids (i.e., air, refrigerant), while red arrows represent warmer temperatures.

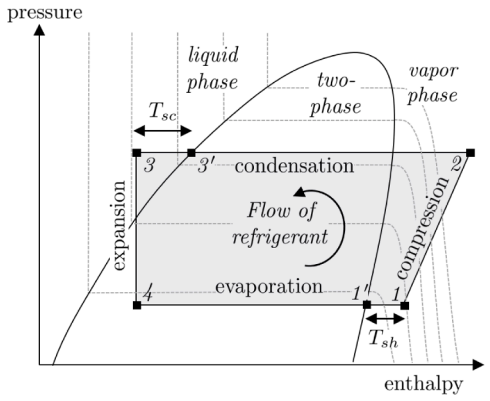


Fig. 3. Mollier diagram for the studied heat pump system in cabin cooling mode. Used refrigerant is R-1234yf. Grey lines represent isothermal conditions.

enthalpy, $h_{3'}(p_3)$ is the enthalpy of the saturated working fluid in liquid state.

At the expansion valve, a pressure drop occurs as the refrigerant changes to a two-phase mixture from the liquid phase. At the evaporator, the two-phase refrigerant extracts the heat energy from the hot air flowing over it, and the refrigerant changes to the vapor phase. The working fluid is in superheated vapor state if its temperature is above the saturation temperature at given pressure p_1. The superheat temperature T_{sh} at a given working fluid pressure is defined as the temperature difference corresponding to $h_1(p_1) - h_{1'}(p_1)$, where h is enthalpy, $h_{1'}(p_1)$ is the enthalpy of the saturated working fluid in a vapor state. It is beneficial to keep the refrigerant in a saturated vapor state at the outlet of the evaporator, as the presence of liquid droplets in the refrigerant going into the compressor can cause damage to its components. To ensure the safety of the compressor, an accumulator is introduced before the compressor, which extracts any remaining liquid droplets in the refrigerant. The refrigerant in a vapor state at low temperature and pressure is then compressed to vapor at high temperature and pressure by the compressor, closing the cycle. The refrigerant temperature at the compressor outlet is constrained as high temperature can cause damage to the compressor due to high frictional forces. Also, the refrigerant pressure after the compressor is constrained by the operational guidelines of the heat pump system.

2.2. Control problem

The high-level control objective of the heat pump-based thermal system in the cabin cooling mode is to ensure the thermal comfort of passengers by tracking the desired air temperature at the evaporator outlet, i.e., $T_{a,eo}$ while maximizing the heat pump efficiency and satisfying the safety constraints. The heat pump efficiency is defined by the Coefficient of Performance (COP), which is the ratio of useful thermal power exchanged at the evaporator and the electrical power consumed by the compressor and fan. It is defined as,

$$\text{COP} = \frac{P_{eva}(T_{sc}, u_{cmp})}{P_{cmp}(u_{cmp}) + P_{fan}(u_{fan})} \tag{1}$$

where P_{fan}, P_{cmp} are the required electrical power by the fan and compressor, respectively. P_{eva} is the cooling power delivered at the evaporator expressed as,

$$P_{eva} = \dot{m}_r(h_1 - h_4) \tag{2}$$

where $\dot{m}_r$ is the refrigerant mass flow rate across the evaporator, and h_1 and h_4 are the enthalpy of the working fluid at the inlet and outlet of the evaporator, respectively. Larger values of T_{sc} increase condensation pressure and decrease enthalpy at the evaporator inlet. As a result, the cooling performance of the evaporator increases with an increase in the specific compression work. Due to these opposing effects with an increase in T_{sc}, a trade-off exists, leading to a maximum COP (Yamanaka et al., 1997). Changing u_{cmp} can modify $\dot{m}_r$ and p_2, thereby affecting P_{eva}. To maximize COP and ensure thermal comfort, the optimal value of T_{sc} is determined by solving the following optimal control problem,

$$\min_{T_{sc}} \quad -\text{COP}(x, u, w, d), \tag{3a}$$

$$\text{s.t.} \quad \dot{x} = f(x, u, w, d, t), \tag{3b}$$

$$y = g(x, u, w, d, t), \tag{3c}$$

$$|r_{T_{a,eo}} - T_{a,eo}| \le \epsilon, \tag{3d}$$

$$\overline{p}_2 - p_2 \ge 0, \tag{3e}$$

$$\overline{T}_2 - T_2 \ge 0, \tag{3f}$$

$$T_{sh} \ge 0, \tag{3g}$$

$$T_{sc} \ge \underline{T}_{sc}. \tag{3h}$$

where f is a $n-$dimensional system function vector, g is a $p-$dimensional output function vector, x is a vector of system states, $u = \begin{bmatrix} u_{exv} & u_{cmp} \end{bmatrix}^\top$ is a vector of control actions, $w = \begin{bmatrix} T_{amb} & RH_{air} \end{bmatrix}^\top$ is the vector of external inputs consisting of known disturbances, $y = \begin{bmatrix} T_{a,eo} & T_3 & p_3 \end{bmatrix}^\top$ are the measured outputs, d represents unknown external disturbances such as a change in blower speed, number of passengers. ϵ is allowable mismatch in desired $T_{a,eo}$, T_{amb} is the ambient air temperature and RH_{air} is the relative humidity of the ambient air. The subcool temperature T_{sc} is required to satisfy $T_{sc} \ge \underline{T}_{sc}$ to prevent frosting of the evaporator as it increases the thermal resistance between refrigerant and air, which reduces the heat transfer across the evaporator (Hermes et al., 2021). The safety constraints include mechanical constraints (Eqs. (3e)–(3g)) and preventation of evaporator frosting (Eq. (3g)). Thermal comfort constraint is represented by (3d).

2.3. Benchmark control approach

Fig. 4 shows the schematic of the benchmark control approach. It is a classical feedback control system with two single-input-single-output (SISO) PID-based feedback controllers $C_{fb,1}, C_{fb,2}$. These two controllers track the setpoints from the reference generators $\mathcal{R}_1, \mathcal{R}_2$, which are 2-D look-up maps as a function of ambient temperature T_{amb} and relative air humidity RH_{air}. $\mathcal{R}_1$ outputs the setpoint $r_{T_{a,eo}}$ whereas $\mathcal{R}_2$ outputs the setpoint $r_{T_{sc}}$. u_{exv} is manipulated to modify the condensation pressure and track the reference subcool temperature $r_{T_{sc}}$. The desired cooling load defined by $r_{T_{a,eo}}$ is met by modulating the heat transfer capacity of

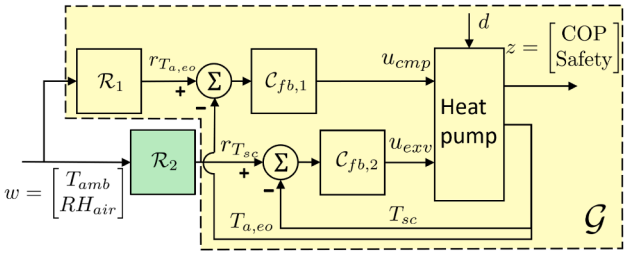


Fig. 4. Schematic of the benchmark heat pump control system.

the evaporator using u_{cmp}, which adjusts $\dot{m}_r$ and p_2. The effect of u_{cmp} on $\dot{m}_r$ and p_2 causes a strong coupling between these two SISO control loops, which makes this a challenging control problem.

2.4. Vehicle test-bench specifications

The vehicle test bench is equipped with a chassis dynamometer inside a climatic chamber where ambient air temperature and ambient air relative humidity can be controlled at stationary conditions. The vehicle can be driven at varying speeds, which can simulate the effects of air-flow across the outer heat exchanger. Moreover, varying windspeed can be realized in the climatic chamber. The cooling power demand can be varied by manipulating the desired cabin air temperature and the blower speed from within the driver's cabin.

The vehicle is equipped with the ETAS rapid prototype system (RPS) module shown in Fig. 5 for control implementation and validation. The ETAS RPS module consists of the benchmark (see Section 2.3) and the RL-based softwares for the heat pump control. The sensor signals are logged using the IPETRONIK data logger. The base software for heating, ventilation and air-conditioning (HVAC) control is implemented on the *Base HVAC ECU*, which communicates with the heat pump control via the LIN communication protocol. The *Powertrain ECU* contains the software for the powertrain control, which uses ethernet and Vehicle CAN communication protocols.

The RL-based control method is developed in the 2017b version of MATLAB and Simulink on a laptop. This control software is then compiled using the ETAS Intecrio to generate the software for RPS implementation. The compiled software is uploaded on the RPS module using the ETAS INCA software, which is also used for pre-calibration, measurement data analysis and fine-tuning on the vehicle test-bench. The compiled software is then run on the ETAS RPS module at an update frequency of 1Hz by bypassing the benchmark control software.

2.5. Benchmark control calibration process

The process of determining the controller parameters to realize the desired system performance is defined as control calibration. The heat pump control system is a non-linear system for which the calibration process is a complex task. This process requires large development times and costs. The benchmark control development process follows a typical V-development cycle (Liu et al., 2016), as shown in Fig. 6. The initial step is the *concept definition* where the controller inputs and outputs, controller type and the control system design are chosen. In the benchmark control system, the controller is map-based and is calibrated by an expert (i.e., a system engineer).

As a first step, high-fidelity system models are generated that can capture multiple known disturbances to the heat pump system, such as varying ambient conditions and vehicle speed. Then, the map-based control blocks $\mathcal{R}_1, \mathcal{R}_2, \mathcal{C}_{fb,1}, \mathcal{C}_{fb,2}$ are calibrated in an off-line (i.e., desktop) environment using the system model. This step is called control strategy in model-in-the-loop (MIL) (Step d) in Fig. 6. The corresponding tasks are performed by a system engineer, who possesses in-depth system knowledge. These belong to the most time-consuming calibration tasks in the V-development cycle. In the benchmark approach, $\mathcal{R}_1, \mathcal{R}_2$ maps are calibrated as a function of known disturbances to optimize the steady-state operation of the heat pump system. For the studied benchmark, the controller is calibrated for two known external disturbances: T_{amb} and RH_{air}. Thereafter, the gains of the PID-based controllers $\mathcal{C}_{fb,1}, \mathcal{C}_{fb,2}$ are tuned off-line by the system engineer.

To prepare the control software for implementation on the vehicle electronic control unit (ECU), the software is first validated in the hardware-in-the-loop (HiL) (Step e) environment. Thereafter, maps $\mathcal{R}_1, \mathcal{R}_2$ and PID-based controllers $\mathcal{C}_{fb,1}, \mathcal{C}_{fb,2}$ are fine-tuned on RPS (Step f). In case of deviation in the control system performance on the test-bench from the simulation results, the simulation models are fine-tuned using the new experimental data. This process is repeated until the desired system performance is achieved on the system test-bench. Although this control development approach is commonly used in the automotive industry, it has a few limitations:

1. It requires system engineers/experts with in-depth system knowledge for control calibration, for example, the impact of reference setpoints r on performance outputs z;
2. It requires a large effort (i.e., experiment times, expert effort) in both generating accurate system models and control calibration;
3. The control performance is sub-optimal due to challenges in optimizing complex non-linear control problems using a map-based approach and introducing safety margins.

3. Reinforcement learning-based control method

We begin this section by presenting how we formulate the heat pump control problem in a Reinforcement Learning framework. Then, we briefly describe how we apply the Safe and Information-seeking exploration (Safe-ISE) method introduced in Garg et al. (2025), on the heat pump control problem.

3.1. Contextual bandits problem

To optimize the steady-state operation of the heat pump, the control problem stated in Eq. (3) is formulated as a Contextual Bandits RL problem. Fig. 7 shows the agent-environment interaction in the proposed RL framework. The RL agent determines the optimal value of $r_{T_{sc}}$, while the control components designated by $\mathcal{G}$ are treated as a part of its environment (see Fig. 4). In the Contextual Bandits problem, the agent's action a_i in a context s_i is independent of past actions and contexts, nor does it affect future actions and contexts, where $i = 1, 2, 3, \ldots$ represents the instances of the agent-environment interactions. This characteristic is consistent with the optimization of the heat pump performance for steady-state operation. The objective of a Contextual Bandits RL problem is to determine the optimal actions a_i^* for a given context s_i such that the average value of the reward R i.e., action-value $q(s_i, a_i)$ is maximized. The key elements in formulating the Contextual RL problem are reward R, action-values $q(s, a)$, context s, action a and safety constraints, which are defined for the heat pump control problem as follows,

1. *Reward R*: To maximize COP and meet the desired $T_{a,eo}$, R is defined as a weighted function of these two terms. It is defined as,
$$R = c_1 \cdot \text{COP} - c_2 \cdot |r_{T_{a,eo}} - T_{a,eo}| \tag{4}$$
here, c_1, c_2 are scaling weights. The thermal comfort constraint defined earlier in Eq. (3d) is treated as a soft constraint with this formulation.
2. *Action-value* $q(s_i, a_i)$: It is determined by taking an average of the reward signal over a specified time duration of λ_2 seconds if the system output y is within the desired tolerance of $\pm\xi$ of the agent's action for the duration of λ_1 seconds as illustrated in Fig. 8. λ_2 serves as an additional parameter to ensure whether the reward signal has achieved steady-state. The main reason behind using λ_2 is to allow the flexibility on number of reward samples to average to determine

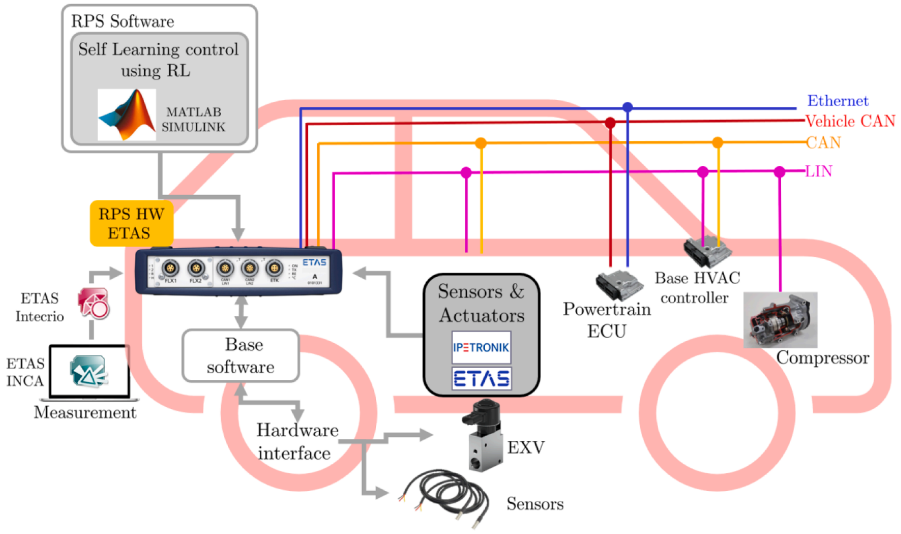


Fig. 5. Illustration of the vehicle RPS specifications used for the validation. HW is hardware. Ethernet, CAN and LIN are the different communication protocols used in the RPS.

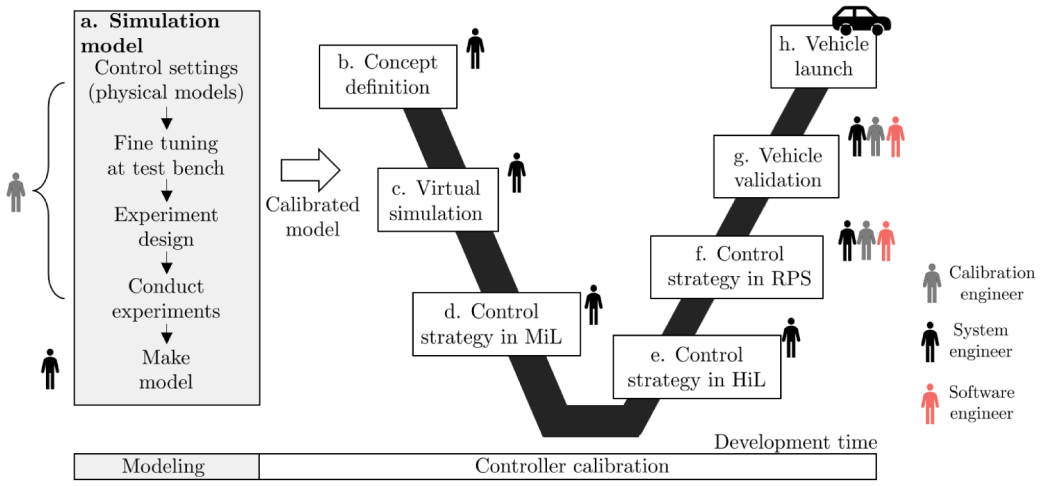


Fig. 6. V-development cycle used in benchmark approach for control development of the heat pump system. MiL is model-in-the-loop, HiL is hardware-in-the-loop and RPS is rapid prototyping system.

the action-value. For example, if the reward signal has larger oscillations before it converges, a small value of λ_1 will only include recent data samples, which increases the likelihood of better capturing the average reward value. In practice, λ_2 can be chosen equivalent to λ_1. $q(s_i, a_i)$ for taking a_i in context s_i is calculated as,

$$q(s_i, a_i) = \left. \frac{\sum_{t}^{t+\lambda_2} R_t \cdot \Delta t_s}{\sum_{t=1}^{t=\lambda_2} \Delta t_s} \right|_{(s,a)} \tag{5}$$

where Δt_s is the sampling time and t represents discrete time instant.

3. *Context* s: The stationary operating point of the heat pump system can be defined by stationary conditions of known external disturbances w (see Fig. 4), such as ambient air temperature T_{amb}, relative humidity of the air RH_{air}, cooling demand of the passenger and vehicle speed. In this work, s is defined as a subset of w i.e., T_{amb} and RH_{air} expressed as,

$$s = \begin{bmatrix} T_{amb} & RH_{air} \end{bmatrix}^\top \tag{6}$$

We assume a discrete context space based on general industry practice for optimizing steady-state operation.

4. *Action* a: The agent's action a is the T_{sc} setpoint i.e., $r_{T_{sc}}$, which is directly correlated with the reward (Eq. (4)). $r_{T_{sc}}$ can take real values, therefore, we consider a continuous action space, i.e., $\mathcal{A} \in \mathbb{R}^+$. Moreover, the action space is constrained by the lower bound on T_{sc} (Eq. (3h)). Also, it is assumed that a safe initial action a_0 for all context values is known from the historical data of the existing heat pump system. The optimal action of the agent a^*_{i+1} in a context s_i is defined as the action with the highest action-value and is expressed as,

$$a^*_{i+1}(s) = \arg\max_a q(s_i, a_i) \tag{7}$$

5. *Safety constraints*: Out of the three safety constraints defined in Eq. (3), this work focuses on one constraint, i.e., $\overline{p}_2 - p_2 \geq 0$ for demonstration purpose. The other two constraints on system states T_2 and T_{sh} are dealt with using the benchmark safety algorithm.

3.2. Exploration method

In this work, we use the Safe and Information-seeking exploration (Safe-ISE) method introduced in our previous work (Garg et al., 2025). The high-level schematic of the method is shown in Fig. 7. To determine the optimal action a^*_i in a given context s_i, the agent explores different actions $a_i \in \mathcal{A}$ for each agent-environment interaction instance $i = 1, 2, 3, \ldots$ and evaluates the corresponding action-value $q(s_i, a_i)$ received from the environment. For exploration, an off-policy learning

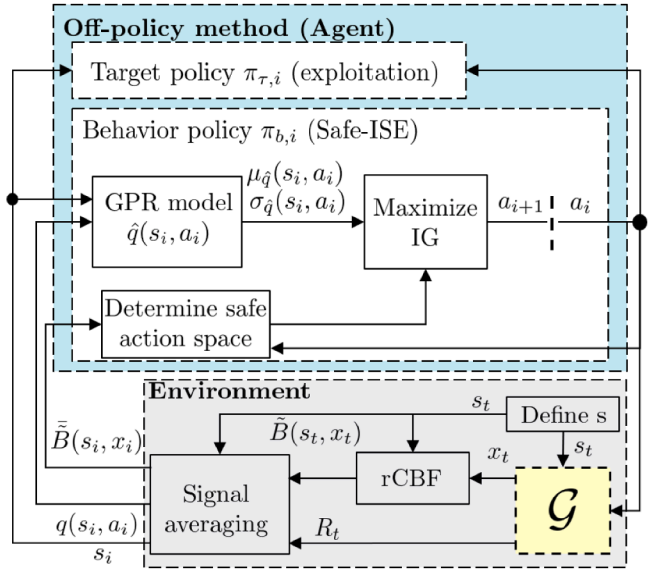


Fig. 7. Illustration of the exploration and exploitation using off-policy learning in RL-based control method. Blocks highlighted in blue are focus of this work. $t = 1, 2, 3, \ldots$ are the discrete time instances and $i = 1, 2, 3, \ldots$ represents the instances of the agent-environment interaction.

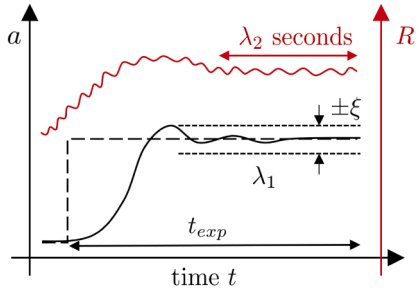


Fig. 8. Approach to determine the action-value $q(s, a)$. The solid black line shows the system output y, and the dotted black line shows the action a taken by the agent.

approach is applied, which uses two policies: 1) Behavior policy and 2) Target policy. The behavior policy π_b, also called logging policy, is used to explore different actions and gather the information required to determine a_i^*. After the exploration process is completed, the behavior policy converges to an optimal deterministic policy called the target policy $\pi_{\tau,i}$.

To estimate the action-values $\hat{q}(s_i, a_i)$ on-line, we use a Gaussian Process Regression (GPR) model, which enables incremental learning from data samples. The GPR model outputs mean predictions $\mu_{\hat{q}}(s_i, a)$ and variance $\sigma_{\hat{q}}(s, a)$ in action-values for all actions in the action-space. GPR demonstrates satisfactory performance given the relatively small number of dimensions and data points in our setting. However, GPR suffers from the computational complexity that scales cubically with the number of data points i.e., $\mathcal{O}(n^3)$. To address scalability issues in very high dimensional state space or larger datasets, sparse Gaussian Process Regression techniques can be employed (Quinonero-Candela & Rasmussen, 2005).

To provide safety information to the agent during exploration, the reciprocal Control Barrier Function (rCBF) is used in this work. It is a modified form of Control Barrier Functions studied extensively in the field of Control Theory for formal proofs of safety in dynamical systems (Marvi & Kiumarsi, 2021; Prajna & Jadbabaie, 2004; Wieland & Allgöwer, 2007). Its value is used to limit the step size of the agent's action if it approaches the boundary of the safe set of system states.

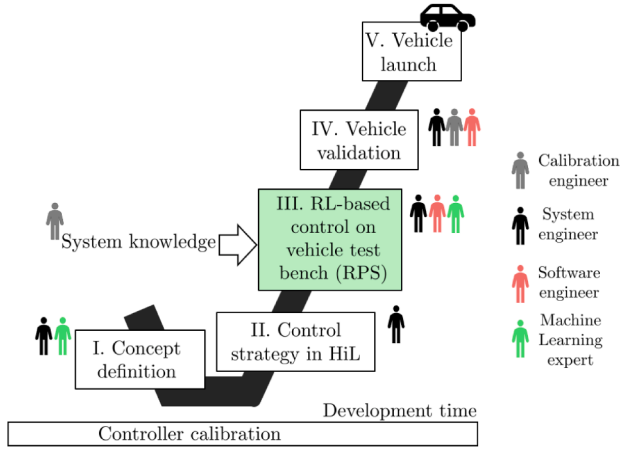


Fig. 9. Illustration of the V-development cycle with the proposed calibration process using RL-based control method for the heat pump system.

For a time-efficient exploration, we use the concept of information gain (IG), which consists of two different metrics: 1) Uncertainty in the action-values of the unexplored actions and 2) Proximity of other local or global maximas with respect to the current maxima in action-values called proximity to maxima (PM). PM is an adapted version of the expected improvement acquisition function used in the Bayesian Optimization (Pelikan et al., 1999), which determines the existence of other maximas in the action space with respect to the current maxima.

3.3. Calibration process for proposed RL-based controller

Fig. 9 shows the adapted V-development cycle for the RL-based Self Learning control method. It illustrates its potential to reduce the calibration effort. The controller parameters are calibrated directly on the vehicle test-bench in the RPS environment without using any prior system model. This eliminates the requirement for accurate system models used in the MiL calibration in the benchmark process (Fig. 6). The control development process begins with the concept definition (Step I), where the system engineer and the Machine Learning expert define the RL-based control system design, controller inputs and outputs. The first step in the calibration process is the control strategy in HiL (step II), where the software is prepared for real-time implementation. The next step in the calibration the application of the RL-based control method on the vehicle test-bench using the RPS (Step III). The input to this calibration step consists of the system knowledge from the calibration engineer for defining the system constraints (Eqs. (3e)–(3h)) and the context space (see Section 3.1). The RL-based method can then autonomously determine the optimal controller settings for the steady-state control of the heat pump system.

With the proposed process, a significant reduction in the expert calibration effort is made by eliminating the steps c-e in the benchmark process (Fig. 6). Moreover, the expert effort required to fine-tune the controller parameters on the RPS in the benchmark process (Step f in Fig. 6) is replaced by the effort from the ML expert for fine-tuning the parameters of the SLC method (Step II in Fig. 9). This transition from model-based control to Self Learning control reduces the amount of system knowledge needed by the calibration engineer for the calibration process.

4. Experimental results

In this section, we present the results of validating the RL-based control method on the vehicle RPS for varying ambient temperature and humidity and for different safety constraints. For all the validation cases, we analyze optimality, safety and calibration effort for the proposed control method and the benchmark controller.

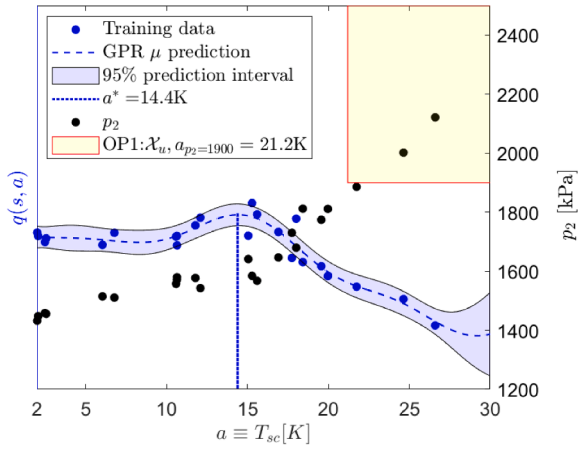


Fig. 10. Results of the steady-state sweep to find true optima a^* and critical action $a_{p_2=\bar{p}_2}$. $\mathcal{X}_u$ represents unsafe actions and corresponding p_2 values for OP1.

4.1. Selected operating conditions

The proposed control method is validated at three different steady-state operating points (OPs) listed in Table 1. For all OPs, a safe initial action $a_0 = 15$K is assumed to be known a-priori from the system knowledge. To study varying ambient conditions i.e., $s \doteq w = [T_{amb}\ RH_{air}]^\top$, the vehicle is mounted in the climatic chamber. OP1 and OP2 are nominal operating points with similar values for ambient conditions. To evaluate the method's flexibility to a change in functional requirement, the safety-related constraint in OP2 is changed from $\bar{p}_2 = 1900$kPa to $\bar{p}_2 = 2100$kPa. The true optimum a^* is similar for OP1 and OP2 due to similar ambient conditions. However, different critical actions $a_{p_2=\bar{p}_2}$ exist for different values of $\bar{p}_2$. To demonstrate the robustness of the control method for varying operating conditions, the method is validated at OP3, where both T_{amb} and RH_{air} are set to values larger than OP1 and OP2. In OP3, the compressor speed was set to a higher value of 2500 rpm to meet the increased cooling load caused due to larger T_{amb} and RH_{air} values.

To find the true a^* and $a_{p_2=\bar{p}_2}$ for comparison, a steady-state T_{sc} sweep is made. Herein, T_{sc} is varied in the range $[3, 30]$K, and the measurement data on $q(s,a)$ (Eq. (5)) and p_2 is collected. The steady-state data points $q(s,a)$ are fitted using the Gaussian Process Regression model. For the fitting, the MATLAB GPR toolbox is used.[1] Fig. 10 shows the sweep results for OP1. $a_{p_2=\bar{p}_2}$ is found by interpolation of the measured data for p_2. a^* for each OP is determined as follows,

$$a^*(s) = \underset{a \doteq T_{sc} \in \mathcal{A}_{eval}}{\arg\max}\ \mu_{\hat{q}}(s,a) \tag{8}$$

where $\mu_{\hat{q}}(s,a)$ is the mean prediction of the GPR model, $\mathcal{A}_{eval} = \{3, 3.1, 3.2, \ldots, 30\}$K represents the set of T_{sc} values for which the GPR model is evaluated.

4.2. Hyperparameter tuning

The hyperparameter values for the RL-based control method are listed in Table 2. The parameters chosen in the simulation study in Garg et al. (2025) are further adapted for the test-bench. For initial exploration around a_0, $\Delta a_0 = 3$K instead of 0.5K is chosen due to the feedback controller's limited sensitivity to minor changes in the reference setpoint. To determine mean values of reward and rCBF, the actual T_{sc} is averaged over a time-window of $\lambda_2 = 60$ s if T_{sc} stays within the tolerance ξ of the agent's action for a time-window of $\lambda_1 = 120$ s. A large

[1] MATLAB function used: fitrgp.m with following specifications, fitrgp(X,Y, 'FitMethod', 'exact', 'KernelFunction', 'squaredexponential', 'KernelParameters', $[\sigma_{l,0}\ \sigma_{f,0}]$,'Sigma',$\sigma_{n,0}$, 'ConstantSigma', true)

Table 1
Specifications of the validation operating points (OPs).

OP	Description	T_{amb} [K]	RH_{air} [%]	Constraint Bound $\bar{p}_2$ [kPa]	u_{cmp} [rpm]	u_{blwr} [%]	u_{fan} [%]	Safe initial action a_0 [K]	True optimal action a^* [K]	Critical action $a_{p_2=\bar{p}_2}$ [K]
OP1	Nominal	308.15	10	1900	2000	54	50	15	14.4	21.2
OP2	Nominal with different constraint bound	308.15	10	2100	2000	54	50	15	14.4	25.3
OP3	Non-nominal with different ambient conditions and constraint bound	313.15	30	2300	2500	54	50	15	16	24

Table 2
List of hyperparameters for RL-based control method method.

Parameter	Description	Category	Value
c_1	Scaling weight for efficiency	Reward function	1
c_2	Scaling weight for thermal comfort		0
Δa_0	Step-size in action for start-up	Initialization	3 K
λ_1	Window length to detect stationary condition	Averaging reward	120 s
λ_2	Window length for averaging R		60 s
ξ	Allowable deviation from setpoint		1 K
p	Number of iterations to check for optimality	Stopping criteria	3
ϵ	Small constant for change in optimal action		0.5K
$\overline{\tilde{B}}$	Maximum allowable value of rCBF $\tilde{B}$		190
$\overline{N}_{int}$	Maximum number of experiments		50

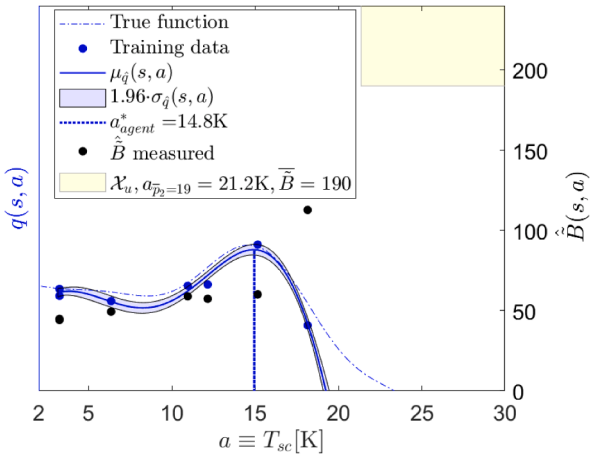


Fig. 11. Comparison of exploration results for OP1 with respect to the true action-value function.

value of λ_1 is chosen to account for the large time constant in the heat pump system. $\xi = 1$K is chosen to account for feedback controller's limited sensitivity and minimize the time required to complete one iteration of agent-environment interaction i.e., t_{exp} (see Fig. 8) at the cost of optimality. Reducing ξ can result in longer t_{exp} before a value of $q(s,a)$ is sampled. The hyperparameters for the GPR model are similar to the values used in our previous work (Garg et al., 2025).

The hyperparameters for the GPR model are not optimized during the learning on the vehicle test-bench and they are kept constant due to challenges in implementing the MATLAB GPR toolbox on the RPS. Therefore, the GPR model with constant hyperparameter values is implemented on the RPS. The hyperparameter values shown in Table 2 are the values obtained by fitting the stationary data points collected from the steady-state sweep using the MATLAB GPR toolbox as discussed in Section 4.1. A squared-exponential kernel function is used due to its suitability for handling smooth functions.

4.3. Validation of optimal and robust performance

4.3.1. Nominal optimal performance:

Fig. 11 shows the result of applying the RL-based control method at OP1. Results show that the agent explores 7 different actions before converging to the action $a^*_{agent} = 14.8$ K, which is close to the true optimal $a^* = 14.4$ K. At all times during the exploration, the agent maintains the safe operation of the heat pump system by not exploring unsafe actions.

The exploration process is further studied by analyzing the incremental learning process shown in Fig. 12. The agent begins the exploration process by taking the safe initial $a_0 = 15$ K. Thereafter, it explores the action around the safe action, i.e., $a = 15 \pm 3$ K to determine the actions that approach the safe boundary. For $a = 18$ K, an increase in $\tilde{B}$ is observed compared with $a_0 = 15$ K. Therefore, the agent updates its estimate of the current safe action space as $\mathcal{A}_{i=3} = [3, 18]$ K. $\mu_{\hat{q}}(s,a)$ of the GPR model at $i = 3$ implies that actions outside [12, 18] K have a smaller $q(s,a)$. Based on this information, the agent's current optimal action is $a^*_{agent,i=3} = 14.5$ K and assumes that there exist no other maxima in $q(s,a)$.

After the initial exploration, the agent determines the action that ensures safety and maximizes the information gain metrics. For $i = 4$, it takes the action $a = 10.9$ K in the vicinity of [12, 18] K that has the maximum $\sigma_{\hat{q}}(s,a)$ from the range of actions that maximize information gain. Here again, the GPR model is updated with the sampled $\{a, q(s,a)\}$ values. Based on the updated $\mu_{\hat{q}}(s,a)$, $a^*_{agent,i=4} = 3$ K, however, there exists large uncertainty in its $q(s,a)$ at $a = 3$ K. Therefore, the agent chooses $a = 3$ K as its next action for $i = 5$. After exploring $a = 3$ K, the agent discovers that the optimal action lies close to its earlier estimate, i.e, $a^*_{agent,i=5} = 14.4$ K. Hereafter, the agent continues to explore the action space until it meets the stopping criteria. The agent converges to $a^*_{agent} = 14.8$ K after 7 iterations, which is within 2% relative error from the true action-values $q^*(s,a)$ (Table 1), and then it ends the exploration.

4.3.2. performance for different safety constraint bounds:

Fig. 13 shows the result of the agent's exploration for OP2. In OP2, $\overline{p}_2$ is relaxed from 1900 to 2100 kPa. This resulted in a different critical action, i.e., $a_{p_2=2100\text{kPa}} = 25.3$ K. On the other hand, this also widens the action-space available for exploration. After the initial exploration similar to OP1, the agent explores an additional action $a = 18.7$ K for iteration $i = 4$ approaching the safe boundary as compared to OP1.

However, based on $\mu_{\hat{q}}(s,a)$, there is an absence of maxima for $a \geq 18.7$ K; therefore, the agent does not explore actions in this direction in the next iterations. The exploration stopped after the agent met the stopping criteria (see Garg et al., 2025 for details on stopping criteria). The agent converges to $a^*_{agent} = 15.3$ K, which is within 2% relative error from the true action-values $q^*(s,a)$ after 8 iterations (See Table 1).

4.3.3. Performance for changing ambient temperature and humidity:

Fig. 14 shows the results of the learning process by the agent in OP3, where both T_{amb} and RH_{air} are set to higher values compared to OP1 and OP2 implying a larger cooling load on the heat pump system. Here again, the agent stops exploration as it meets the stopping criteria SC1. It is seen that after 7 iterations, the agent converges to $a^*_{agent} = 15.15$K and $q(s,a)$ with a 2% relative error from the true action-values $q^*(s,a)$. This result demonstrates that despite the change in the ambient conditions, the agent can converge to close-to-optimum while requiring a similar number of experiments compared to OP1 and OP2.

Table 3 shows the comparison results between the RL-based control method and the benchmark map-based controller. The three performance metrics are safety and deviations with respect to the true a^* and action-values $q^*(s,a)$ for the validation OPs. For all OPs, the agent satisfies the safety constraints during exploration. Moreover, it is seen that the RL-based control method outperforms the benchmark controller in convergence to true a^*, and converges to within $\pm 2\%$ of the true optimum q^*.

4.3.4. Impact on calibration effort:

Following the method described in Garg (2024), the calibration effort is calculated considering the entire development cycle from concept definition towards vehicle validation. The calibration process with the proposed RL-based control method results in a significant 69% reduction in total calibration time compared to the benchmark process. This significant reduction is achieved by minimizing the experiment and simulation times while requiring minimal system knowledge.

5. Conclusions and future work

In this work, we present the successful experimental demonstration of a Reinforcement Learning-based Self Learning control method that can automatically optimize control settings on the vehicle test bench

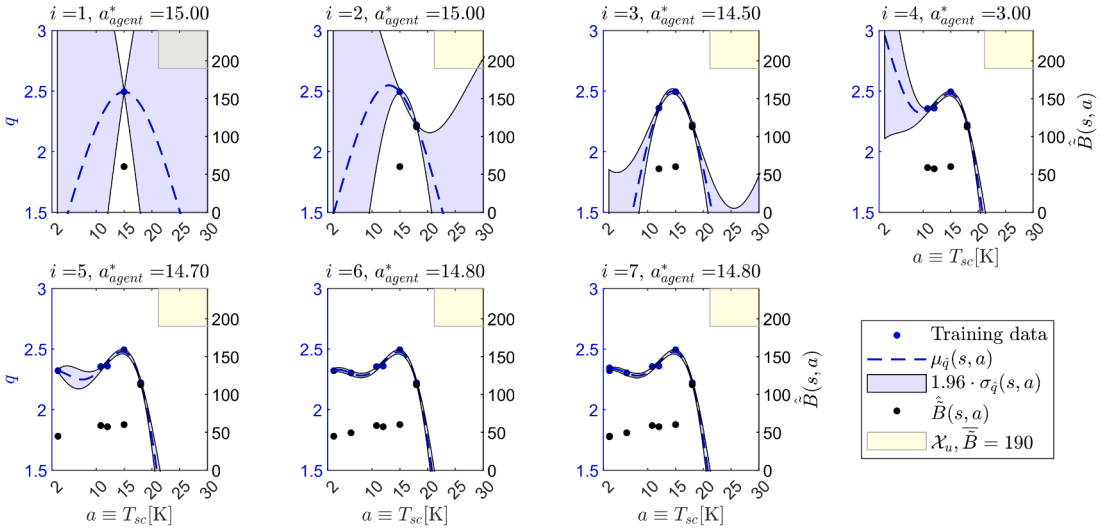


Fig. 12. Incremental learning of the agent in OP1.

Table 3
Comparison of the map-based benchmark and RL-based control method.

OP	Safe operation ($\tilde{B} \leq \overline{\tilde{B}}$)		Deviation from a^* $\frac{a^*_{agent}-a^*}{a^*} \times 100$[%]		Deviation from q^* $\frac{q^*_{agent}-q^*}{q^*} \times 100$[%]	
	Benchmark	RL	Benchmark	RL	Benchmark	RL
OP1	Yes	Yes	−10.42%	**−2.78** %	−2%	**0.4** %
OP2	Yes	Yes	−10.42%	**−6.25** %	−2%	**−2** %
OP3	Yes	Yes	4.37%	**−1.87** %	-0.44%	**−1.3**%

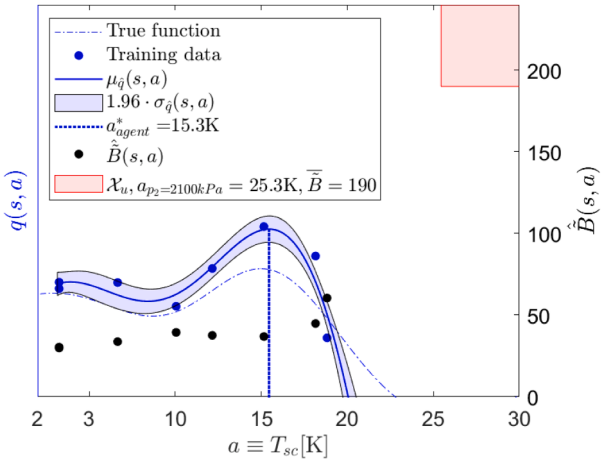


Fig. 13. Comparison of exploration results for OP2 with respect to the true action-value function.

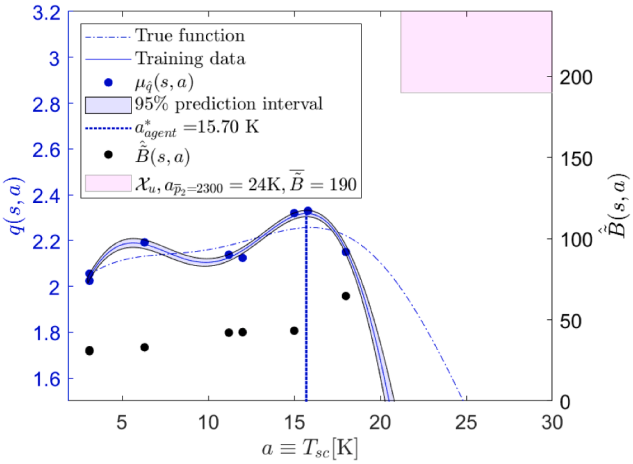


Fig. 14. Comparison of exploration results for OP3 with respect to the true action-value function.

without a prior system model. The proposed control method uses a Safe and Information-seeking exploration method to maintain safe operation and minimizes the number of experiments during exploration. The method is applied for calibrating the reference generator to optimize the steady-state operation of the vehicle heat pump system in the cabin cooling mode. The proposed control method outperforms the benchmark map-based control and achieves convergence with an accuracy of ±2% relative error from the true optimum in the action-values across all the validation operating points. The ease of calibration and autonomy of the proposed method is demonstrated experimentally by varying a functional requirement. Moreover, the robustness of the proposed method to adapt to changing operating conditions is showcased for varying ambient temperature and relative humidity in the vehicle climatic chamber. The proposed RL-based control method significantly reduces the calibration time by 69% in comparison to the benchmark approach by circumventing the generation of system models and model-in-the-loop control development.

Future work will focus on applying RL to a multi-input-multi-output (MIMO) problem, i.e., calibration of $\mathcal{R}_1$ and $\mathcal{R}_2$ to deal with the input-output coupling in the heat pump control system and deal with multiple system constraints. Additionally, the method's robustness will be evaluated further by examining the impact of different refrigerant types and hyperparameters settings. Further research will also investigate the Reinforcement Learning-based control for learning of $\mathcal{R}_1$ and $\mathcal{R}_2$ on the road.

CRediT authorship contribution statement

Prasoon Garg: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization; **Emilia Silvas:** Writing – review & editing, Supervision; **Frank Willems:** Writing – review & editing, Supervision, Resources, Funding acquisition.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgement

The authors would like to thank DENSO for financial support and the Energy Systems R&D team at DENSO Aachen Engineering Centre (AEC) for constructive discussions.

References

Aradi, S. (2022). Survey of deep reinforcement learning for motion planning of autonomous vehicles. *IEEE Transactions on Intelligent Transportation Systems*, *23*(2), 740–759.

Atkinson, C. (2014). Fuel efficiency optimization using rapid transient engine calibration. In *Sae world congress, sae technical paper 2014-01-2359* 2014-01-2359 in SAE Technical Paper. Society of Automotive Engineers (SAE).

Garg, P. (2024). Intelligent Data-Driven Control Calibration: For Automotive Powertrains. PhD thesis Eindhoven University of Technology.

Garg, P., Puts, R., Mulder, L.J., van Moergastel, B., & Willems, F. (2023). Automated calibration of an automotive thermal control system using reinforcement learning. *AVL 10th International Symposium on Design Methodology*.

Garg, P., Silvas, E., & Willems, F. (2021). Potential of machine learning methods for robust performance and efficient engine control development. *IFAC-PapersOnLine*, *54*(10), 189–195.

Garg, P., Silvas, E., & Willems, F. (2025). Safe and time-efficient exploration in reinforcement learning-based control of a vehicle thermal system. *Control Engineering Practice*, *164*, 106458.

Hermes, C.J.L., Boeng, J., da Silva, D.L., Knabben, F.T., & Sommers, A.D. (2021). Evaporator frosting in refrigerating appliances: fundamentals and applications. *Energies*, *14*(18), 5991.

Karlsson, M., Ekholm, K., Strandh, P., Johansson, R., & Tunestål, P. (2010). Multiple-input multiple-output model predictive control of a diesel engine. *IFAC Proceedings Volumes*, *43*(7), 131–136.

Koch, L., Picerno, M., Badalian, K., Lee, S.-Y., & Andert, J. (2023). Automated function development for emission control with deep reinforcement learning. *Engineering Applications of Artificial Intelligence*, *117*, 105477.

Kormushev, P., Calinon, S., & Caldwell, D.G. (2013). Reinforcement learning in robotics: Applications and real-world challenges. *Robotics*, *2*(3), 122–148.

Lahlou, A., Ossart, F., Boudard, E., Roy, F., & Bakhouya, M. (2020). Optimal management of thermal comfort and driving range in electric vehicles. *Energies*, *13*(17), 4471.

Lei, N., Zhang, H., Hu, J., Hu, Z., & Wang, Z. (2025). Sim-to-real design and development of reinforcement learning-based energy management strategies for fuel cell electric vehicles. *Applied Energy*, *393*, 126030.

Liu, B., Zhang, H., & Zhu, S. (2016). An incremental v-model process for automotive development. In *23rd Asia-pacific software engineering conference (APSEC)* (pp. 225–232). IEEE.

Marvi, Z., & Kiumarsi, B. (2021). Safe reinforcement learning: A control barrier function optimization approach. *International Journal of Robust and Nonlinear Control*, *31*(6), 1923–1940.

Nian, R., Liu, J., & Huang, B. (2020). A review on reinforcement learning: Introduction and applications in industrial process control. *Computers & Chemical Engineering*, *139*, 106886.

Norouzi, A., Shahpouri, S., Gordon, D., Shahbakhti, M., & Koch, C.R. (2023). Safe deep reinforcement learning in diesel engine emission control. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, *237*(8), 1440–1453.

Pelikan, M., Goldberg, D.E., & Cantú-Paz, E. (1999). BOA: The Bayesian optimization algorithm. In *Proceedings of the 1st annual conference on genetic and evolutionary computation-volume 1* (pp. 525–532).

Pereirinha, P.G., González, M., Carrilero, I., Anseán, D., Alonso, J., & Viera, J.C. (2018). Main trends and challenges in road transportation electrification. *Transportation Research Procedia*, *33*, 235–242.

Prajna, S., & Jadbabaie, A. (2004). Safety verification of hybrid systems using barrier certificates. In *International workshop on hybrid systems: computation and control* (pp. 477–492). Springer.

Qi, X., Luo, Y., Wu, G., Boriboonsomsin, K., & Barth, M. (2019). Deep reinforcement learning enabled self-learning control for energy efficient driving. *Transportation Research Part C: Emerging Technologies*, *99*, 67–81.

Quinonero-Candela, J., & Rasmussen, C.E. (2005). A unifying view of sparse approximate Gaussian process regression. *Journal of machine learning research*, *6*(Dec), 1939–1959.

Ramos, M., Manzie, C., & Shekhar, R. (2017). Online optimisation of fuel consumption subject to NOx constraints. *IFAC-PapersOnLine*, *50*(1), 8901–8906. 20th IFAC World Congress.

Sanguesa, J.A., Torres-Sanz, V., Garrido, P., Martinez, F.J., & Marquez-Barja, J.M. (2021). A review on electric vehicles: Technologies and challenges. *Smart Cities*, *4*(1), 372–404.

Sutton, R.S., & Barto, A.G. (2018). Reinforcement learning: An introduction. MIT press.

van der Weijst, R., van Keulen, T., & Willems, F. (2019). Constrained multivariable extremum-seeking for online fuel-efficiency optimization of diesel engines. *Control Engineering Practice*, *87*, 133–144.

Wieland, P., & Allgöwer, F. (2007). Constructive safety using control barrier functions. *IFAC Proceedings Volumes*, *40*(12), 462–467.

Willems, F. (2017). The self-learning powertrain : towards smart and green transport. Eindhoven University of Technology. Inaugural lecture.

Yamanaka, Y., Matsuo, H., Tuzuki, K., Tsuboko, T., & Nishimura, Y. (1997). Development of sub-cool system. *SAE Transactions*, *106*, 129–134.

Zhang, H., Yang, G., Lei, N., Chen, C., Chen, B., & Qiu, L. (2025). Scenario-aware electric vehicle energy control with enhanced vehicle-to-grid capability: A multi-task reinforcement learning approach. *Energy*, *335*, (p. 138189).