

Article

# Controller Design for Air Conditioner of a Vehicle with Three Control Inputs Using Model Predictive Control

Trevor Parent, Jeffrey J. Defoe  and Afshin Rahimi \* 

Department of Mechanical, Automotive & Materials Engineering, University of Windsor, 401 Sunset Ave., Windsor, ON N9B 3P4, Canada; parenttrev@gmail.com (T.P.); jdefoe@uwindsor.ca (J.J.D.)

\* Correspondence: arahimi@uwindsor.ca

**Abstract:** Fuel consumption optimization is a critical field of research within the automotive industry to meet consumer expectations and regulatory requirements. A reduction in fuel consumption can be achieved by reducing the energy consumed by the vehicle. Several subsystems contribute to the overall energy consumption of the vehicle, including the air conditioning (A/C) system. The loads within the A/C system are mainly contributed by the compressor, condenser fan, and underhood aerodynamic drag, which are the components targeted for overall vehicle energy use reduction in this paper. This paper explores a new avenue for A/C system control by considering the power consumption due to vehicle drag (regulated by the condenser fan and active grille shutters (AGS)) to reduce the energy consumption of the A/C system and improve the overall vehicle fuel economy. The control approach used in this paper is model predictive control (MPC). The controller is designed in Simulink, where the compressor clutch signal, condenser fan speed, and AGS open-fraction are inputs. The controller is connected to a high-fidelity vehicle model in Gamma Technologies (GT)-Suite (which is treated as the real physical vehicle) to form a software-in-the-loop simulation environment, where the controller sends actuator inputs to GT-Suite and the vehicle response is sent back to the controller in Simulink. Quadratic programming is used to solve the MPC optimization problem and determine the optimal input trajectory at each time step. The results indicate that using MPC to control the compressor clutch, condenser fan, and AGS can provide a 37.6% reduction in the overall A/C system energy consumption and a 32.7% reduction in the error for the air temperature reference tracking compared to the conventional baseline proportional integral derivative control present in the GT-Suite model.

**Keywords:** model predictive control; vehicle air conditioning; fuel efficiency; fuel consumption optimization; vehicle energy efficiency; software-in-the-loop simulation



**Citation:** Parent, T.; Defoe, J.J.; Rahimi, A. Controller Design for Air Conditioner of a Vehicle with Three Control Inputs Using Model Predictive Control. *Modelling* **2024**, *5*, 117–152. <https://doi.org/10.3390/modelling5010008>

Academic Editors: Tomasz Nowakowski, Artur Kierzkowski, Agnieszka A. Tubis, Franciszek Restel, Tomasz Kisiel, Anna Jodejko-Pietruczuk and Mateusz Zajac

Received: 21 November 2023

Revised: 20 December 2023

Accepted: 27 December 2023

Published: 3 January 2024



**Copyright:** © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

## 1. Introduction

The most effective method for A/C system control has been continuously evolving throughout the years as the state of the art for control design has also continued to evolve. Early implementations of controllers on A/C systems include on/off, proportional integral derivative (PID), and sliding-mode controllers. Høgh and Nielsen [1] presented a new and innovative control strategy that used a compressor and an electronic expansion valve for better control of the superheat level. The system efficiency was also analyzed to find the optimal superheat setpoint, which could reduce energy consumption. The refrigerant system was modeled to allow for model-based feedback and feedforward, which improved control of the superheat level. Koo et al. [2] discussed the use of a proportional–integral (PI)-controlled cascade structure to control the relative filling and evaporation pressure of a target system. They also introduced the super-twisting-algorithm-based 2-sliding mode controller (SMC) design for the target system. The 2-SMC design was found to be more effective than the classical first-order sliding mode controller (1-SMC) in handling plant parameter uncertainty. Liu et al. have worked on a survey [3] where they provided a

comprehensive overview of different automatic control techniques that had been applied in A/C systems. The authors described the characteristics of A/C systems and discussed the variables that should be controlled. They used classification of the control approaches to explain the main features of these strategies used in A/C systems. Their paper clearly showed that, with the improvement in people's living standards and the emergence of global environmental and energy problems, there were increasing requirements for more advanced A/C control strategies. The authors summarized studies of control methods presented in the paper, and analyzed the development and new trends of A/C control approaches. Additionally, the paper discussed studies of PID control applied in A/C systems, including more advanced PID control methods to improve control system performance and to make full use of PID control advantages in the multi-input–multi-output (MIMO) A/C system control, and simplify the design and operation procedure of the control strategies. The above-mentioned controller designs are typically applied to the compressor to maintain the cabin at a specific temperature. These controllers are simple to design but most do not attempt to reduce the energy consumption of the system. Cvok et al. [4] proposed a method to optimize and control an electric vehicle heating, ventilation, and air conditioning (HVAC) system for better efficiency and thermal comfort. The authors used dynamic programming to find optimal control actions based on a simplified cabin model and static HVAC maps. They also designed a hierarchical control system that used an instantaneous optimization-based allocation algorithm and a superimposed cabin temperature controller. They verified their method by simulation for a cool-down scenario and compared it with the dynamic programming benchmark. They showed that their method could approach the benchmark by tuning the controller bandwidth and the allocation cost function. Kim et al. [5] presented the development of a mathematical dynamic model and observer-based set point control of a vapor compression cycle (VCC) system. The mean void fraction and moving boundary control volumes in a heat exchanger (HEX) were used for a lumped parameter model of an air conditioning system. A MIMO dynamic model of the VCC was presented in a nonlinear state space representation (SSR) and was reformulated into a linearized regular form for control design. A modified observer-based controller with SMC was introduced to attenuate the effects of observing approximation errors and external disturbances. The proposed control methodology ensured uniformly ultimately bounded tracking results of the desired low pressure and suction superheat. A simulation of controlling the VCC illustrated the effectiveness and performance in the presence of disturbances. Xie et al. [6] proposed an intelligent air conditioning system control strategy that can learn passengers' thermal comfort preferences. The proposed control algorithm directly and automatically decreased the cabin temperature to the passengers' preferred temperature without any manual adjustment.

More recent developments in sliding-mode controller applications show potential for energy savings related to the operation of the compressor and fans. However, these control methods still lack any consideration of vehicle drag and AGS control. For example, Huang et al. [7] presented an energy-saving controller for automotive air-conditioning/refrigeration (A/C-R) systems. The proposed controller consisted of two different time-scale layers. The outer or the slow time-scale layer, called a setpoint optimizer, is used to find the set points related to energy efficiency by using the steady-state model, whereas the inner or the fast time-scale layer is used to track the obtained set points. In the inner loop, thanks to its robustness, an SMC was utilized to track the set point of the cargo temperature.

In recent years, model predictive control applications for A/C systems have seen more research interest in academia. MPC lends itself well to A/C system control because it can directly consider constraints on inputs, states, and outputs in the control problem formulation. Liberati et al. [8] address the emerging challenge of managing smart microgrids for electric vehicle charging and various user loads while incorporating renewable energy sources and storage. Their work emphasizes the importance of advanced power systems and building automation control for real-world applications. The article introduces an economic model predictive control method that optimizes electric and heating resources in smart buildings,

achieving nearly zero energy consumption and seamless participation in demand response programs. Simulation results, based on an Italian case study, showcase the approach's effectiveness in managing electric and thermal flexibility within a building. In a review by Borreggine et al. [9] on the state-of-the-art issues and solutions for MPC in power electronics, the authors reviewed the MPC techniques for power electronics and electrical drives, with a focus on finite control set MPC and explicit MPC. The paper discussed the advantages and drawbacks of both methods, such as flexibility, performance, computation effort, and steady-state error. The paper also presented several applications of MPC for different types of converters, loads, and powertrain configurations. The paper highlighted the potential of MPC for powertrain control, especially for hybrid and electric vehicles.

More recently, MPC applications have been applied to automotive A/C systems. Most of the current literature on automotive A/C system control focuses on minimizing the compressor power consumption and the tracking error of the discharge air (cooling requirement) in electric vehicles (EVs), as discussed in a review by Huang et al. [10]. Wang et al. [11] proposed a nonlinear MPC control for EVs to minimize evaporator blower power consumption but did not consider the AGS in the analysis. Pereira et al. [12] proposed an energy management system (EMS) for a fuel cell (FC) hybrid electric vehicle based on nonlinear model predictive control (NMPC) and a recurrent neural network (RNN). The NMPC allowed the formulation of control objectives not allowed by a linear MPC, such as maximum efficiency point tracking of the FC, while the RNN accurately predicted the FC nonlinear dynamics. The EMS was implemented on a low-cost development board and tested in real time on a hardware-in-the-loop test bench equipped with a real 3-kW FC stack. The experimental results demonstrated that the NMPC EMS was able to meet the vehicle's energy demand and operate the FC in its most efficient region. A comparative study was performed between the proposed NMPC, a linear MPC, and hysteresis band control. The results of this comparative study demonstrated that the NMPC provided better fuel economy and could reduce FC degradation. Xie et al. [13] presented an Intelligent Model Predictive Control (IMPC) strategy for the A/C–cabin coupled system that includes the influences of vehicle speed and external environment on the heat exchange with the cabin. The IMPC strategy integrates the vehicle speed previewer and the self-adaptor for passengers' thermal comfort. The IMPC has a more dynamic response of compressor speed to the car speed change and can automatically adjust cabin temperature, making it satisfy the thermal preference of the passenger with little control error of cabin temperature. The IMPC strategy saves more energy than other control strategies researched in their paper. Glos et al. [14] presented an application of NMPC algorithms for energy-efficient control of cabin temperature and air quality in fully electric vehicles (FEVs). The authors developed simplified dynamic models of the FEV cabin and HVAC system and formulated an optimal control problem that minimized power consumption while satisfying the user's comfort and safety requirements. The NMPC algorithms were implemented on a microcontroller and tested in a Processor in the Loop simulation. The paper showed that the NMPC approach could reduce the heat losses through ventilation and improve the range of the FEV. The paper also discussed the challenges and limitations of the NMPC implementation, such as computational complexity, state estimation, and actuator constraints. Kibalama et al. [15] presented an energy management strategy for the air conditioning system of an electrified vehicle, based on MPC. The authors used a nonlinear model of the vapor compression refrigeration system and formulated a constrained optimal control problem to minimize the energy consumption while tracking a desired evaporator pressure. They proposed a novel implementation method that reduced the computation time by approximating the value function from a global optimal solution obtained by dynamic programming (DP). They showed that their MPC approach achieved near-optimal performance on different drive cycles and thermal load requirements, and validated it on real-time hardware. The paper did not address the robustness of the controller to uncertainties or disturbances, or the extension to other thermal management systems.

A recent study by Göltz and Sawodny [16] investigated tracking performance and energy consumption of the A/C system using various control schemes, including MPC. However, this work mainly focuses on battery thermal management for battery electric vehicles (BEVs). Additionally, opening the AGS increases the cooling effort supported by the front-end airflow, allowing other components, such as the compressor, to reduce their working load and still satisfy the cooling requirements of the vehicle, as shown in a study by El-Sharkawy et al. [17]. This AGS airflow cooling comes at a cost. Shigarkanthi et al. [18] demonstrated that, as the AGS is opened, the drag force of the underhood vehicle airflow increases due to an increase in drag coefficient, concluding that the AGS has an impact on the fuel economy and power consumption of the vehicle. By controlling the condenser fan and AGS along with the compressor, it is expected that the energy used by the system can be further reduced but the cost of increased underhood drag must be considered.

Chen et al. [19] discuss the significance of real-time optimal control in improving the energy efficiency of cabin climate control systems in EVs. They introduced a novel linear-time-varying (LTV) MPC approach, which simplifies the optimization process and conducts a comprehensive parametric study to identify robust weighting factors. Comparative analysis reveals that the LTV-MPC outperforms rule-based and nonlinear economic MPC (NEMPC) controllers, achieving faster temperature regulation with significantly lower energy consumption while maintaining comparable computational efficiency.

Coppola et al. [20] focused on improving energy performance in platoons of autonomous connected vehicles. Their study presents an innovative Eco-Driving Control Architecture that utilizes NMPC for optimizing energy consumption while maintaining precise leader tracking. Analytical proofs of exponential stability and robust stability conditions are provided, demonstrating the effectiveness of the approach in various operational scenarios and parameter uncertainties within a Mixed Traffic Simulator (MiTraS) co-simulation platform. The results validate the architecture's ability to ensure eco-driving behavior for the entire vehicle platoon.

He et al. [21] introduced a stochastic model predictive controller (SMPC) for enhancing the energy efficiency of EV air conditioning systems. It employs a Markov-chain-based velocity predictor to anticipate future disturbances, and systematically quantifies the sensitivity of the electrified AC system to solar radiation, ambient temperature, and relative air flow speed. Comparative analysis with a rule-based controller and dynamic programming benchmark reveals that SMPC improves AC energy economy by 12%, reduces cabin temperature variation by over 50.4%, and enhances overall cabin comfort.

Huang et al. [22] presented a robust model predictive controller for enhancing temperature performance and reducing energy consumption in automotive air-conditioning/refrigeration (A/C-R) systems with three-speed and continuously-varying compressors. It includes a discrete MPC with a carefully chosen terminal weight for robustness under various conditions. The study also presents hybrid controllers that combine on/off control and discrete MPC, as well as a continuous MPC for systems with variable components, demonstrating energy savings of up to 23% compared to conventional on/off controllers in both experimental and simulation results.

Based on the reviewed literature, the existing work on the application of MPC in A/C for commercial vehicles does not consider the impact of the AGS actuator nor the power consumption due to underhood aerodynamic drag [16]. Additionally, some studies have not considered all possible control inputs in the overall design, such as missing AGS [11]. Furthermore, most studies, especially in recent years, focus on EVs as energy consumption and management has a much greater role in such vehicles [12–15]. This has left fossil-fueled energy management and A/C energy efficiency on the back burner in recent years. Additionally, the applications of nonlinear controllers, although more advanced and capable, require more computation and consideration compared to simpler alternatives.

**Contribution:** To the best of the authors' knowledge [23,24], the consideration of AGS control to reduce the overall vehicle power consumption and regulate the underhood airflow for cabin cooling has not been considered in the literature. Therefore, the work

in this paper addresses a gap for non-electric vehicles, and additional control inputs and modeling enhancements not currently addressed in the literature. The starting point for this study is the modeling and control work by Zhang [25], where various control approaches were attempted for an internal-combustion-powered vehicle, with both the condenser fan and compressor clutch as actuators. Zhang [25] found that none of the approaches attempted was satisfactory and recommended MPC be attempted. In this paper, we apply MPC for the same vehicle, introduce the AGS as a new actuator into the control problem, and consider the additional underhood drag power consumption for the A/C system.

Several linear approximation MPCs have been formulated in the past, with the most popular being adaptive, as showcased by Dhar and Bhasin [26], and gain-scheduled model predictive controllers, as showcased by Hu et al. [27] for a commercial vehicle air brake system. These controller types will be discussed in more detail in Section 2.1. However, adaptive MPC is more promising for the application in this study due to its effectiveness for control designs where the structure of the optimization problem (number of states and constraints) remains the same across different operating points. Further discussions and rationales for the choice of adaptive MPC in this study are provided in Section 2.1. To address the identified gap in the literature and extend the work by Zhang [25] based on the system modeling improvements for the A/C system by the authors in [28], this paper has two main objectives:

**Objective 1.** In addition to controlling the compressor clutch only, determine the improvement in A/C system energy use and air-to-cabin reference temperature tracking that can be achieved by switching from PID (benchmark) to MPC.

**Objective 2.** Identify what further improvement, if any, can be achieved by including the condenser fan rotational speed and AGS position in the MPC formulation.

The remainder of this paper is organized as follows. Section 2 details background materials for the reader to understand the subsequent sections of the paper as well as the implemented methodology in this study. Section 3 includes a description of the problem formulation. Section 4 discusses the findings and implications of the results, as well as a discussion of the analysis. Finally, the conclusions of this study are offered in Section 5, followed by recommendations for future studies.

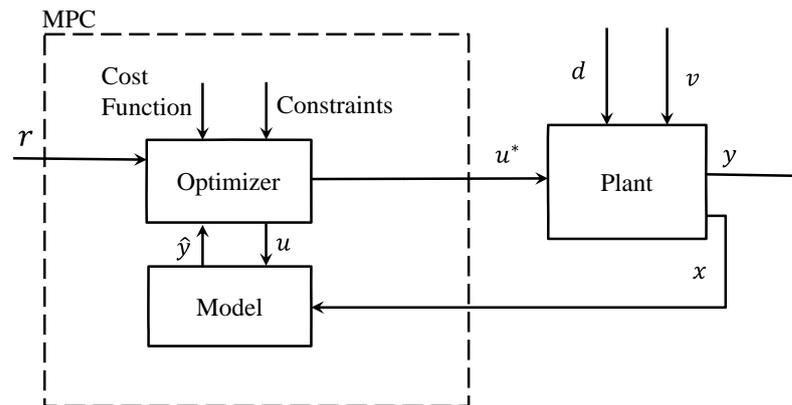
## 2. Materials and Methods

This section provides preliminary background on the theories used in this paper to help the reader follow the content more easily and to make the paper self-contained. Additionally, it provides the implemented methodology for the controller design and formulation based on the materials to help the reader follow the content with ease.

### 2.1. MPC Design

Model predictive control is an optimal control technique that leverages a precise mathematical model of a real system to proactively predict its future behavior. This predictive capability empowers MPC to anticipate potential system deviations and effectively optimize control inputs to achieve predefined performance objectives. By employing sophisticated optimization algorithms, MPC with the help of its more advanced derivatives can handle complex, nonlinear engineering systems with a high degree of accuracy and robustness, making it a valuable tool for a wide range of engineering applications. MPC's ability to incorporate constraints on both system states and control inputs ensures safe and reliable operation, even in the presence of uncertainties and disturbances. Its adaptability to various real-world scenarios, such as process control, robotics, and energy management systems, underscores its versatility and practical relevance. The combination of predictive modeling, optimization, and constraint handling renders MPC an essential asset in the pursuit of efficient and stable control solutions for complex engineering systems. A simple block diagram of an MPC is shown in Figure 1, where the dashed box includes the MPC components and their interactions, namely the Optimizer block and the Model block. The model block, also known as the prediction model, contains a simplified mathematical

model of the real system (plant). The prediction model computes the predicted output,  $\hat{y}$ . The optimizer block contains an optimization algorithm used to determine the optimal control input  $u$  based on the error predicted by the prediction model. The details of the optimizer block in this work are outlined in Section 3.3. When the optimizer computes a control signal, it sends it to the prediction model and the behavior of the real system is predicted again. This process repeats several times over an optimization window, called the prediction horizon, to forecast the future behavior of the real system. When the entire control sequence is completed, only the immediate next control action,  $u^*$ , is sent to the real system with the states  $x$  and output  $y$ , designated as the plant block in Figure 1.



**Figure 1.** MPC block diagram.

The plant can be a real physical system, such as a physical vehicle, or a virtual representation of the system, such as a high-fidelity model of a physical vehicle. In this paper, the plant is a high-fidelity model, introduced in Section 2.3. The connection to the real system in this work is detailed in Section 3.1 based on the previous work by the authors [28].

### 2.1.1. Type Selection

Selecting the appropriate MPC technique based on the system's characteristics is an important task in the overall MPC design process. A linear time-invariant MPC method is suitable for a linear system with linear constraints and a quadratic cost function. This scenario results in a convex optimization problem with a unique global optimum. Numerous numerical methods and software can be employed to solve this well-studied problem. When moving to a nonlinear system, two methods are discussed: adaptive and gain-scheduled MPC controllers. These methods rely on linearization to approximate the nonlinear system around an operating point. While effective near this point, linearization's limitations outside this range are acknowledged.

Adaptive MPC involves computing a linear model on the fly as conditions change, updating the internal plant model for the MPC controller. This approach maintains the optimization problem structure across various operating points. Alternatively, gain-scheduled MPC involves offline linearization at multiple operating points, designing separate MPC controllers for each. While offering controller independence, this method requires managing controller switching and may demand more memory.

It is beneficial to use adaptive MPC when the optimization problem structure remains consistent across various conditions. When the structure changes, gain-scheduled MPC is preferable. However, a nonlinear system that resists accurate linearization requires nonlinear MPC. This advanced method employs a nonlinear plant model for precise predictions but solving the associated nonconvex optimization problem in real time can be challenging.

Therefore, adaptive and gain-scheduled MPC can be used for systems that can be approximated by linear models, and nonlinear MPC can be employed for highly nonlinear

systems. The implementation efficiency of nonlinear MPC must be considered as it relies on specialized nonlinear solvers.

The A/C system used in this study was developed and published by the authors of this study in [28]. This model is an extension to a previously developed model by the industrial partner's collaborators where effects of underhood aerodynamics and modeling of system parameters using polynomial approximations are added for a more extensive and semi-analytical model of the system under study. The model in [28] requires an adaptive MPC for the control strategy development because it meets the following requirements for a choice of the adaptive MPC over a traditional MPC:

1. Time-varying dynamics: The system's dynamics in [28] change over time due to various factors such as varying operating conditions or system wear and tear.
2. Uncertain models: Some of the model parameters [28] are not precisely known as they are embedded in the model developed by the industrial partner where the exact values of the parameters are unknown and approximate values were used to build the semi-analytical model, and there is uncertainty about the model's accuracy.
3. Nonlinear systems: The system in [28] is highly nonlinear and for controlling systems with nonlinear behavior, where a linear model may not be sufficient, and an adaptive model helps to approximate the nonlinear dynamics better.
4. Disturbance rejection: As mentioned in item 2, some of the system parameters are not entirely known throughout the simulation as they are provided by the software-in-the-loop and depend on the driving cycle that the tests are being run over. When dealing with systems subject to disturbances that are not entirely predictable or change over time, adaptive MPC is the plausible choice.
5. Parameter variations: Some of the system parameters in [28] change over time, and their values are embedded in the software-in-the-loop provided by the industrial partner. In cases where there are abrupt changes or slow variations in system parameters, adaptive MPC can be useful.

Therefore, a variation of adaptive MPC, Successive Linearization Based Model Predictive Control (SLMPC) [29] (see Appendix A for detailed derivation), is chosen for the problem introduced in this paper due to its ability to reduce the linearization error caused by the system operating far from the trim state. In the adaptive MPC, a linear model of the system is computed on the fly as the operating conditions change and the internal prediction model is updated with each new linearized model. As the operating conditions change, only the model of the system is updated and the MPC formulation remains the same. As discussed in Section 2.3, the plant model for the A/C system used in this study has varying system parameters that are controlled by other subsystems in the vehicle, therefore making this system time-varying. The use of adaptive MPC ensures that, despite variations in system parameters, at every operating point, the linearization guarantees close agreement between the linearized model and the nonlinear plant model for the A/C system.

In addition to the type of MPC controller, there are parameters to be discussed in the design process for any MPC controller. These parameters are discussed next.

### 2.1.2. Design Parameters

For MPC design, there are essential parameters that require careful consideration to enhance both controller performance and computational efficiency. These parameters encompass the controller sample time, prediction horizon, control horizon, constraint handling, and weight assignment. Each parameter plays a pivotal role in optimizing the MPC algorithm's real-time performance while ensuring effective control of dynamic systems.

- **Sample time:** The choice of the sample time determines the controller's operational frequency, impacting its ability to respond to disturbances and setpoint changes. Balancing between rapid disturbance rejection and computational load, it is advisable to accommodate 10 to 20 samples within the open-loop system's rise time [30]. This ensures a responsive controller without overwhelming computational demands. Further details about this parameter in this study are provided in Section 3.4.2.

- **Prediction horizon:** The prediction horizon influences the extent to which the controller anticipates future plant behavior. A cautious prediction horizon encompasses a timeframe that captures significant system dynamics, preventing untimely responses to disturbances. To account for open-loop transient responses, selecting 10 to 20 samples is recommended [30]. Further details about this parameter in this study are provided in Section 3.4.2.
- **Control horizon:** Conversely, the control horizon determines the number of future control inputs that shape the predicted plant output. A smaller control horizon minimizes computational efforts but may compromise maneuverability. An optimal balance can be struck by setting the control horizon to 10–20% of the prediction horizon, ensuring a minimum of 2–3 steps to account for effective control [30]. Further details about this parameter in this study are provided in Section 3.4.2.
- **Constraints:** Constraints on inputs, input rate changes, and outputs can be imposed within the MPC framework. These constraints may be either hard or soft. The distinction lies in the feasibility of violation: hard constraints remain inviolable, while soft constraints can be breached within limits. Balancing the conflicting interests of inputs and outputs, it is advised to adopt soft constraints for outputs, eschewing simultaneous hard constraints on inputs and their rate of change. This minimizes the risk of infeasibility and ensures more adaptable control [30]. Further details about this parameter in this study are provided in Sections 3.3 and 3.4.3.
- **Weights:** Weight assignment is a critical aspect of optimizing control performance. Weights gauge the significance of objectives in MPC. These objectives involve tracking setpoints and achieving smooth control maneuvers. Achieving a harmonious performance balance necessitates cautiously weighing input rates and outputs relative to each other. Additionally, adjusting weights within these categories fine-tunes performance to the specific control objectives [30]. Further details about this parameter in this study are provided in Sections 3.3 and 3.4.4.

One can see the role of distinct MPC design parameters in shaping controller efficacy and computational efficiency. By strategically tuning these parameters, control engineers can harness MPC's potential to efficiently regulate dynamic systems while navigating the intricacies of real-time optimization.

## 2.2. Discrete State-Space General Form

MPC strategies commonly use discrete state-space formulations for the mathematical modeling of the system of interest. A continuous-time linear state-space model form must be discretized to form a discrete-time linear state-space model. Typically, MPC algorithms are designed in discrete time. A great amount of MPC designs have been developed using discrete-time approaches and a much smaller amount of continuous-time MPCs are present in the literature. In this work, a discrete-time MPC has been chosen due to the nature of the communication between the MPC and the real system, as discussed in Section 3.1.

The standard form of a continuous state-space system can be written as [31]:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned}\quad (1)$$

where  $A$ ,  $B$ ,  $C$ , and  $D$  are matrices describing the system dynamics.  $x$  is the state vector,  $u$  is the controlled input vector, and  $y$  is the output vector.

The continuous system in (1) can be written in a discrete state-space system as [32]:

$$\begin{aligned}x_{k+1} &= A_d x_k + B_d u_k \\ y_k &= C_d x_k + D_d u_k\end{aligned}\quad (2)$$

where  $A_d$ ,  $B_d$ ,  $C_d$ , and  $D_d$  are matrices describing the linearized system dynamics.  $k$  denotes the current timestep,  $x_k$  is the state vector,  $u_k$  is the controlled input vector, and  $y_k$  is the output vector. One can use forward Euler's method to convert (1) to (2).

**Forward Euler's method** is a simple discretization method for continuous-time systems. It is based on the approximation of the first derivative at  $t = kT$  where  $T$  is the time step for the discretization. Having [31]

$$\dot{x}(t) = \frac{dx(t)}{dt} \approx \frac{1}{T}(x((k+1)T) - x(kT)) \quad (3)$$

Applying the approximate formula in (3) to the SSR system equation, we have [31]

$$\begin{aligned} \frac{1}{T}(x((k+1)T) - x(kT)) &\approx Ax(kT) + Bu(kT) \\ x((k+1)T) &\approx (I + TA)x(kT) + TBu(kT) \end{aligned} \quad (4)$$

or [31]

$$\begin{aligned} x[k+1] &\approx (I + TA)x[k] + TBu[k] \\ x[k+1] &\approx A_d x[k] + B_d u[k] \end{aligned} \quad (5)$$

where  $A_d = I + TA$  and  $B_d = TB$  are the discrete system matrices for the discrete SSR of the system with sampling time  $T$ . The same approach can be followed to obtain  $C_d$  and  $D_d$  that similar to  $B_d$  would only have a multiplication by  $T$ . The sampling time for the discretization in this study was chosen as  $T = 1$  s as detailed in Section 2.1.2. In this paper, the system equations of the dynamics of an automotive A/C system, as detailed in Section 2.3, are considered. These equations are nonlinear and, therefore, are linearized to conform to the standard continuous-time form in (1). These continuous-time equations are then discretized to conform to the discrete-time form in (2) using the forward Euler method. The details of the full process are described in Section 3.

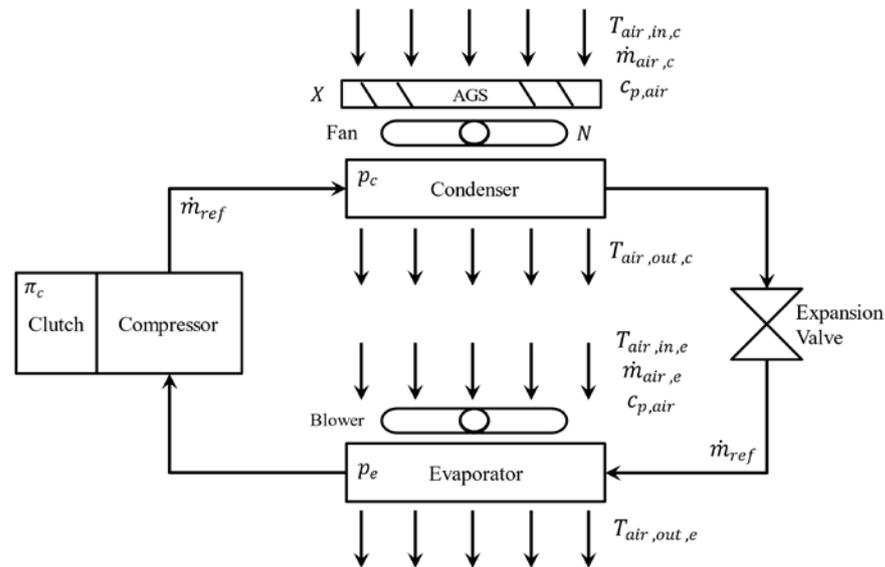
### 2.3. Introduction of A/C System

A simple diagram of the automotive A/C system used in this study is depicted in Figure 2. The A/C system comprises four main components, the compressor, condenser, evaporator, and expansion valve. The controllable actuators of interest in this work are the compressor clutch, which switches the compressor (used to move the refrigerant throughout the cooling circuit) on and off, and the fan and AGS used to regulate the front-end airflow passing through the condenser. Compressor clutch  $\pi_c$  is not a continuous variable in practice and is a discrete binary variable as either engaged (with the value of 1) or disengaged (with the value of 0). In this study,  $\pi_c$  is initially treated as a continuous variable to be computed via optimization. When the MPC computes the continuous compressor clutch signal  $\pi_c$ , it rounds it to a value of either 0 or 1 prior to sending it to the GT-Suite model to ensure it is in the discrete form that GT-Suite expects (see Appendix B for a discussion on MPC relaxation and justification for applicability of this approach in this context).

The model used by the MPC is a semi-analytical nonlinear model of the A/C system by Parent et al. [28] (the authors of this study) as a prerequisite to run this study for controller design development. The dynamics of this model can be described by the heat exchanger pressures  $p_c$  for condenser and  $p_e$  for evaporator. The first system output of interest is the temperature of the evaporator air stream. The underlying continuous-time mathematical equations for the heat exchanger pressures and temperatures are [28]:

$$\begin{aligned} \dot{p}_c &= \frac{[Q_c \rho_a c_{p,air} (T_{air,in} - T_{air,out}) + \pi_c \dot{m}_{ref} (h_{in} - h_{out})]}{V \left[ (1-\gamma) \frac{\partial(\rho_l h_l)}{\partial p} + \gamma \frac{\partial(\rho_g h_g)}{\partial p} + (\rho_g h_g - \rho_l h_l) \frac{\partial \bar{T}}{\partial p} - 1 + \frac{m_{H^2O}}{V} \left( \frac{\partial T_{wall}}{\partial p} \right) \right]} \\ \dot{p}_e &= \frac{[Q_e \rho_a c_{p,air} (T_{air,in} - T_{air,out}) + \pi_c \dot{m}_{ref} (h_{in} - h_{out})]}{V \left[ (1-\gamma) \frac{\partial(\rho_l h_l)}{\partial p} + \gamma \frac{\partial(\rho_g h_g)}{\partial p} + (\rho_g h_g - \rho_l h_l) \frac{\partial \bar{T}}{\partial p} - 1 + \frac{m_{H^2O}}{V} \left( \frac{\partial T_{wall}}{\partial p} \right) \right]} \\ T_{air,out,c} &= T_{ref,c} + (T_{air,in,c} - T_{ref,c})^{(-NTU_c)(K_{ca})} \\ T_{air,out,e} &= T_{ref,e} + (T_{air,in,e} - T_{ref,e})^{(-NTU_e)(K_{ca})} \\ NTU &= \frac{\alpha A_s [1 - F_{fin}(1 - \eta_{FA})]}{m_a c_{p,air}} \end{aligned} \quad (6)$$

where  $Q_c$  is the volume flow rate of the air through the condenser [ $\text{m}^3/\text{s}$ ],  $\rho_a$  is the air density [ $\text{kg}/\text{m}^3$ ],  $h_{in}$  and  $h_{out}$  are the inlet and outlet refrigerant enthalpies [joules],  $\rho_l, \rho_g$  are the densities [ $\text{kg}/\text{m}^3$ ] and  $h_l, h_g$  are the enthalpies [joules] for the refrigerant calculated at saturated liquid and vapor states, respectively.  $c_H$  is the specific heat of the HEX material [ $\text{J}/\text{K}\cdot\text{kg}$ ],  $c_{p,air}$  is the specific heat of air at constant pressure [ $\text{J}/\text{K}\cdot\text{kg}$ ],  $\bar{\gamma} \in [0, 1]$  is the mean void fraction of the refrigerant,  $V$  is the volume of the HEX [ $\text{m}^3$ ],  $m_H$  is the material mass of the HEX [kg],  $T_{wall}$  is the wall temperature of the HEX [K],  $p_c$  and  $p_e$  are the refrigerant pressures in the condenser and evaporator [kPa],  $\pi_c$  is the compressor clutch engagement [on/off],  $\dot{m}_{ref}$  is the refrigerant mass flow rate [kg/s],  $Q_e$  is the volume flow rate of the air from the blower fan [ $\text{m}^3/\text{s}$ ],  $T_{air,in}$  and  $T_{air,out}$  are the inlet air temperatures to the condenser and evaporator [K],  $T_{ref}$  is the HEX refrigerant temperature [K],  $NTU$  is the number of transfer units of the air, and  $K_{ca}$  and  $K_{ea}$  act as multipliers on the Nusselt number.  $\alpha$  is the convective heat transfer coefficient of the air [ $\text{W}/\text{m}^2\cdot\text{K}$ ],  $A_s$  is the external surface area of the HEX [ $\text{m}^2$ ],  $F_{fin} \in [0, 1]$  is the fraction of air-to-structure surface area on the fins,  $\eta_{EA}$  is the air-side fin efficiency, and  $\dot{m}_a$  is the mass flow rate of the air [kg/s].



**Figure 2.** Simple A/C system diagram.

From the developments in [28], the definitions for  $Q_c$  and  $T_{ref,c}$  in continuous time are as follows:

$$\begin{aligned} Q_c &= 0.07722\hat{N} + 0.01057v + 4.913 \times 10^{-5}v^2 + 0.2039X\hat{N} + 0.002671Xv \\ T_{ref,c} &= 8.4 \cdot 10^{-9}p_c^3 - 4.9 \cdot 10^{-5}p_c^2 + 0.12p_c - 34.9 \end{aligned} \quad (7)$$

where  $\hat{N}$  is the normalized rotational speed of the condenser fan between zero and one,  $X$  is the open-fraction of the AGS between zero and one, and  $v$  is the vehicle speed [m/s]. The full list of fitted curves is available in Appendix A of [28] in Equations (17)–(20) and not further provided here for brevity.

Note that, in the system model in (6), there are terms that are updated from outside of the A/C system model and controller by other vehicle subsystems. These parameters are updated as the vehicle is operating and not controlled by the controller designed in this study for the A/C system. Hence, they are treated as system parameters in (6) that vary with time. The varying system parameters controlled by other subsystems in the vehicle dynamics are defined as follows:

$$\theta = \begin{bmatrix} \dot{m}_{ref} \\ Q_e \\ T_{air,in,c} \\ T_{air,in,e} \\ h_{in,c} \\ v \\ h_{in,comp} \end{bmatrix} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \\ \theta_4 \\ \theta_5 \\ \theta_6 \\ \theta_7 \end{bmatrix} \quad (8)$$

where  $\dot{m}_{ref}$  is the mass flow rate of the refrigerant set by the compressor in GT-Suite [kg/s],  $Q_e$  is the volumetric flow rate of the air set by the evaporator blower model in GT-Suite [ $\text{m}^3/\text{s}$ ],  $T_{air,in,c}$  and  $T_{air,in,e}$  are the air temperatures entering the condenser and evaporator [K],  $h_{in,c}$  is the inlet enthalpy for the condenser [joules],  $v$  is the vehicle speed [m/s], and  $h_{in,comp}$  is the inlet enthalpy for the compressor [joules]. All these parameters are measured and sent from the vehicle energy management (VEM) model to the MPC. It is important to note that this makes the system under study a nonlinear time-varying system, which would require an adaptive MPC type to handle the nonlinearities by linearizing the model at each operating point, as discussed in Section 2.1.1.

Since system energy consumption is of interest, the model includes the power consumption definitions in continuous time as [28]:

$$\begin{aligned} \dot{W}_{comp} &= \frac{\dot{m}_{ref}(h_{out}-h_{in})}{\eta_{comp}} \\ \dot{W}_{fan} &= \frac{Q_c \Delta p_t}{\eta_{fan}} \\ \Delta \dot{W}_{drag} &= \frac{1}{2} \rho_a A_{fr} v^3 \Delta C_d \\ g_2 = \dot{W}_{total} &= \dot{W}_{comp} + \dot{W}_{fan} + \Delta \dot{W}_{drag} \end{aligned} \quad (9)$$

where  $\dot{W}_{comp}$  is the compressor's power consumption [j/s],  $\dot{W}_{fan}$  is the fan's power consumption [j/s], and  $\Delta \dot{W}_{drag}$  is power consumption associated with additional drag on the vehicle due to underhood airflow, which is set by the vehicle speed, fan rotational speed, and AGS open-fraction [j/s].  $\dot{W}_{total}$  is the total power consumption from A/C comprising of the previous three components [j/s].  $\dot{m}_{ref}$  is the mass flow rate of the refrigerant [ $\text{m}^3/\text{s}$ ],  $h_{in}$  and  $h_{out}$  are the enthalpies of the refrigerant at the inlet and outlet of the compressor, respectively [j/kg],  $\eta_{comp}$  is the ratio of isentropic efficiency to volumetric efficiency of the compressor,  $\Delta p_t$  is the pressure rise required of the fan [Pa],  $\eta_{fan}$  is the fan's adiabatic efficiency,  $A_{fr}$  is the frontal area of the vehicle [ $\text{m}^2$ ],  $\Delta C_d$  is the increase in drag coefficient due to additional flow through the underhood at a given vehicle speed due to the AGS being open and/or the fan being on, and  $v$  is the vehicle speed [m/s].

With all the required background information discussed above, next, the problem definition is discussed in more detail.

### 3. Problem Definition

This section provides a step-by-step formulation of the problem studied in this paper.

#### 3.1. Introduction

To conduct simulations to determine the desired MPC design parameters, the industrial partner supplied a high-fidelity model of the physical vehicle in GT-Suite v2020 (Build 3.0002), a multi-physics simulation software created by Gamma Technologies. This high-fidelity model is treated as a substitute for the real physical vehicle in this work (plant). GT-Suite v2020 (Build 3.0002) and MathWorks' Simulink 2021b have built-in cross-platform communication capabilities that allow signals to be sent to and from the two software packages. Therefore, MATLAB and Simulink are used for the MPC implementation. In the GT-Suite model, there are numerous independent cases set up for simulation, each representing a different standard drive cycle with time-series data loaded for each of the

driving conditions, some of which are the system parameters in (8) required by the MPC from GT-Suite, such as the vehicle speed. The remaining elements of (8) are computed by the GT-Suite model and sent to the MPC, such as  $m_{ref}$  and  $Q_e$ . In this work, the MPC simulations are conducted for the SC03 drive cycle [33], which is a standard drive cycle used by the Environmental Protection Agency (EPA) to evaluate the vehicle fuel economy where the A/C system is active. The total duration of this drive cycle is 575 s.

### 3.2. System Equations

In this paper, an adaptive MPC, known as successive linearization-based MPC (SLMPC), is applied to the automotive A/C system illustrated in Figure 2, i.e., the A/C system is the plant. To define the elements of  $A$ ,  $B$ ,  $C$ , and  $D$  used in the prediction model described in (2), the system of nonlinear equations described in (6)–(9) must be linearized. These equations are linearized on the fly using the adaptive MPC approach described in Section 2.1, allowing the trim state to always be the current working state of the plant. The Jacobian matrices that map the system nonlinear equations in (6) to the linearized state-space representation  $A$ ,  $B$ ,  $C$ ,  $D$  are calculated at every time step using the MATLAB *jacobian* function, where *jacobian*( $f, v$ ) computes the Jacobian matrix of symbolic function  $f$  with respect to  $v$ . The  $(i, j)$  element of the result of  $\partial f(i)/\partial v(j)$  is as follows:

$$\begin{cases} A = \text{jacobian}([f_1, f_2], [x_1, x_2]) \\ B = \text{jacobian}([f_1, f_2], [u_1, u_2, u_3]) \\ C = \text{jacobian}([g_1, g_2], [x_1, x_2]) \\ D = \text{jacobian}([g_1, g_2], [u_1, u_2, u_3]) \end{cases} \quad (10)$$

where  $f_1$  is the definition of  $\dot{p}_c$ ,  $f_2$  is the definition of  $\dot{p}_e$ ,  $g_1$  is the definition of  $T_{air,out,e}$  in (6), and  $g_2$  is the definition of  $\dot{W}_{total}$  in (9).  $x$  is defined in (27);  $u$  is defined in (28). Additionally, the collective terms  $g_1$  and  $g_2$  capture the system outputs  $y$  as defined in (29). Note that, due to the form of the definitions for  $f_1$ ,  $f_2$ ,  $g_1$ , and  $g_2$ , the full form of the Jacobians in (10) is not provided here as it would be infeasible to fit all the terms in a presentable format.

The numerical values of the state-space matrices are computed using a successive linearization approach [29], SLMPC, as a common approach adopted for MPC designs [34]. The concept of this approach is simple. At the beginning of each control iteration, the Jacobian matrices are evaluated at the current operating conditions of the complex system model, that being the current system states, controlled inputs, and system parameters. This will update the numerical values of the state-space matrix entries. By re-evaluating the state-space model at the current operating conditions, one ensures that the trim state is close to the current operating point. The intention of this is to reduce the linearization error caused by deviations from the trim state. Thus, the working condition of the complex system model will be treated as the trim state. Full theoretical development and proof of applicability for SLMPC can be found in [29]. This allows the controller to be initialized at the current system state at each time step, allowing accuracy to be maintained throughout the simulation since the controller will always work in the vicinity of the calculated trim state.

### 3.3. Quadratic Programming Formulation

Quadratic programming is an optimization process used to solve mathematical problems containing quadratic functions. The optimal solution to a quadratic programming problem is found by minimizing the following objective function,  $J$  [32]:

$$J = \min \left[ \frac{1}{2} X_{qp}^T H X_{qp} + f^T X_{qp} \right] \quad (11)$$

where  $X_{qp}$  is a vector containing the control objectives shown in (12),  $H$  is a symmetric matrix containing relative weightings for each control objective in the quadratic term of the cost function, and  $f$  is a vector containing relative weightings for each control objective in

the linear term of the cost function. In (11), the elements of the vector  $X_{qp}$  are minimized to achieve the smallest possible value for the objective function.

For the MPC controllers, the controller is typically designed to track reference values for the states and outputs of the system while also considering the magnitude and rate of change of the controlled input values. In this case, the elements of  $X_{qp}$  are as follows [32]:

$$X_{qp} = \begin{bmatrix} \Delta u_0 \\ x'_1 \\ \vdots \\ \Delta u_{h-1} \\ x'_h \\ y'_1 \\ \vdots \\ y'_h \end{bmatrix}_{h(n_x+n_u+n_y)} \tag{12}$$

where the definitions of these terms are provided further below. This description of  $X_{qp}$  is for a discrete-time case since the approach used for the controller design in this study is also discrete-time due to the nature of the controller type. The dimension of the vector depends on the prediction horizon length  $h$ , the number of states  $n_x$ , number of inputs  $n_u$ , and the number of outputs  $n_y$ . The elements of the  $H$  matrix can be used to change the relative importance of each of the control objectives corresponding to the terms in the  $X_{qp}$  vector as outlined in (12).

The typical state-space equations can be modified to include the output and state reference tracking deviations. This will make forming the QP constraints much easier. The goal of the following modifications to the state-space model form is to explicitly show the output setpoint deviations, state setpoint deviations, and the rate of change of the controlled inputs since these are included in the  $X_{qp}$  vector of the objective function. In QP problem formulations, linear equality constraints containing the  $X_{qp}$  vector can be applied to the problem. These constraints can be used to mathematically define the elements of  $X_{qp}$ . This can be done by treating the discrete state-space model in (2) as an equality constraint and reorganizing the formulation to contain the elements of  $X_{qp}$  explicitly. The state-space formulation described in (2) is reorganized to explicitly include the elements of (12) using the approach outlined in [32]. As a special case of the general form of the objective function for QP in (11) when  $H$  is symmetric positive-definite, the objective function reduces to  $J = \min \left[ \frac{1}{2} \|X_{qp} - X_{qp,s}\|_{H'}^2 \right]$  where  $H = H'TH'$  follows from the Cholesky decomposition of  $H$  and  $f = -H'TX_{qp,s}$ . Hence, given the system in (2), the special case of the objective function in QP can be expanded as [32]:

$$\left\{ \begin{aligned} J &= \sum_{i=1}^h \left( \|y_{k+i} - y_{k+i,s}\|_{Q_y}^2 + \|x_{k+i} - x_{k+i,s}\|_{Q_x}^2 + \|u_{k+i-1}\|_{R_u}^2 + \|\Delta u_{k+i-1}\|_{R_{\Delta u}}^2 \right) \\ S.t. &: \text{Dynamic Model and Constraints} \end{aligned} \right. \tag{13}$$

where the subscript  $s$  denotes a setpoint,  $i$  is the current step in the time horizon window, and  $I$  is the identity matrix. Note that subscript  $i$  is from 0 to the time horizon  $h$  as the dimensions of these vectors change with the size of the time horizon. To make the formulation easier, the system equations are written in

$$\begin{cases} x_{k+1} = A_d x_k + B_d \Delta u_k + B_d u_{k-1} \\ u_k = u_{k-1} + \Delta u_k \\ y_{k+1} = C_d x_{k+1} + D_d \Delta u_k + D_d u_{k-1} \end{cases} \tag{14}$$

where  $\Delta u_k \triangleq u_k - u_{k-1}$ . Rewriting the model in this form incorporates the rate of change of the controller inputs into the state-space model. The next step is to rewrite the state-space model in terms of setpoint deviations. So, the general form of the model equation becomes:

$$\begin{cases} x_{k+1} - x_{k+1,s} = A_d(x_k - x_{k,s}) + A_d x_{k,s} + B_d \Delta u_k + B_d u_{k-1} - x_{k+1,s} \\ u_k = u_{k-1} + \Delta u_k \\ y_{k+1} - y_{k+1,s} = C_d(x_{k+1} - x_{k+1,s}) + C_d x_{k+1,s} + D_d \Delta u_k + D_d u_{k-1} - y_{k+1,s} \end{cases} \quad (15)$$

Or in a compact form as

$$\begin{cases} \begin{bmatrix} x_{k+1} - x_{k+1,s} \\ u_k \end{bmatrix} = \begin{bmatrix} A_d & B_d \\ 0_{n_u \times n_x} & I_{n_u \times n_u} \end{bmatrix} \begin{bmatrix} x_k - x_{k,s} \\ u_{k-1} \end{bmatrix} \\ \quad + \begin{bmatrix} B_d \\ I_{n_u \times n_u} \end{bmatrix} \Delta u_k - \begin{bmatrix} x_{k,s} \\ 0_{n_u \times 1} \end{bmatrix} + \begin{bmatrix} A_d & B_d \\ 0_{n_u \times n_x} & I_{n_u \times n_u} \end{bmatrix} \begin{bmatrix} x_{k,s} \\ 0_{n_u \times 1} \end{bmatrix} \\ y_{k+1} - y_{k+1,s} = \begin{bmatrix} C_d & 0_{n_y \times n_u} \end{bmatrix} \begin{bmatrix} x_{k+1} - x_{k+1,s} \\ u_k \end{bmatrix} \\ \quad + \begin{bmatrix} 0_{n_u \times n_x} & D_d \end{bmatrix} \Delta u_k + \begin{bmatrix} 0_{n_u \times n_x} & D_d \end{bmatrix} \begin{bmatrix} x_k - x_{k,s} \\ u_{k-1} \end{bmatrix} \\ \quad - y_{k+1,s} + \begin{bmatrix} C_d & 0_{n_y \times n_u} \end{bmatrix} \begin{bmatrix} x_{k+1,s} \\ 0_{n_u \times 1} \end{bmatrix} \end{cases} \quad (16)$$

If we define

$$\begin{aligned} x'_k &= \begin{bmatrix} x_k - x_{k,s} \\ u_{k-1} \end{bmatrix}, \quad A' = \begin{bmatrix} A_d & B_d \\ 0_{n_u \times n_x} & I_{n_u \times n_u} \end{bmatrix}, \quad B' = \begin{bmatrix} B_d \\ I_{n_u \times n_u} \end{bmatrix}, \quad x'_{k,s} = \begin{bmatrix} x_{k,s} \\ 0_{n_u \times 1} \end{bmatrix} \\ y'_k &= y_k - y_{k,s}, \quad C' = \begin{bmatrix} C_d & 0_{n_y \times n_u} \end{bmatrix}, \\ D' &= \begin{bmatrix} 0_{n_u \times n_x} & D_d \end{bmatrix}, \quad Q'_x = \begin{bmatrix} Q_x & 0_{n_x \times n_u} \\ 0_{n_u \times n_x} & R_u \end{bmatrix} \end{aligned} \quad (17)$$

The form in (16) can be rearranged for time step 0 in the prediction horizon, where we initiate the dynamics with  $x_0, y_0$ , and  $u_{-1}$ , and given an equality constraint can be written in the form of  $A_{eq} X_{qp} = B_{eq}$

$$\begin{cases} x'_1 = A' x'_0 + B' \Delta u_0 - x'_{1,s} + A' x'_{0,s} \\ y'_1 = C' x'_1 + D' \Delta u_0 + D' x'_0 - y_{1,s} + C' x'_{1,s} \end{cases} \xrightarrow{A_{eq} X_{qp} = B_{eq} \text{ Form}} \begin{cases} x'_1 - B' \Delta u_0 = A' \begin{bmatrix} x_0 \\ u_{-1} \end{bmatrix} - x'_{1,s} \\ y'_1 - C' x'_1 - D' \Delta u_0 = -y_{1,s} + C' x'_{1,s} + D' \begin{bmatrix} x_0 \\ u_{-1} \end{bmatrix} \end{cases} \quad (18)$$

And for any other time step between  $i = 1$  up to  $h - 1$  it can be written as follows:

$$\begin{cases} x'_{k+1} = A' x'_k + B' \Delta u_k - x'_{k+1,s} + A' x'_{k,s} \\ y'_{k+1} = C' x'_{k+1} + D' \Delta u_k + D' x'_k - y_{k+1,s} + C' x'_{k+1,s} \end{cases} \xrightarrow{A_{eq} X_{qp} = B_{eq} \text{ Form}} \begin{cases} x'_{k+1} - A' x'_k - B' \Delta u_k = A' x'_{k,s} - x'_{k+1,s} \\ y'_{k+1} - C' x'_{k+1} - D' \Delta u_k = -y_{k+1,s} + C' x'_{k+1,s} + D' x'_k \end{cases} \quad (19)$$

Hence, the objective function in (13) becomes:

$$\begin{cases} J = \sum_{i=1}^h \left( \|y'_{k+i}\|_{Q_y}^2 + \|x'_{k+i}\|_{Q'_x}^2 + \|\Delta u_{k+i-1}\|_{R_{\Delta u}}^2 \right) \\ \text{S.t. : Dynamic Model and Constraints} \end{cases} \quad (20)$$

which can be written in compact form as:

$$\begin{cases} J = \sum_{i=1}^h X_{qp}^T H X_{qp} \\ \text{S.t.} : A_{eq} X_{qp} = B_{eq} \text{ and } lb_X \leq X_{qp} \leq ub_X \end{cases} \quad (21)$$

where

$$A_{eq} = \begin{bmatrix} -B' & I_{n_x} & 0 & 0 & \cdots & \cdots & \cdots & \cdots & 0 \\ 0 & -A' & -B' & I_{n_x} & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & -A' & -B' & I_{n_x} & 0 & \cdots & 0 \\ -D' & -C' & \cdots & \cdots & \cdots & 0 & I_{n_y} & \cdots & 0 \\ \vdots & \ddots & 0 \\ -D' & \cdots & \cdots & \cdots & \cdots & -C' & \cdots & 0 & I_{n_y} \end{bmatrix}_{h(n_x+n_u+n_y) \times h(n_x+2n_u+n_y)} \quad (22)$$

$$B_{eq} = \begin{bmatrix} A' \begin{bmatrix} x_0 \\ u_{-1} \end{bmatrix} - x'_{1,s} \\ A' x'_{1,s} - x'_{2,s} \\ \vdots \\ A' x'_{h-1,s} - x'_{h,s} \\ -y_{1,s} + C' x'_{1,s} + D' \begin{bmatrix} x_0 \\ u_{-1} \end{bmatrix} \\ \vdots \\ -y_{h,s} + C' x'_{h,s} + D' x'_h \end{bmatrix}_{h(n_x+n_u+n_y) \times 1} \quad (23)$$

with the lower and upper bounds for the entire length of the prediction horizon as

$$lb = \begin{bmatrix} lb_u - ub_u \\ [lb_x - x_{1,s} \\ lb_u \\ \vdots \\ lb_u - ub_u \\ [lb_x - x_{h,s} \\ lb_u \\ lb_y - y_{1,s} \\ \vdots \\ lb_y - y_{h,s} \end{bmatrix}_{h(n_x+n_u+n_y) \times 1} \quad (24)$$

$$ub = \begin{bmatrix} ub_u - lb_u \\ [ub_x - x_{1,s} \\ ub_u \\ \vdots \\ ub_u - lb_u \\ [ub_x - x_{h,s} \\ ub_u \\ ub_y - y_{1,s} \\ \vdots \\ ub_y - y_{h,s} \end{bmatrix}_{h(n_x+n_u+n_y) \times 1} \quad (25)$$

where  $lb_u$  and  $ub_u$  are the lower and upper bounds on the inputs, and  $lb_x$ ,  $ub_x$ ,  $lb_y$ , and  $ub_y$  are the lower and upper bounds for the states and outputs, respectively. Each element of the  $X_{qp}$  vector can be weighted individually, providing more significance to prioritizing certain control objectives over others. The weighting matrix  $H$  for the control objectives within the objective function (output reference tracking, state reference tracking, input value magnitude, and rate of change of the input commands) is [32]

$$H = \begin{bmatrix} \begin{bmatrix} R_{\Delta u} & 0 \\ 0 & Q'_x \end{bmatrix} & 0 & \cdots & \cdots & \cdots & 0 \\ 0 & \ddots & \ddots & \ddots & \ddots & \vdots \\ \vdots & \ddots & \begin{bmatrix} R_{\Delta u} & 0 \\ 0 & Q'_x \end{bmatrix} & 0 & \cdots & 0 \\ 0 & \cdots & 0 & Q_y & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & \cdots & \cdots & 0 & Q_y \end{bmatrix}_{h(n_x+2n_u+n_y) \times h(n_x+2n_u+n_y)} \quad (26)$$

$$Q'_x = \begin{bmatrix} Q_x & 0_{n_x \times n_u} \\ 0_{n_u \times n_x} & R_u \end{bmatrix}_{(n_x+n_u) \times (n_x+n_u)}$$

where  $R_{\Delta u}$  is the weighting of the rate of change of the inputs,  $Q_x$  is the weighting of the state reference tracking deviations,  $R_u$  is the weighting of the input magnitudes, and  $Q_y$  is the weighting of the output reference tracking deviations.  $R_{\Delta u}$ ,  $Q_x$ ,  $R_u$ , and  $Q_y$  are diagonal matrices with the respective weights along the diagonal.

### 3.4. Implementation

The cabin cooling effort of the A/C system is controlled using three actuators: the compressor clutch,  $\pi_c$ , condenser fan,  $\hat{N}$ , and the AGS,  $X$ . The compressor clutch connects the compressor to the engine, enabling the compressor to operate and increase the pressure of the refrigerant leaving through the discharge port. As mentioned in Section 2.3, the compressor clutch  $\pi_c$  is not a continuous variable in practice and is a discrete binary variable. In this study,  $\pi_c$  is initially treated as a continuous variable to be computed via QP optimization. When the MPC computes the continuous compressor clutch signal  $\pi_c$ , it rounds it to a value of either 0 or 1 prior to sending it to the GT-Suite model to ensure it is in the discrete form that GT-Suite expects. Alternative methods exist that are better suited for dealing with mixed-variable systems like the one described. For instance, techniques such as hybrid predictive control models that incorporate both integer and continuous variables have been explored in the literature [35–38] and can be further investigated in future studies. These methods provide a more holistic approach to dealing with mixed-variable systems, allowing for more accurate and efficient control strategies in scenarios where variables exhibit both continuous and discrete behaviors.

The condenser fan and AGS regulate the flow of the ambient air stream entering the vehicle’s front end, which passes through the condenser to exchange heat with the refrigerant. The fan’s rotational speed is continuously controllable, and the AGS area open-fraction can continuously vary between fully closed and fully open. Note that some underhood airflow can still occur when fully closed due to the vehicle’s motion and/or fan operation.

Based on the approach detailed in Section 2.2, using a time step of one second leaves only the  $A$  matrix changed due to the addition of the identity matrix, as shown in (3)–(5). By convention, the time step for MPC–plant interaction is the same as the one used within the MPC for future time predictions using the model up to the prediction horizon [32]. The time step of one second was selected after assessing the open-loop response of the air temperature output subjected to a unit step from each of the three controlled inputs for the continuous-time linear model. All other parameters, such as the external inputs, were held

constant at the trim state selected based on an operating point observed for the GT-Suite (real system) model.

### 3.4.1. States, Inputs, and Outputs

For the simulations in this study, from the A/C system equations in (6)–(9) the state vector  $x$  is defined as:

$$x = \begin{bmatrix} p_c \\ p_e \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (27)$$

where  $p_c$  and  $p_e$  are the condenser and evaporator refrigerant pressures. The input vector  $u$  is defined as:

$$u = \begin{bmatrix} \pi_c \\ \hat{N} \\ X \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (28)$$

where  $\pi_c$  is a continuous compressor clutch signal,  $\hat{N}$  is the normalized rotational speed of the condenser fan between zero and one, and  $X$  is the open fraction of the AGS between zero and one.

The output vector,  $y$ , is defined as:

$$y = \begin{bmatrix} T_{air,out,e} \\ \dot{W}_{total} \end{bmatrix} = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} \quad (29)$$

where  $T_{air,out,e}$  is the temperature of the air leaving the evaporator (entering the cabin) [K], and  $\dot{W}_{total}$  is the total power consumption of the compressor [Watt] or [J/s] that is comprised of  $\dot{W}_{comp}$ ,  $\dot{W}_{fan}$ , and  $\Delta\dot{W}_{drag}$ , as discussed in (9). The evaporator outlet air temperature is chosen as one of the outputs for the model because the controller's goal is to control the actuators to maintain this air temperature close to a reference or set value, thus regulating the cabin cooling performance. The other output,  $\dot{W}_{total}$ , is chosen because the second goal of this controller is to minimize the power consumption of the A/C system while satisfying the cooling objective.

It is important to note that, in the proposed formulation, feedback is taken into account by using the direct system outputs as the inputs for the controller. This means that no estimator or observer was used to obtain the state variables or the output variables of the system. In this study, it is assumed that the system outputs are measurable and noise-free, and that the system model is accurate. Therefore, a possible model mismatch is not considered.

However, in reality, these assumptions may not hold true. There may be measurement errors, nonlinearities, uncertainties, or disturbances that affect the system's performance and stability. In such cases, using an observer or an estimator could help to improve the feedback quality and robustness of the controller. This limitation can be further explored in future studies with the implementation of an observer or an estimator to handle potential system mismatches.

### 3.4.2. Sampling Time, Prediction Horizon, and Control Horizon

As per the recommendations and discussion provided in Section 2.1.2, the time unit of measurement for the governing equations used to derive the linearized model is in seconds. Thus, the units of the discretization time step are in seconds to maintain consistency. The rise time of the A/C system in this study is roughly 12.7 s, meaning a time step between 0.64 s to 1.3 s would fit 10 to 20 samples within the open-loop rise time, meaning a sampling or discretization time step of  $T = 1$  s is satisfactory based on this best practice.

As per the recommendation for the choice of prediction horizon, in this study, a prediction horizon of 10 s or 10 time steps is chosen. This reduces the computational burden for the optimization process required during the prediction horizon by choosing the lower bound of the recommended range. However, in future studies, one can further

optimize the prediction horizon. Additionally, a control horizon of two steps is chosen for the implementation of the adaptive MPC as it provides a sufficient balance between computation, efficiency, and performance for the MPC design.

It should be noted that the choices above were based on trial and error in the study to obtain satisfactory results for the industrial partner in this first phase of investigations for the implementation of an MPC controller for an A/C system in a vehicle. In future studies, additional investigations shall be pursued to further improve and optimize these choices for the controller design.

### 3.4.3. Constraints

As discussed in Section 3.3, constraints can be set on the optimization. The values of these three controlled inputs have physical limitations based on their range of operation. These are hard constraints that cannot be exceeded by any means. The mathematical number sets and limitations of these three controlled variables are provided by the industrial partner as:

$$\begin{aligned} \pi_c \in \mathbb{N} \text{ s.t.} \quad \pi_c = [0, 1] &:= \left[ l_{b_{u_1}}, u_{b_{u_1}} \right] \\ \hat{N} \in \mathbb{Q} \text{ s.t.} \quad 0 \leq \hat{N} \leq 1 &:= \left[ l_{b_{u_2}}, u_{b_{u_2}} \right] \\ X \in \mathbb{Q} \text{ s.t.} \quad 0 \leq X \leq 1 &:= \left[ l_{b_{u_3}}, u_{b_{u_3}} \right] \end{aligned} \quad (30)$$

The states also have constraints but they are not due to physical limitations. The constraints on the states are imposed to prevent problems arising in the HEXs due to excessive or lack of pressure. The state constraints are as follows and are defined by the industrial partner:

$$\begin{aligned} 100 \text{ kPa} \leq p_e \leq 1300 \text{ kPa} &:= \left[ l_{b_{x_1}}, u_{b_{x_1}} \right] \\ 700 \text{ kPa} \leq p_c \leq 3000 \text{ kPa} &:= \left[ l_{b_{x_2}}, u_{b_{x_2}} \right] \end{aligned} \quad (31)$$

For the QP algorithm to run successfully, constraints must also be present for the outputs as well but, since the outputs do not have bounds for the application in this study, they are set to infinity as

$$\begin{aligned} -\infty K \leq T_{air,out,e} \leq +\infty K &:= \left[ l_{b_{y_1}}, u_{b_{y_1}} \right] \\ -\infty \text{ Watt} \leq \dot{W}_{total} \leq +\infty \text{ Watt} &:= \left[ l_{b_{y_2}}, u_{b_{y_2}} \right] \end{aligned} \quad (32)$$

Feasibility and infeasibility issues in MPC controllers with QP optimization can affect the stability and performance of the closed-loop system. If the QP problem has hard constraints, such as input or output limits, it may be impossible to satisfy them at some time steps, leading to infeasibility. If the QP solver detects infeasibility, it terminates immediately and returns an error in MATLAB. To avoid this situation, some methods have been proposed, such as using soft constraints, additional constraint checking horizon, terminal condition and cost, or disturbance preview [38].

In this study, the infeasibility problem shall not occur as the only hard constraints in the QP optimization in this study are on the inputs outlined in (30) and these constraints shall be met as, in any A/C system of a vehicle, there must be a combination of these inputs that provides some sort of long-term asymptotic stability and tracking for the cabin temperature even without the use of an MPC controller. This was primarily achieved by the baseline PID controlled in previous works and can be the bare minimum solution to this optimization problem that would exist at all times. Therefore, since there is no time limit on the settling time for the temperature tracking of the cabin, the control inputs and hard constraints in (30) and (31), as shown in simulation results for the ranges of  $p_c$  and  $p_e$ , shall not prevent the optimization problem from converging to a solution at any given time step and, hence, the infeasibility issue shall not arise.

### 3.4.4. Weights

Additionally, from Section 3.3, the elements of the  $X_{qp}$  vector defined in (12) can be weighted based on importance and contribution to the objective function through the weighting matrix  $H$  described in (26). As defined in (27)–(29), for this work, there are two states ( $n_x = 2$ ), three controlled inputs ( $n_i = 3$ ), and two outputs ( $n_y = 2$ ). The specific weighting matrices inside (26) are defined as follows:

$$\begin{aligned} Q_x &= \begin{bmatrix} K_{p_c} & 0 \\ 0 & K_{p_e} \end{bmatrix} \\ Q_y &= \begin{bmatrix} K_{T_{air}} & 0 \\ 0 & K_{\dot{W}_T} \end{bmatrix} \\ R_u &= \begin{bmatrix} K_{\pi_c} & 0 & 0 \\ 0 & K_{\hat{N}} & 0 \\ 0 & 0 & K_X \end{bmatrix} \\ R_{\Delta u} &= \begin{bmatrix} K_{\Delta\pi_c} & 0 & 0 \\ 0 & K_{\Delta\hat{N}} & 0 \\ 0 & 0 & K_{\Delta X} \end{bmatrix} \end{aligned} \quad (33)$$

where  $K_{p_c}$  and  $K_{p_e}$  are the weightings for the state deviations,  $K_{T_{air}}$  is the weighting for the air temperature tracking deviation,  $K_{\dot{W}_T}$  is the weighting for the total power consumption,  $K_{\pi_c}$ ,  $K_{\hat{N}}$ , and  $K_X$  are the weightings for the magnitude of the controlled inputs, and  $K_{\Delta\pi_c}$ ,  $K_{\Delta\hat{N}}$ , and  $K_{\Delta X}$  are the weightings for the rate of change of the controlled inputs. It is worth noting that the goal of the MPC is to reduce energy consumption, but the power consumption is modeled and weighted because energy is the time integral of power. Thus, reducing power consumption will also reduce energy consumption.

### 3.4.5. Scenarios

Two control input cases are considered: first, only controlling the compressor clutch and, second, adding the condenser fan speed and AGS position as additional controlled inputs. Thus, the  $u_k$  and  $\Delta u_k$  elements of the  $X_{qp}$  vector for the first case with clutch-only control are:

$$u_k = [\pi_c] \Delta u_k = [\Delta\pi_c] \quad (34)$$

and for the second case with full control of compressor clutch engagement, condenser fan speed, and open-fraction AGS position are:

$$u_k = \begin{bmatrix} \pi_c \\ \hat{N} \\ X \end{bmatrix}; \Delta u_k = \begin{bmatrix} \Delta\pi_c \\ \Delta\hat{N} \\ \Delta X \end{bmatrix} \quad (35)$$

It is important to note that, as the input vector changes in cases 1 and 2, the applied constraints in the optimization problem are also adjusted to only include the applied control inputs.

### 3.4.6. Summary

The overall system implementation is shown in Figure 3, where the structure of implemented SLMC is shown with notations that are the same as in Figure 1. Additionally, the environment under which each component is developed/operated is noted in red color.

To summarize this section and formalize the control problem investigated in this study, we have the continuous SSR system form in (1) that can be written in the discrete SSR system form in (2) with  $x$  defined in (27),  $u$  defined in (28), and  $y$  defined in (29). The matrices  $A$ ,  $B$ ,  $C$ , and  $D$  in (1) are the Jacobians calculated from (6)–(9) using (10). These matrices can be converted to their discrete form using Euler's method described in (3)–(5) to obtain  $A_d$ ,  $B_d$ ,  $C_d$ , and  $D_d$  in (2) with the sampling step of  $T = 1$  s.

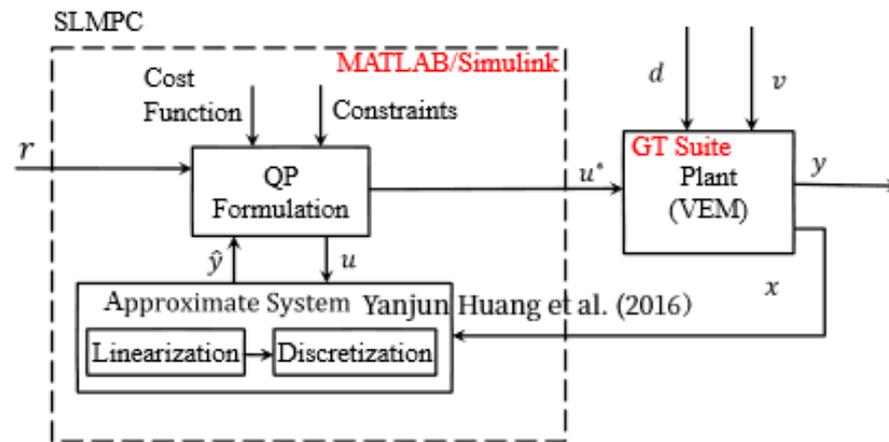


Figure 3. Successive Linearization Based MPC block diagram [22].

The goal of the optimization problem in SLMPC is to minimize the objective function in (11) in the general case and for this study in a simplified form in (13) and with constraints formed as (21), where the elements of  $X_{qp}$  and  $H$  are defined in general in (12)–(26) with specific definitions for this study in (33) subject to constraints in (24) and (25) in general form and for this study in (30)–(32).

Finally, the investigations are conducted for two scenarios, 1. clutch only as input for the MPC design as outlined in (34), and 2. full control input including compressor clutch engagement, condenser fan speed, and open-fraction AGS position as outlined in (35). The time or prediction horizon for the MPC design was set to 10 time steps, and since the time step for this study was chosen as  $T = 1$  s it would equal a 10 s look ahead in time for the controller.

The results and analysis of the simulations with the VEM model, which was provided by the industrial partner in the GT-suite environment and is used as the plant for the simulations, are presented in Section 4, including some discussions on the observations made from the outcomes.

#### 4. Results and Discussion

To evaluate the performance of the air temperature tracking, the root mean squared error (RMSE) between the actual air temperature from the GT-Suite model and the target reference temperature was calculated as:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n_{points}} (x_{actual,i} - x_{reference,i})^2}{n_{points}}} \tag{36}$$

where  $x_{actual,i}$  is the actual value from the GT-Suite model,  $x_{reference,i}$  is the target reference value, and  $n_{points}$  is the total number of data points. The choice for the performance evaluation in (36) is to ensure that the tracking of the reference value is monitored and considered throughout the entire simulation, not just for a portion of it. That is why the summation in (36) is from 1 to the total number of data points or simulation outputs.

The performance of the GT-Suite model with the baseline PID control scheme is shown in Table 1, where  $n_{\Delta\pi_c}$  is the clutch actuation frequency in the number of actuations per minute. These metrics will be referenced to the MPC results to compare the improvement of the new control scheme.

Table 1. Baseline VEM model performance metrics.

Model	$T_{eao}RMSE$ [C]	$W_T$ [kJ]	$n_{\Delta\pi_c}$
Baseline GT-Suite	6.27	$1.26 \times 10^3$	0

#### 4.1. MPC Tuning Process

The weighting parameters of the MPC controller are set for the individual elements of the objective function as  $X_{qp}$ , defined in (12). The most crucial elements of (12) to be minimized are (i) the air temperature setpoint deviation corresponding to  $K_{T_{air}}$  in (33), (ii) the instantaneous power consumption corresponding to  $K_{\dot{W}_T}$  in (33), and (iii) the rate of change of the clutch command corresponding to  $K_{\Delta\pi_c}$  in (33). The rate of change of the clutch command is critical to consider since the durability of the compressor can be a concern if a chattering phenomenon occurs, where the compressor clutch is constantly cycled between engaging and disengaging. Wear and tear on the compressor clutch can lead to reliability issues within vehicles and can increase expenses if the compressor requires maintenance more frequently. To quantify the actuation frequency of the clutch, the total number of clutch actuations throughout the cycle is recorded and divided by the 9.583 min (575 s) duration of the test cycle used in this study. With these three as the crucial weightings of interest for the MPC tuning, all other weightings in (33) are set to a value of one, while these three ( $K_{T_{air}}$ ,  $K_{\dot{W}_T}$ , and  $K_{\Delta\pi_c}$ ) are varied to ensure a reasonable contribution of various terms in the objective function.

There are optimal methods for tuning MPCs, as reported by Vazques et al. [39] using an artificial intelligence approach for real-time tuning of MPC for power converters and by Yamashita et al. [40] using a multi-objective tuning technique for MPC tuning in grinding circuits. However, in this study, a variation in the preliminary optimization method known as grid search with a systematic distribution of the grid points in the desired search space was considered as the focus of this study was more on the feasibility of implementing an MPC controller for the vehicle's A/C system rather than optimizing it. More advanced and optimal tuning techniques can be considered in future studies. A preliminary sensitivity assessment was conducted to determine the search space ranges for these three weighting values,  $K_{\Delta\pi_c}$ ,  $K_{T_{air}}$ , and  $K_{\dot{W}_T}$ , to ensure the entire relevant search space is being explored.  $K_{\Delta\pi_c}$ ,  $K_{T_{air}}$ , and  $K_{\dot{W}_T}$  were systematically increased while all other weighting factors were held constant. For example,  $K_{\Delta\pi_c}$  was increased until no further significant change was observed in the system's clutch actuation frequency.

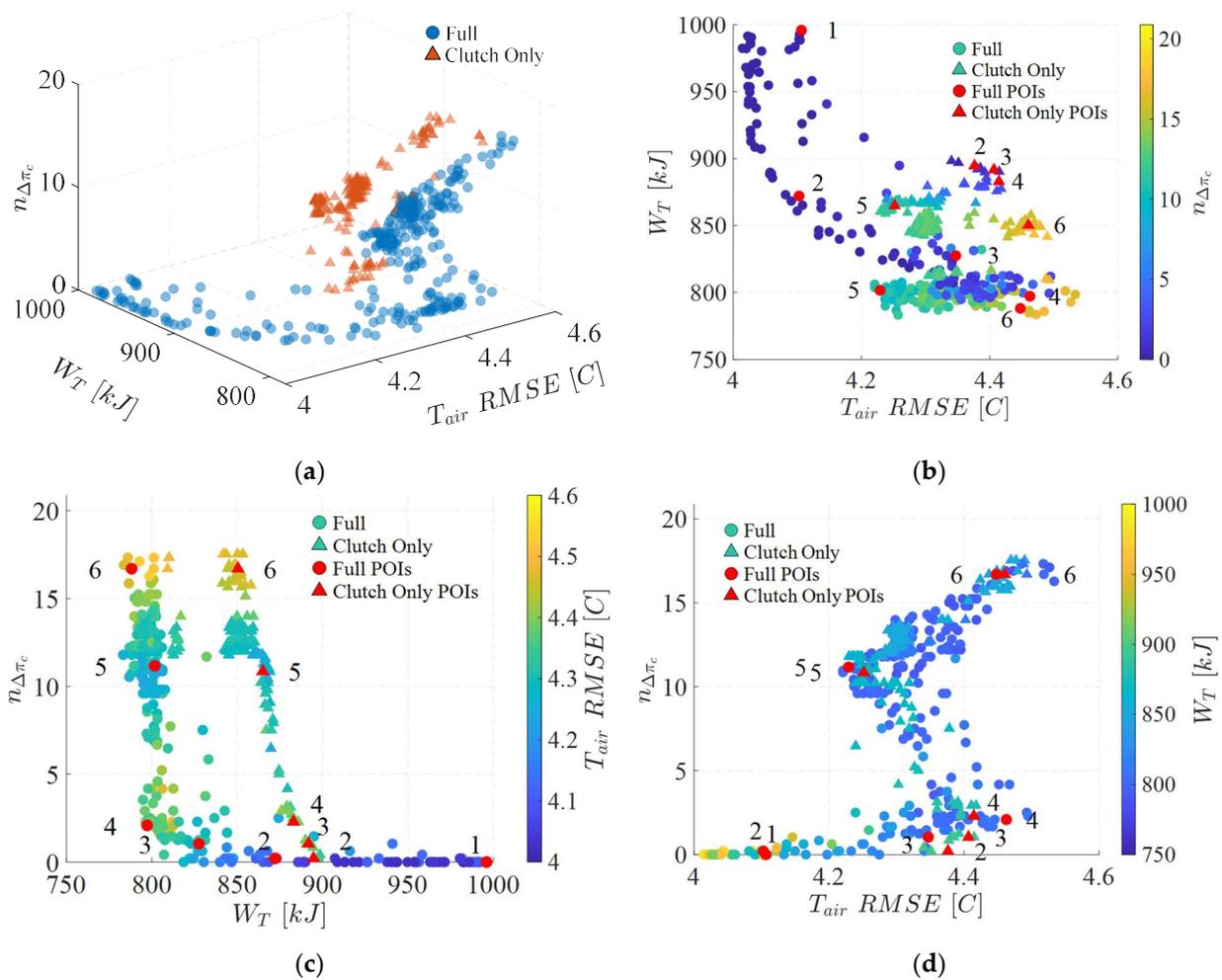
To form the optimization search space, the valid weighting ranges for  $K_{\Delta\pi_c}$  was identified as 1–1000 since values greater than 1000 presented no change to the clutch actuation frequency. The valid weighting range for  $K_{T_{air}}$  was chosen between 1–50, as the values above this range provided no significant change to the air temperature's RMSE. Finally, the valid weighting range for  $K_{\dot{W}_T}$  was chosen to be between 1 and 50 since integer values greater than 50 caused the compressor to shut off completely for the entire cycle, meaning the A/C system was inactive, and the air temperature's RMSE reached extreme and unrealistic values. Therefore, the optimization process for the MPC tuning could be described as

$$\begin{aligned} &\text{maximize (or minimize)} && f_{GS} \left( K_{\Delta\pi_c}, K_{T_{air}}, \text{ and } K_{\dot{W}_T} \right) \\ &\text{subject to} && 1 \leq K_{\Delta\pi_c} \leq 1000 \\ &&& 1 \leq K_{T_{air}} \leq 50 \\ &&& 1 \leq K_{\dot{W}_T} \leq 50 \end{aligned} \quad (37)$$

where  $f_{GS}(\cdot)$  is the function whose value needs to be maximized or minimized during the optimization process. A Latin Hypercube Sampling (LHS) approach outlined by Huntington and Lyrantzis [41] was used to determine the grid points (candidate solutions distribution) in the search space for the three MPC weightings within the defined search spaces in (37). A sample space of 300 unique combinations of weighting values was randomly generated in MATLAB using LHS to adequately fill the search space defined in (37). The MPC was simulated with the GT-Suite model in two different scenarios discussed in Section 3.4.5 for the SC03 cycle. The plant was operated with the MPC for

each set of 300 weighting combinations to assess the system’s behavior with the new control scheme.

As stated earlier, the three main metrics of interest are (i) the RMSE for the air temperature setpoint deviation, (ii) the total energy consumption throughout the cycle, and (iii) the clutch actuation frequency, each of which could be considered as the definition of  $f_{GS}(\cdot)$  in (37). In a traditional approach to grid search optimization, one would minimize or maximize the value of  $f_{GS}(\cdot)$  in (37) as one of the metrics outlined above. However, in this study, to provide a more insightful illustration of how the samples in the search space in (37) compare to one another against any of the above metrics, the results of the evaluation for all samples in the search space in (37) against all metrics above are shown in Figure 4. To further discuss the optimum points for this optimization approach, the points of interest (POIs) are introduced next to provide trade-offs and decision-making recommendations once the outcomes of all evaluations are obtained and visualized.



**Figure 4.** Evaluation of full control and clutch-only control MPCs for SC03 cycle: (a) 3D space with all three key metrics, (b) air temperature RMSE vs. total energy side profile, (c) total energy vs. clutch actuation frequency side profile, (d) air temperature RMSE vs. clutch actuation frequency side profile.

#### 4.2. Points of Interest

First, we examine the three metrics of interest for both MPC configurations, as shown in Figure 4a. Four observations can be made here.

1. First, the upper bound of energy consumption is 1000 kJ, 21% lower than that for the baseline control scheme for the plant. Hence, all assessed MPC configurations significantly reduce A/C system energy consumption.

2. Second, in a similar way, the largest air temperature error is 4.6 °C, 27% less than that for the baseline controls. This suggests that the model being re-linearized at every operating point is key to good control performance, as the baseline controls do not include this.
3. Third, clutch actuation frequency ranges from 0 (same as baseline) to ~16 actuations per minute over the SC03 cycle, with higher clutch actuations, in general, being traded for lower energy consumption. This suggests there is no single optimal value of the QP weights but rather a trading relationship the control designer must choose within.
4. Finally, the difference between the total energy consumption using clutch-only MPC and full MPC is apparent, with clutch-only MPC yielding higher energy consumption. The impact of the inclusion of two extra actuators into the MPC framework is easier to establish by examining the same data in each of the three two-dimensional views in Figure 4b–d.

In Figure 4b, the data is projected onto the energy–temperature error plane. It is clear that including all actuators in the MPC framework lowers energy consumption for a given temperature error at a comparable or lower clutch actuation frequency. Giving the MPC the control ability over all actuators also enables higher-energy solutions with few to no clutch actuations, which is impossible with clutch-only control.

In Figure 4c, the data is projected onto the clutch actuation frequency–energy plane. The three-actuator MPC is shown to lower energy consumption by 50–100 kJ over the drive cycle at comparable numbers of clutch actuation frequency and again the higher-energy solutions with little to no clutch activation are apparent. In Figure 4d, the data is projected onto the clutch actuation frequency–temperature error plane. Here, it can be readily seen that the three-actuator MPC enables similar temperature errors and clutch actuation frequency but at reduced energy.

Some POIs are gathered in Table 2. When the clutch does not actuate in POI 1 for full control, the energy consumption is at its highest. However, the air temperature’s RMSE is at its lowest. This scenario is most similar to the baseline control since the built-in compressor displacement control is the main driving force for the refrigerant control. As the compressor is active during the entire cycle, the energy consumption is high since the compressor is the primary energy consumption for the A/C system. Comparing this POI with the baseline performance, the energy consumption is reduced by 20.9% and the air temperature’s RMSE is reduced by 34.6%. This demonstrates that, by just using MPC control of the compressor, fan, and AGS, there is a significant improvement to the reference temperature tracking and energy consumption.

**Table 2.** MPC performance comparison to baseline control’s performance.

Model	Full Control			Clutch-Only Control		
	$T_{eao}RMSE$ [C]	$W_T$ [kJ]	$n_{\Delta\pi_c}$	$T_{eao}RMSE$ [C]	$W_T$ [kJ]	$n_{\Delta\pi_c}$
Baseline GT-Suite	6.27	1262.8	0	6.27	1262.8	0
POI 1	4.10	998.6	0	–	–	–
POI 2	4.10	872.1	0.20	4.38	894.7	0.20
POI 3	4.35	827.6	1.01	4.40	891.7	1.01
POI 4	4.46	797.3	2.03	4.41	882.9	2.03
POI 5	4.22	797.0	10.7	4.25	864.9	10.7
POI 6	4.44	788.2	16.2	4.46	850.4	16.2

For the clutch-only control results, the smallest clutch actuation frequency was found to be 0.2, meaning the clutch has at least one on/off actuation over the 575 s drive cycle. This can possibly be attributed to the fact that the baseline fan and AGS control inside the GT-Suite model are poor, typically running the fan when it is not needed for additional cooling and maintaining the AGS fully open regardless of the cooling requirement, causing the clutch to be more inclined to be the primary source of cooling regulation.

An interesting impact can be observed by increasing the compressor clutch actuation frequency by a small margin, from POI 1 to POI 2 and 3, which is performed by decreasing the weighting value of  $K_{\Delta\pi_c}$  or increasing the weighting values of  $K_{T_{air}}$  and  $K_{\dot{W}_T}$ . For full control, by increasing the actuation frequency from zero to 0.2, the energy consumption is decreased by 30.9% compared to the baseline. Similarly, from an actuation frequency of zero to 1.01, the energy consumption is reduced by 34.5% compared to the baseline. However, the RMSE for the air temperature is increased from 4.10 °C to 4.35 °C. POIs 5 and 6 demonstrate that, when the compressor clutch actuation frequency is above 2.03 for full control, no significant improvement to the energy consumption is achieved. However, in this region, the best air temperature's RMSE is found between a clutch actuation frequency of roughly 9.6–11.2. Comparing the POI results between the full control and clutch-only control clearly shows that, at the same clutch actuation frequency, the control of the fan and AGS provides an additional benefit to power consumption and air temperature tracking. The largest difference between the full and clutch-only control can be seen by comparing both cases at POI 3, where the full-control MPC uses 9.7% less energy compared to the clutch-only control for the same clutch actuation frequency. The weights associated with the POIs gathered in Table 2 are shown in Table 3.

**Table 3.** Weighting parameters for MPC simulations.

Model	Full Control			Clutch-Only Control		
	$K_{T_{air}}$	$K_{\dot{W}_T}$	$K_{\Delta\pi_c}$	$K_{T_{air}}$	$K_{\dot{W}_T}$	$K_{\Delta\pi_c}$
POI 1	1.0	35.3	956.4	–	–	–
POI 2	3.8	28.7	371.5	0.5	18.1	53.4
POI 3	7.2	35.6	837.1	0.8	20.0	223.5
POI 4	5.8	4.7	624.4	6.6	36.4	929.3
POI 5	22.4	3.1	551.4	16.4	35.4	563.3
POI 6	32.0	15.6	37.7	49.5	12.4	51.8

The system behavior for POIs 2 and 4 are shown in Figures 5 and 6, respectively, where they are both compared to the baseline system behavior. By investigating the behavior of the clutch-only MPC results shown in Figures 5 and 6, it can be immediately observed that the condenser fan rotational speed and AGS open fraction are controlled in the same manner as the baseline.

The impact of the full MPC on energy consumption can be recognized by comparing the condenser fan and AGS control to the baseline. For example, in POI 2, the full MPC is able to completely shut off the condenser fan for the majority of the cycle. In order to cause a sharp increase in air temperature, the clutch is disengaged, allowing the system to heat up quickly. However, when doing so, the full MPC controller drives the rotational speed of the condenser fan to its maximum value so as to not cause a significant overshoot in air temperature. This can be seen more often in POI 4, where the clutch actuation frequency is higher. In doing so, more energy is saved by disengaging the clutch and allowing the other two actuators to work toward satisfying the air temperature reference tracking objective. In POIs 5 and 6, the clutch actuation frequency is significantly higher than in POI 4, as seen in Table 2.

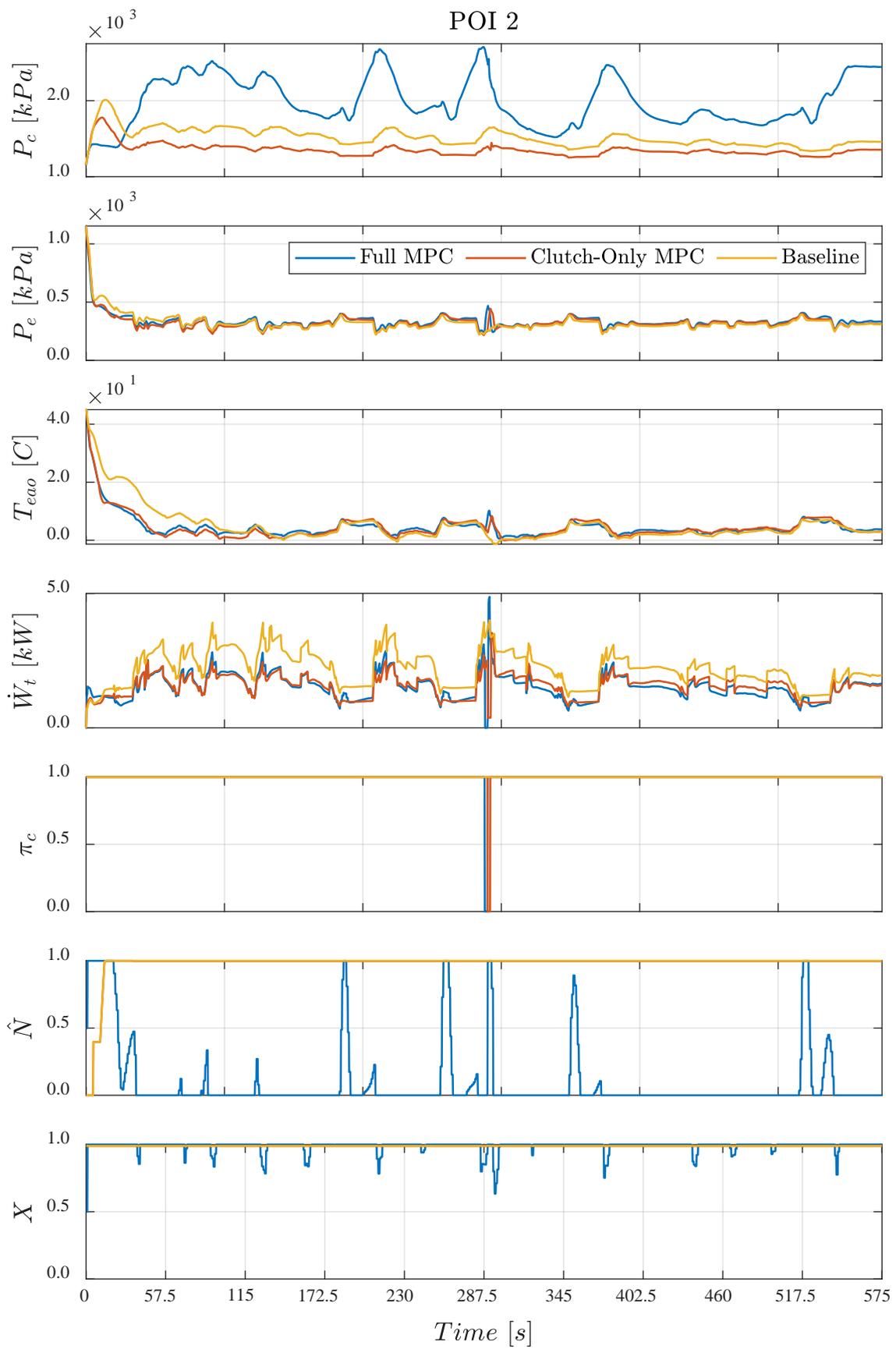
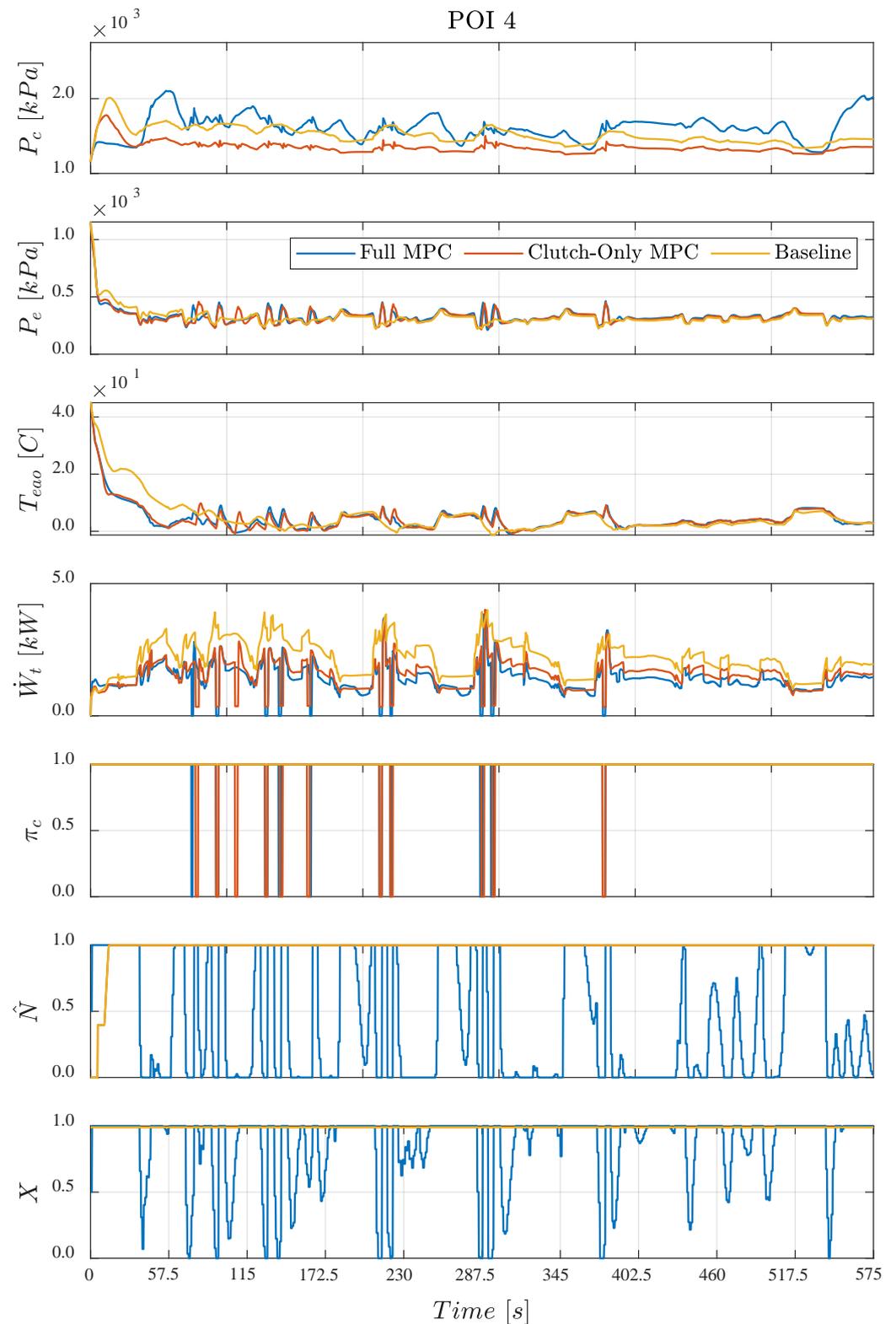


Figure 5. System behavior for baseline vs. clutch-only MPC vs. full MPC for POI 2.



**Figure 6.** System behavior for baseline vs. clutch-only MPC vs. full MPC for POI 4.

The key takeaway is that there are multiple combinations of the weighting parameters which yield good performance, trading mainly between energy consumption and clutch actuation frequency. Choosing one is the designer’s job, depending on the balance desired between energy use and clutch/compressor life longevity. For the vehicle of interest in this work, the fuel economy is 9.8 L/100 km when the A/C system is not active. In terms of

fuel economy, the A/C system can typically use roughly 1.0 L/100 km, increasing the fuel consumption to 10.8 L/100 km [42]. In this work, the greatest observed energy savings by the full MPC is 37.6%, which would reduce the fuel economy number to 10.4 L/100 km, roughly yielding a 3.5% savings in fuel economy.

## 5. Conclusions and Future Work

In this paper, two MPC schemes were applied to the A/C system of a vehicle, that is, one controlling only the compressor clutch and one additionally controlling the condenser fan and AGS. The analysis showed that, by switching to MPC control, an improvement of 32.7% to the air temperature reference tracking and 37.6% to the energy consumption can be achieved. By disengaging the compressor when it is not needed, up to 37.6% in energy savings was recorded using the full-control MPC. However, achieving a steady-state solution at the air temperature setpoint of 4 °C was not achievable by controlling the fan and AGS alone, leading to a cycling behavior of the discrete compressor clutch. By increasing the actuation frequency of the compressor clutch from zero to 2.03, the energy benefit using the full-control MPC was improved from 20.9% to 36.9%. However, increasing the clutch actuation frequency further resulted in no further energy benefit. Nevertheless, the air temperature RMSE was improved by 5.4% by increasing the clutch actuation frequency from 2.03 to 10.7, at the expense of more wear and tear on the clutch itself.

From the analysis done in this work, it can be concluded that switching to an MPC scheme is worthwhile for A/C system control, even if there is no change in the components being actuated. This can be seen by the energy saving and air temperature tracking benefit of 32.6% and 32.2% provided by the clutch-only MPC. Further benefit can be obtained, especially in terms of clutch durability, by including all three actuators in the MPC scheme. The full-control MPC is able to provide a 34.5% energy benefit with a clutch actuation frequency of 1.01, whereas the clutch-only control provides a 32.7% energy benefit with a clutch actuation frequency of 16.2. The discrete number of points investigated in the Latin Hypercube Sampling did not permit formal identification of a Pareto front. However, the trades seen between the outputs strongly suggest such a front could be identified for any given vehicle; thus, optimal weightings could be chosen based on the designer's choice between clutch actuation frequency, energy consumption, and temperature error.

**Author Contributions:** Conceptualization, T.P., J.J.D. and A.R.; methodology, T.P., J.J.D. and A.R.; software, T.P. and A.R.; validation, T.P., J.J.D. and A.R.; formal analysis, T.P. and A.R.; investigation, T.P.; resources, J.J.D. and A.R.; data curation, T.P.; writing—original draft preparation, T.P.; writing—review and editing, J.J.D. and A.R.; visualization, T.P. and A.R.; supervision, J.J.D. and A.R.; project administration, J.J.D. and A.R.; funding acquisition, J.J.D. and A.R. All authors have read and agreed to the published version of the manuscript.

**Funding:** This research was funded by Mitacs Canada and Stellantis Canada, grant number IT25381.

**Data Availability Statement:** Data are contained within the article.

**Acknowledgments:** We thank Kevin Laboe, Nandan Sawkar, and Hamed Kharrati for supporting this work by providing insight, experimental data, and guidance.

**Conflicts of Interest:** The authors declare no conflicts of interest.

## Appendix A. Successive Linearization Based MPC

Traditional MPC relies on discrete-time linear models, but most dynamic systems are nonlinear and continuous. To address this, one approach is to employ successive linearization. This involves linearizing the continuous nonlinear system at each sampling time around the current state, discretizing it, and using the resulting model to predict future values. A QP problem is then formulated to find a control sequence that minimizes a cost function. See Figure 3 for an illustration of this MPC approach based on successive linearization. The contents of this appendix are adapted from [29] for completeness of the presented work in this study.

### Appendix A.1. Continuous Nonlinear System Linearization

Consider  $n_x$ ,  $n_u$ , and  $n_o$ , as states, inputs, and outputs of a system, respectively resulting in state  $\mathbf{x} \in R^{n_x}$ , input  $\mathbf{u} \in R^{n_u}$ , and output  $\mathbf{y} \in R^{n_o}$  vectors. The outputs  $y_i$  and states  $x_i$  would be expressed as  $y_i = z_i(\mathbf{x}, \mathbf{u})$ ,  $i = 1, \dots, n_o$  and  $\dot{x}_i = f_i(\mathbf{x}, \mathbf{u})$ ,  $i = 1, \dots, n_x$ . The system dynamics in SSR form would be

$$\dot{\mathbf{x}} = F(\mathbf{x}, \mathbf{u}), \quad \mathbf{y} = Z(\mathbf{x}, \mathbf{u}) \quad (\text{A1})$$

Generally, the states and inputs in (A1) can be constrained as  $\mathbf{x}_l \leq \mathbf{x} \leq \mathbf{x}_u$  with the lower and upper bounds as  $\mathbf{x}_l, \mathbf{x}_u$ , and  $\mathbf{u}_l \leq \mathbf{u} \leq \mathbf{u}_u$  with lower and upper bounds as  $\mathbf{u}_l, \mathbf{u}_u$ , respectively. One can linearize (A1) by finding Jacobians at a set point as

$$A_c = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} & \dots & \frac{\partial f_1}{\partial x_{n_x}} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} & \dots & \frac{\partial f_2}{\partial x_{n_x}} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_{n_x}}{\partial x_1} & \frac{\partial f_{n_x}}{\partial x_2} & \dots & \frac{\partial f_{n_x}}{\partial x_{n_x}} \end{bmatrix} \quad (\text{A2})$$

which can be written in compact form as  $A_c(i, j) = \frac{\partial f_i}{\partial x_j}$ . Following the same convention, the remaining SSR matrices can be formed as  $B_c(i, j) = \frac{\partial f_i}{\partial u_j}$ ,  $C_c(i, j) = \frac{\partial z_i}{\partial x_j}$ , and  $D_c(i, j) = \frac{\partial z_i}{\partial u_j}$ . Let us denote the desired set point by  $\mathbf{x}_{op}$ , with time derivative  $\dot{\mathbf{x}}_{op}$ , and the control input  $\mathbf{u}_{op}$ . The partial derivatives in  $A_c, B_c, C_c$ , and  $D_c$  must be evaluated at  $\mathbf{x} = \mathbf{x}_{op}$ ,  $\dot{\mathbf{x}} = \dot{\mathbf{x}}_{op}$ , and  $\mathbf{u} = \mathbf{u}_{op}$ . If we denote the linearized system state by  $\mathbf{x}_L$ , control by  $\mathbf{u}_L$ , and output by  $\mathbf{y}_L$ , the system evolution can be formulated as

$$\dot{\mathbf{x}}_L = \dot{\mathbf{x}}_{op} + A_c(\mathbf{x}_L - \mathbf{x}_{op}) + B_c(\mathbf{u}_L - \mathbf{u}_{op}). \quad (\text{A3})$$

Gathering the constant terms of (A3) in a new term  $\mathcal{K} = \dot{\mathbf{x}}_{op} - A_c\mathbf{x}_{op} - B_c\mathbf{u}_{op} \in R^{n_x}$ , the SSR of the linearized system becomes

$$\dot{\mathbf{x}}_L = A_c\mathbf{x}_L + B_c\mathbf{u}_L + \mathcal{K}, \quad \mathbf{y}_L = C_c\mathbf{x}_L + D_c\mathbf{u}_L. \quad (\text{A4})$$

### Appendix A.2. Continuous Linearized System Discretization

The next step in this process is to discretize the continuous linearized system. To do so, let us multiply both sides of (A4) by  $e^{-A_c t}$

$$e^{-A_c t} \dot{\mathbf{x}}_L(t) = e^{-A_c t} A_c \mathbf{x}_L(t) + e^{-A_c t} (B_c \mathbf{u}_L(t) + \mathcal{K}) \quad (\text{A5})$$

and reorganize (A5) as

$$\frac{d}{dt} \left( e^{-A_c t} \mathbf{x}_L(t) \right) = e^{-A_c t} (B_c \mathbf{u}_L(t) + \mathcal{K}) \quad (\text{A6})$$

The analytical solution for (A6) can be obtained as

$$\mathbf{x}_L(t) = e^{A_c t} \mathbf{x}_L(0) + \int_0^t e^{A_c(t-\tau)} (B_c \mathbf{u}_L(\tau) + \mathcal{K}) d\tau \quad (\text{A7})$$

If we denote the discrete-time indices as  $\mathbf{x}_L[k] = \mathbf{x}_L(kT_s)$ , with the sampling time  $T_s$ , and considering  $t = kT_s$  in (A7), we obtain

$$\mathbf{x}_L[k+1] = e^{A_c T_s} \mathbf{x}_L[k] + \int_{kT_s}^{(k+1)T_s} e^{A_c(kT_s+T_s-\tau)} (B_c \mathbf{u}_L(\tau) + \mathcal{K}) d\tau \quad (\text{A8})$$

With the assumption that  $u_L = u_L[k]$  does not change throughout a time step and  $A_c$  is invertible, (A8) becomes

$$\mathbf{x}_L[k+1] = e^{A_c T_s} \mathbf{x}_L[k] + A_c^{-1} (e^{A_c T_s} - I) (B_c \mathbf{u}_L[k] + \mathcal{K}) \tag{A9}$$

which gives (A9) as the exact solution of discretized (A4) with the system output as

$$\mathbf{y}_L[k] = C_c \mathbf{x}_L[k] + D_c \mathbf{u}_L[k]. \tag{A10}$$

The discrete-time SSR matrices can be defined as  $A_d = e^{A_c T_s}$ ,  $B_d = A_c^{-1} (e^{A_c T_s} - I) B_c$ ,  $K = A_c^{-1} (e^{A_c T_s} - I) \mathcal{K}$ ,  $C_d = C_c$ , and  $D_d = D_c$  to write (A9) and (A10) in a compact form. Since we are in the discrete-time domain from here on, we can denote  $\mathbf{x}_L[k]$  as  $\mathbf{x}_D^k$  and  $\mathbf{u}_L[k]$  as  $\mathbf{u}_D^k$ . Hence, the discrete-time linearized SSR yields

$$\mathbf{x}_D^{k+1} = A_d \mathbf{x}_D^k + B_d \mathbf{u}_D^k + \mathbb{K}, \mathbf{y}_D^k = C_d \mathbf{x}_D^k + D_d \mathbf{u}_D^k. \tag{A11}$$

*Appendix A.3. Quadratic Programming Problem Derivation*

In MPC formulation, we compute the system’s future evolution within a finite horizon known as the prediction horizon. Predictions for a given set of control inputs are based on a discrete-time linear system model. The goal is to minimize a cost function to track a reference trajectory while adhering to constraints. At the current time step,  $k$ , the cost function typically involves the tracking error, represented by the difference between the output  $\mathbf{y}_D^k$  and the desired reference trajectory  $\mathbf{r}^k$  as  $\mathbf{e}^k = \mathbf{y}_D^k - \mathbf{r}^k$ , as well as control inputs  $\mathbf{u}_D^k$ . The error’s evolution over the prediction horizon commencing from time step  $k$  can be defined as

$$\begin{cases} \mathbf{e}^k = C_d \mathbf{x}_D^k + D_d \mathbf{u}_D^k - \mathbf{r}^k \\ \mathbf{e}^{k+1} = C_d \mathbf{x}_D^{k+1} + D_d \mathbf{u}_D^{k+1} - \mathbf{r}^{k+1} = C_d A_d \mathbf{x}_D^k + C_d B_d \mathbf{u}_D^k + C_d \mathbb{K} + D_d \mathbf{u}_D^{k+1} - \mathbf{r}^{k+1} \\ \mathbf{e}^{k+2} = C_d \mathbf{x}_D^{k+2} + D_d \mathbf{u}_D^{k+2} - \mathbf{r}^{k+2} = C_d A_d^2 \mathbf{x}_D^k + C_d A_d B_d \mathbf{u}_D^k + C_d B_d \mathbf{u}_D^{k+1} + C_d A_d \mathbb{K} + C_d \mathbb{K} + D_d \mathbf{u}_D^{k+2} - \mathbf{r}^{k+2} \\ \vdots \end{cases} \tag{A12}$$

The error and control signals over the horizon, spanning  $n_p$  time steps, can be denoted as  $\vec{\mathbf{e}} \in \mathbb{R}^{n_p \cdot n_o}$  and  $\vec{\mathbf{u}} \in \mathbb{R}^{n_p \cdot n_u}$ , respectively. Hence, once can write the system’s behavior over the horizon as

$$\begin{aligned} \underbrace{\begin{bmatrix} \mathbf{e}^k \\ \mathbf{e}^{k+1} \\ \mathbf{e}^{k+2} \\ \vdots \\ \mathbf{e}^{k+n_p-1} \end{bmatrix}}_{\vec{\mathbf{e}}_k} &= \underbrace{\begin{bmatrix} C_d \\ C_d A_d \\ C_d A_d^2 \\ \vdots \\ C_d A_d^{n_p-1} \end{bmatrix}}_{P \in \mathbb{R}^{n_p n_o \times n_x}} \mathbf{x}_D^k + \\ &\underbrace{\begin{bmatrix} D_d & 0 & 0 & \dots \\ C_d B_d & D_d & 0 & \dots \\ C_d A_d B_d & C_d B_d & D_d & \dots \\ \vdots & \vdots & \vdots & \ddots \\ C_d A_d^{n_p-2} B_d & C_d A_d^{n_p-3} B_d & C_d A_d^{n_p-4} B_d & \dots \end{bmatrix}}_{H \in \mathbb{R}^{n_p n_o \times n_p n_u}} \vec{\mathbf{u}}_k - \underbrace{\begin{bmatrix} \mathbf{r}^k \\ \mathbf{r}^{k+1} \\ \mathbf{r}^{k+2} \\ \vdots \\ \mathbf{r}^{k+n_p-1} \end{bmatrix}}_{\vec{\mathbf{r}}_k} \end{aligned} \tag{A13}$$

where  $P, H,$  and  $E$  denote the output prediction matrices, while  $\mathbb{K}$  represents the constant term from (A11). The vectors  $\vec{e}_k, \vec{u}_k,$  and  $\vec{r}_k$  encompass the combined error, control input, and reference vectors for the horizon commencing at time  $k$ . The concise expression for (A13) would then be as follows

$$\vec{e}_k = Px_D^k + H\vec{u}_k + K \tag{A14}$$

where  $K = E\mathbb{K} - \vec{r}_k \in R^{n_p \cdot n_o}$  collects all constant terms.

The cost function, which quantifies the controller’s performance, is defined as

$$J = \frac{1}{2} \left( \vec{e}_k^T Q \vec{e}_k + \vec{u}_k^T R \vec{u}_k \right) \tag{A15}$$

where  $Q \in \mathbb{R}^{n_p n_o \times n_p n_o}$  and  $R \in \mathbb{R}^{n_p n_u \times n_p n_u}$  are weight matrices for error and input, respectively. Substituting (A14) into (A15) yields

$$J = \frac{1}{2} \left( (Px_D^k + H\vec{u}_k + K)^T Q (Px_D^k + H\vec{u}_k + K) + \vec{u}_k^T T \vec{u}_k \right)$$

After eliminating terms unrelated to the control input  $\vec{u}_k,$  and combining the quadratic and linear components, we arrive at

$$J = \frac{1}{2} \vec{u}_k^T \underbrace{(H^T Q H + R)}_G \vec{u}_k + \underbrace{(x_D^{kT} P^T + K^T)}_{W^T} Q H \vec{u}_k \tag{A16}$$

Utilizing matrices  $G$  and  $W,$  the objective function  $J$  becomes

$$J = \frac{1}{2} \vec{u}_k^T G \vec{u}_k + W^T \vec{u}_k \tag{A17}$$

Nevertheless, when control inputs carry a non-zero weight in the cost function, a steady-state output error may arise. In such cases, it might be more cost-effective to allow this error in  $J$  instead of raising the control inputs. A solution is to define the cost function in terms of control input increments rather than their absolute values. This approach minimizes the impact of control input values on  $J$ . This requires transforming the control vector elements as  $u_D^k = u_D^{k-1} + \delta u_1, u_D^{k+1} = u_D^k + \delta u_2 = u_D^{k-1} + \delta u_1 + \delta u_2, \dots, u_D^{k+n_p-1} = u_D^{k+n_p-2} + \delta u_{n_p} = u_D^{k-1} + \delta u_1 + \dots + \delta u_{n_p}$  or, concisely, as

$$\underbrace{\begin{bmatrix} u_D^k \\ u_D^{k+1} \\ \vdots \\ u_D^{k+n_p-1} \end{bmatrix}}_{\vec{u}_k} = \underbrace{\begin{bmatrix} u_D^{k-1} \\ u_D^{k-1} \\ \vdots \\ u_D^{k-1} \end{bmatrix}}_{\vec{u}_{k-1}} + \underbrace{\begin{bmatrix} I & 0 & \dots \\ I & I & \dots \\ \vdots & \vdots & \ddots \\ I & I & \dots \end{bmatrix}}_{\Delta \in \mathbb{R}^{n_p n_u \times n_p n_u}} \underbrace{\begin{bmatrix} \delta u_1 \\ \delta u_2 \\ \vdots \\ \delta u_{n_p} \end{bmatrix}}_{\vec{\delta u}_k}$$

where  $\Delta$  is a lower diagonal matrix with identity  $I \in R^{n_u \times n_u}$  or a zero  $0 \in R^{n_u \times n_u}$  matrix of the same size as its elements. Hence, (A14) can be rewritten as

$$\vec{e}_k = Px_D^k + H \left( \vec{u}_{k-1} + \Delta \vec{\delta u}_k \right) + K. \tag{A18}$$

Since  $\vec{u}_{k-1}$  is constant throughout the horizon, it can be included in  $K$ . The objective function  $J_\Delta$  in  $\delta u$  is formulated as

$$J_\Delta = \frac{1}{2} \vec{\delta u}_k^T G_\Delta \vec{\delta u}_k + W_\Delta^T \vec{\delta u}_k \tag{A19}$$

where  $G_\Delta = \Delta^T G$  and  $W_\Delta^T = \vec{u}_{k-1}^T G \Delta + W^T \Delta$ .

A key aspect of MPC is its ability to handle prescribed constraints on both state variables and control inputs. To enforce constraints on the states, it is essential to establish the state vector's trajectory over the prediction horizon.

$$\begin{aligned}
 \begin{bmatrix} \mathbf{x}_D^k \\ \mathbf{x}_D^{k+1} \\ \mathbf{x}_D^{k+2} \\ \vdots \\ \mathbf{x}_D^{k+n_p} \end{bmatrix} &= \begin{bmatrix} I \\ A_d \\ A_d^2 \\ \vdots \\ A_d^{n_p} \end{bmatrix} \mathbf{x}_D^k + \\
 \underbrace{\begin{bmatrix} 0 \\ B_d \\ A_d B_d \\ \vdots \\ A_d^{n_p-1} B_d \end{bmatrix}}_{P_x \in \mathbb{R}^{(n_p+1)n_x \times n_x}} &+ \underbrace{\begin{bmatrix} 0 & 0 & 0 & \cdots \\ 0 & 0 & 0 & \cdots \\ B_d & 0 & 0 & \cdots \\ \vdots & \vdots & \vdots & \ddots \\ A_d^{n_p-2} B_d & A_d^{n_p-3} B_d & \cdots & \cdots \end{bmatrix}}_{H_x \in \mathbb{R}^{(n_p+1)n_x \times n_p n_u}} \cdot \\
 \begin{bmatrix} \mathbf{u}_D^k \\ \mathbf{u}_D^{k+1} \\ \mathbf{u}_D^{k+2} \\ \vdots \\ \mathbf{u}_D^{k+n_p-1} \end{bmatrix} &+ \underbrace{\begin{bmatrix} 0 \\ I \\ I + A_d \\ \vdots \\ I + \sum_{i=1}^{n_p-1} A_d^i \end{bmatrix}}_{E_x \in \mathbb{R}^{(n_p+1)n_x \times n_x}} \mathbb{K}
 \end{aligned} \tag{A20}$$

where  $\vec{x}_k$  holds  $n_p + 1$  state vectors from the time horizon, and  $P_x$ ,  $H_x$ , and  $E_x$  are state prediction matrices. Hence, (A20) becomes

$$\vec{x}_k = P_x \mathbf{x}_D^k + H_x \vec{u}_k + E_x \mathbb{K} \tag{A21}$$

The state constraint  $\mathbf{x}_l \leq V \vec{x}_k \leq \mathbf{x}_u$  can be written as  $\mathbf{x}_l - V P_x \mathbf{x}_D^k - V E_x \mathbb{K} \leq V H_x \vec{u}_k \leq \mathbf{x}_u - V P_x \mathbf{x}_D^k - V E_x \mathbb{K}$  where  $V \in \mathbb{R}^{(n_p+1)n_c \times (n_p+1)n_x}$  picks  $n_c$  states with active constraints. If all states are constrained, then  $V$  is an identity matrix. With  $\vec{\delta u}_k$  and defining  $U = V H_x$ , the QP problem can be characterized as

$$\begin{aligned}
 \underset{\vec{\delta u}_k}{\operatorname{argmin}} J &= \frac{1}{2} \vec{\delta u}_k^T G_\Delta \vec{\delta u}_k + W_\Delta^T \vec{\delta u}_k \\
 \text{subject to } U_l &\leq U \vec{\delta u}_k \leq U_u \\
 u_l &\leq \vec{\delta u}_k \leq u_u
 \end{aligned} \tag{A22}$$

where  $U_l = \mathbf{x}_l - U^k$  and  $U_u = \mathbf{x}_u - U^k$ ,  $U^k = V P_x \mathbf{x}_D^k + V E_x \mathbb{K} + V H_x \vec{u}_{k-1}$ , while  $u_l = \Delta^{-1}(\mathbf{x}_l - \vec{u}_{k-1})$  and  $u_u = \Delta^{-1}(\mathbf{x}_u - \vec{u}_{k-1})$ . It is important to highlight that the constraints and bounds in (A22) can be either constant or time-varying.

In a typical QP solver, matrices  $G$  and  $W$ , in conjunction with constraint and bound information, are employed to determine an optimal solution. In closed-loop MPC, only the first control vector within the horizon is implemented on the system, while the subsequent control inputs are disregarded. Following the current time step, new system measurements are acquired. This iterative process, involving linearization, discretization, prediction

matrix generation, and the application of the first control input, is repeated at each time step in a closed-loop manner.

## Appendix B. Relaxation of MPC Formulation

The concept of using continuous variables to represent discrete ones is rooted in optimization and relaxation methods. In this approach, discrete variables are temporarily considered as continuous, simplifying the problem into a relaxation of the original. Solving this relaxed problem is more manageable. However, rounding the solution to the nearest integers does not always guarantee an optimal solution for the original problem because the relaxation can overestimate feasible solutions. Consequently, additional techniques like branch-and-bound algorithms are commonly employed to systematically explore and find optimal integer solutions.

In mathematical optimization and related fields, relaxation serves as a modeling strategy to approximate complex problems with more tractable ones. It allows for insights into the original problem. For example, in integer programming, linear programming relaxations enable rational solutions without integer constraints. In combinatorial optimization, Lagrangian relaxations address complex constraints, simplifying problem solutions. These relaxation techniques often complement branch-and-bound algorithms by helping to establish bounds.

Axehill et al.'s work [43] addresses the control of hybrid systems in Mixed Logical Dynamical (MLD) form, particularly those combining continuous and binary control signals. While binary states are not explicitly considered, this problem shares similarities with the finite alphabet control problem.

In standard linear MPC, the control problem is a QP task. In the hybrid context, binary variables are introduced, making it a Mixed Integer Quadratic Programming (MIQP) problem, often called Mixed Integer Predictive Control (MIPC). This shift from QP to MIQP transforms it into a computationally challenging non-convex problem. The contents of this appendix are adapted from [43] for completeness of the presented work in this study.

### Appendix B.1. Notation

In this section,  $\mathbb{S}^n$  represents symmetric matrices with  $n$  columns and  $\mathbb{S}_{++}^n$  ( $\mathbb{S}_+^n$ ) indicates symmetric positive (semi) definite matrices with  $n$  columns. The superscript  $*$  denotes values of variables and functions at their optimum. The function  $\text{diag}(\cdot)$  is defined such that, when applied to a vector, it forms a diagonal matrix with the vector's elements along the diagonal. When applied to a matrix, it creates a vector consisting of the matrix's diagonal elements. We frequently use the sets  $\mathcal{T} = \{0, \dots, N-1\}$  and  $\mathcal{I} = \{1, \dots, Nm\}$ . Additional notation details can be found in [43].

### Appendix B.2. Control Problem

For a system in the following form

$$\begin{cases} x(0) = x_0 \\ x(t+1) = Ax(t) + Bu(t), \forall t \in T \end{cases} \quad (\text{A23})$$

where  $A \in \mathbb{R}^{n \times n}$ ,  $B \in \mathbb{R}^{n \times m}$ ,  $x(t) \in \mathbb{R}^n$  is the state, and  $x_0 \in \mathbb{R}^n$  is the initial state. We consider an MIPC problem over a prediction horizon of length  $N$ . We use an all-binary control input  $u(t) \in \{0, 1\}^m$  to keep derivations as simple as possible. It is worth noting that the results remain applicable even when combining real-valued and binary control signals. The objective function to minimize is given by:

$$\begin{aligned} J_{\text{MPC}}(x(\cdot), u(\cdot)) \\ = \frac{1}{2} \sum_{t=0}^{N-1} \left( \|x(t)\|_{Q_x}^2 + \|u(t)\|_{Q_u}^2 \right) + \frac{1}{2} \|x(N)\|_{Q_x}^2 \end{aligned} \quad (\text{A24})$$

where  $\|v\|_Q^2 = v^T Q v$ ,  $Q_x \in \mathbb{S}_+^n$  and where  $Q_u \in \mathbb{S}_{++}^m$ . At each time step, the system is additionally subjected to  $c$  linear inequality constraints represented as:

$$\begin{aligned} H_x x(t) + H_u u(t) + h &\leq 0, \forall t \in \mathcal{T} \\ H_x x(N) + h &\leq 0 \end{aligned} \quad (\text{A25})$$

where  $H_x \in \mathbb{R}^{c \times n}$ ,  $H_u \in \mathbb{R}^{c \times m}$ , and  $h \in \mathbb{R}^c$ . The MIPCC problem can be expressed as an MIQP problem in two equivalent forms. In the first form, the equality constraints representing the system's dynamics are preserved, resulting in an MIQP problem given as:

$$\begin{aligned} \underset{x,u}{\text{minimize}} \quad & J_{\text{MIQP}_1}(x,u) \\ \text{s.t.} \quad & [\text{AB}][x^T u^T]^T = b, \\ & [H_x H_u][x^T u^T]^T + h \leq 0 \\ & u_i \in \{0,1\}, \forall i \in \mathcal{I} \end{aligned} \quad (\text{A26})$$

where  $J_{\text{MIQP}_1}(x,u) = \frac{1}{2}x^T Q_x x + \frac{1}{2}u^T Q_u u$  for a suitable choice of  $A, B, H_x, H_u, h, b$ , and matrices  $Q_x$  and  $Q_u$ . The  $x \in \mathbb{R}^{(N+1)n}$  and  $u \in \mathbb{R}^{Nm}$  denote system states and control inputs, respectively. In the second form, the equality constraints in (A23) can be employed to remove the states as  $x = S_x x_0 + S_u u$ . This results, with a suitable  $S_x$  and  $S_u$ , in an MIQP problem similar to (A26) in a new form as

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & J_{\text{MIQP}_2}(u) \\ \text{s.t.} \quad & (H_x S_u + H_u)u + h + H_x S_x x_0 \leq 0, \\ & u_i \in \{0,1\}, \forall i \in \mathcal{I} \end{aligned} \quad (\text{A27})$$

where  $J_{\text{MIQP}_2}(u) = \frac{1}{2}u^T (S_u^T Q_x S_u + Q_u)u + (S_x x_0)^T Q_x S_u u + \kappa$  and  $\kappa = \frac{1}{2}(S_x x_0)^T Q_x (S_x x_0)$  is a constant. The optimal objective function values for the problems in (A26) and (A27) are equal, i.e.,  $J_{\text{MPC}}^* = J_{\text{MIQP}_1}^* = J_{\text{MIQP}_2}^*$ .

### Appendix B.3. Relaxations

A simple method to relax the integer constraints substituting  $u_i \in \{0,1\}$  with the relaxed constraint  $u_i \in [0,1]$ . This results in a QP problem, known as a QP relaxation. Following this approach, the QP relaxation for the problem in (A26) becomes:

$$\begin{aligned} \underset{x,u}{\text{minimize}} \quad & J_{\text{QP}_1}(x,u) \\ \text{s.t.} \quad & [\text{AB}][x^T u^T]^T = b, \\ & [H_x H_u][x^T u^T]^T + h \leq 0 \\ & 0 \leq u_i \leq 1, \forall i \in \mathcal{I} \end{aligned} \quad (\text{A28})$$

where  $J_{\text{QP}_1}(x,u) = J_{\text{MIQP}_1}(x,u)$ . Hence, the QP relaxation of (A27) yields

$$\begin{aligned} \underset{u}{\text{minimize}} \quad & J_{\text{QP}_2}(u) \\ \text{s.t.} \quad & (H_x S_u + H_u)u + h + H_x S_x x_0 \leq 0 \\ & 0 \leq u_i \leq 1, \forall i \in \mathcal{I} \end{aligned} \quad (\text{A29})$$

where  $J_{\text{QP}_2}(u) = J_{\text{MIQP}_2}(u)$ .

This relaxation is a semidefinite programming (SDP) problem and cannot generally form a QP. The SDP relaxation of (A26) yields

$$\begin{aligned}
 & \underset{U, x, u}{\text{minimize}} && J_{\text{SDP}_1}(U, x, u) \\
 & \text{s.t.} && [\text{AB}] [x^T u^T]^T = b, \\
 & && [\text{H}_x \text{H}_u] [x^T u^T]^T + h \leq 0 \\
 & && U_{ii} = u_i, \forall i \in \mathcal{I} \\
 & && U \succeq uu^T
 \end{aligned} \tag{A30}$$

where  $J_{\text{SDP}_1}(U, x, u) = \frac{1}{2}x^T Q_x x + \frac{1}{2}\text{tr}(Q_u U)$ ,  $U \in \mathbb{S}^{Nm}$  and where  $U_{ii}$  denotes matrix  $U$ 's diagonal element  $i$ . The relaxation employed in (A30) is referred to as the equality-constrained SDP relaxation, which can equivalently be rewritten as

$$\begin{aligned}
 & \underset{U(\cdot), x(\cdot), u(\cdot)}{\text{minimize}} && J_{\text{SDP}_1'}(U(\cdot), x(\cdot), u(\cdot)) \\
 & \text{s.t.} && \text{(A23) and (A25)} \\
 & && U_{ii}(t) = u_i(t), \forall t \in \mathcal{T}, \forall i \in \{1, \dots, m\} \\
 & && U(t) \succeq u(t)u(t)^T, \forall t \in \mathcal{T}
 \end{aligned} \tag{A31}$$

with

$$\begin{aligned}
 & J_{\text{SDP}_1'}(U(\cdot), x(\cdot), u(\cdot)) \\
 & = \frac{1}{2} \sum_{t=0}^{N-1} (x(t)^T Q_x x(t) + \text{tr}(Q_u U(t))) \\
 & + \frac{1}{2} x(N)^T Q_x x(N)
 \end{aligned} \tag{A32}$$

where  $U \in \mathbb{S}^{Nm}$  in (A30) is substituted by  $N U(t) \in \mathbb{S}^m$ . Therefore, the number of variables (matrix elements) in (A31) increases linearly with the prediction horizon  $N$ , in contrast to the quadratic growth in (A30).

Likewise, the SDP relaxation of (A27) yields

$$\begin{aligned}
 & \underset{U, u}{\text{minimize}} && J_{\text{SDP}_2}(U, u) \\
 & \text{s.t.} && (\text{H}_x S_u + \text{H}_u)u + h + \text{H}_x S_x x_0 \leq 0 \\
 & && U_{ii} = u_i, \forall i \in \mathcal{I} \\
 & && U \succeq uu^T
 \end{aligned} \tag{A33}$$

where  $J_{\text{SDP}_2}(U, u) = \frac{1}{2}\text{tr}((S_u^T Q_x S_u + Q_u)U) + (S_x x_0)^T Q_x S_u u + \kappa$ ,  $U \in \mathbb{S}^{Nm}$  and  $\kappa$  is a constant. It is important to note that the number of elements in  $U$  in (A32) increases quadratically with  $N$ . Further proof of the feasibility and quality of the relaxed solution can be found in [43].

Yang et al. [44] focused on developing real-time MPC for embedded systems, incorporating both continuous and discrete control inputs. They cast the online optimization in MPC as a MIQP and employed a branch and bound algorithm that handles multiple relaxed QPs. Due to embedded system constraints, they limited the number of relaxed problems and introduced custom heuristics to generate early feasible solutions.

For problems involving continuous variables in the relaxed states, an additional step is needed to revert the result to the feasible form of the original states, whether continuous or discrete. This process, referred to as "Rounding Heuristics", plays a crucial role. It involves determining how to round the solution, with an improperly rounded solution causing delays in the optimization process. The most naive rounding is to round obtained solution  $x(t)$  to 1  $x(t) \geq 0.5$  in the relaxed solution where the range for  $x(t)$  is  $x \in [0, 1]$  in continuous space or  $x = 0$  or 1 in discrete space. There is a need to reformulate the problem with appropriate weights if there are multiple discrete variables in the system's operation, as further discussed in [44].

## References

1. Høgh, G.; Nielsen, R. Model Based Nonlinear Control of Refrigeration Systems. Master's Thesis, Aalborg University, Aalborg Municipality, Denmark, 2008.
2. Koo, B.; Yoo, Y.; Won, S. Super-Twisting Algorithm-Based Sliding Mode Controller for a Refrigeration System. In Proceedings of the 2012 12th International Conference on Control, Automation and Systems, Jeju, Republic of Korea, 17–21 October 2012; pp. 34–38.
3. Liu, J.; Zhou, H.; Zhou, X.; Cao, Y.; Zhao, H. Automotive Air Conditioning System Control—A Survey. In Proceedings of the 2011 International Conference on Electronic & Mechanical Engineering and Information Technology, Harbin, China, 12–14 August 2011; pp. 3408–3412.
4. Cvok, I.; Škugor, B.; Deur, J. Control Trajectory Optimisation and Optimal Control of an Electric Vehicle HVAC System for Favourable Efficiency and Thermal Comfort. *Optim. Eng.* **2021**, *22*, 83–102. [[CrossRef](#)]
5. Kim, N.; Park, Y.; Son, J.E.; Shin, S.; Min, B.; Park, H.; Kang, S.; Hur, H.; Ha, M.Y.; Lee, M.C. Robust Sliding Mode Control of a Vapor Compression Cycle. *Int. J. Control Autom. Syst.* **2018**, *16*, 62–78. [[CrossRef](#)]
6. Xie, Y.; Liu, Z.; Liu, J.; Li, K.; Zhang, Y.; Wu, C.; Wang, P.; Wang, X. A Self-Learning Intelligent Passenger Vehicle Comfort Cooling System Control Strategy. *Appl. Therm. Eng.* **2020**, *166*, 114646. [[CrossRef](#)]
7. Huang, Y.; Khajepour, A.; Ding, H.; Bagheri, F.; Bahrami, M. An Energy-Saving Set-Point Optimizer with a Sliding Mode Controller for Automotive Air-Conditioning/Refrigeration Systems. *Appl. Energy* **2017**, *188*, 576–585. [[CrossRef](#)]
8. Liberati, F.; Giorgio, A.D.; Giuseppi, A.; Pietrabissa, A.; Habib, E.; Martirano, L. Joint Model Predictive Control of Electric and Heating Resources in a Smart Building. *IEEE Trans. Ind. Applicat.* **2019**, *55*, 7015–7027. [[CrossRef](#)]
9. Borreggine, S.; Monopoli, V.G.; Rizzello, G.; Naso, D.; Cupertino, F.; Consoletti, R. A Review on Model Predictive Control and Its Applications in Power Electronics. In Proceedings of the 2019 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE), Torino, Italy, 2–4 July 2019; pp. 1–6.
10. Huang, Y.; Wang, H.; Khajepour, A.; He, H.; Ji, J. Model Predictive Control Power Management Strategies for HEVs: A Review. *J. Power Sources* **2017**, *341*, 91–106. [[CrossRef](#)]
11. Wang, H.; Kolmanovsky, I.; Amini, M.R.; Sun, J. Model Predictive Climate Control of Connected and Automated Vehicles for Improved Energy Efficiency. In Proceedings of the 2018 Annual American Control Conference (ACC), Milwaukee, WI, USA, 27–29 June 2018; pp. 828–833.
12. Pereira, D.F.; Lopes, F.D.C.; Watanabe, E.H. Nonlinear Model Predictive Control for the Energy Management of Fuel Cell Hybrid Electric Vehicles in Real Time. *IEEE Trans. Ind. Electron.* **2021**, *68*, 3213–3223. [[CrossRef](#)]
13. Xie, Y.; Liu, Z.; Li, K.; Liu, J.; Zhang, Y.; Dan, D.; Wu, C.; Wang, P.; Wang, X. An Improved Intelligent Model Predictive Controller for Cooling System of Electric Vehicle. *Appl. Therm. Eng.* **2021**, *182*, 116084. [[CrossRef](#)]
14. Glos, J.; Otava, L.; Vaclavek, P. Non-Linear Model Predictive Control of Cabin Temperature and Air Quality in Fully Electric Vehicles. *IEEE Trans. Veh. Technol.* **2021**, *70*, 1216–1229. [[CrossRef](#)]
15. Kibalama, D.; Liu, Y.; Stockar, S.; Canova, M. Model Predictive Control for Automotive Climate Control Systems via Value Function Approximation. *IEEE Control Syst. Lett.* **2022**, *6*, 1820–1825. [[CrossRef](#)]
16. Göltz, S.; Sawodny, O. Design and Comparison of Model-Based Controllers for an Automotive Air Conditioning System in an Electric Vehicle. *Control Eng. Pract.* **2023**, *130*, 105376. [[CrossRef](#)]
17. El-Sharkawy, A.E.; Kamrad, J.C.; Lounsbury, T.H.; Baker, G.L.; Rahman, S.S. Evaluation of Impact of Active Grille Shutter on Vehicle Thermal Management. *SAE Int. J. Mater. Manuf.* **2011**, *4*, 1244–1254. [[CrossRef](#)]
18. Shigarkanthi, V.; Damodaran, V.; Soundararaju, D.; Kanniah, K. Application of Design of Experiments and Physics Based Approach in the Development of Aero Shutter Control Algorithm. *SAE Tech. Paper* **2011**. [[CrossRef](#)]
19. Chen, Y.; Kwak, K.H.; Kim, J.; Kim, Y.; Jung, D. Energy-efficient Cabin Climate Control of Electric Vehicles Using Linear Time-varying Model Predictive Control. *Optim. Control Appl. Methods* **2023**, *44*, 773–797. [[CrossRef](#)]
20. Coppola, A.; Lui, D.G.; Petrillo, A.; Santini, S. Eco-Driving Control Architecture for Platoons of Uncertain Heterogeneous Nonlinear Connected Autonomous Electric Vehicles. *IEEE Trans. Intell. Transport. Syst.* **2022**, *23*, 24220–24234. [[CrossRef](#)]
21. He, H.; Jia, H.; Sun, C.; Sun, F. Stochastic Model Predictive Control of Air Conditioning System for Electric Vehicles: Sensitivity Study, Comparison, and Improvement. *IEEE Trans. Ind. Inf.* **2018**, *14*, 4179–4189. [[CrossRef](#)]
22. Huang, Y.; Khajepour, A.; Bagheri, F.; Bahrami, M. Optimal Energy-Efficient Predictive Controllers in Automotive Air-Conditioning/Refrigeration Systems. *Appl. Energy* **2016**, *184*, 605–618. [[CrossRef](#)]
23. Parent, T. Developing a Compressor, Fan, and Active Grille Shutter Control Strategy for Air Conditioner Duty Cycles to Improve Overall Vehicle Power Consumption. Master's Thesis, University of Windsor, Windsor, ON, Canada, 2023.
24. Parent, T. Developing a Compressor, Fan, and Active Grille Shutter Control Strategy for Air Conditioner Duty Cycles to Improve the Overall Vehicle Power Consumption. Master's Thesis, Politecnico di Torino, Torino, Italy, 2023.
25. Zhang, Q.Z. Modeling, Energy Optimization and Control of Vapor Compression Refrigeration Systems For Automotive Applications. Ph.D. Thesis, Ohio State University, Columbus, OH, USA, 2014.
26. Dhar, A.; Bhasin, S. Indirect Adaptive MPC for Discrete-Time LTI Systems with Parametric Uncertainties. *IEEE Trans. Automat. Contr.* **2021**, *66*, 5498–5505. [[CrossRef](#)]
27. Hu, D.; Li, G.; Deng, F. Gain-Scheduled Model Predictive Control for a Commercial Vehicle Air Brake System. *Processes* **2021**, *9*, 899. [[CrossRef](#)]

28. Parent, T.; Defoe, J.J.; Rahimi, A. Nonlinear Modeling of an Automotive Air Conditioning System Considering Active Grille Shutters. *Modelling* **2023**, *4*, 70–86. [[CrossRef](#)]
29. Zhakatayev, A.; Rakhim, B.; Adiyatov, O.; Baimyshev, A.; Varol, H.A. Successive Linearization Based Model Predictive Control of Variable Stiffness Actuated Robots. In Proceedings of the 2017 IEEE International Conference on Advanced Intelligent Mechatronics (AIM), Munich, Germany, 3–7 July 2017; pp. 1774–1779.
30. Alhajeri, M.; Soroush, M. Tuning Guidelines for Model-Predictive Control. *Ind. Eng. Chem. Res.* **2020**, *59*, 4177–4191. [[CrossRef](#)]
31. Gajić, Z. *Linear Dynamic Systems and Signals*; Pearson Education: Upper Saddle River, NJ, USA, 2003; ISBN 978-0-201-61854-9.
32. Rawlings, J.B.; Mayne, D.Q. *Model Predictive Control: Theory and Design*; Nob Hill Pub: Madison, WI, USA, 2009; ISBN 978-0-9759377-0-9.
33. EPA SC03 Supplemental Federal Test Procedure (SFTP) with Air Conditioning. Available online: <https://www.epa.gov/emission-standards-reference-guide/epa-sc03-supplemental-federal-test-procedure-sftp-air> (accessed on 1 February 2021).
34. Dhar, A.; Bhasin, S. Adaptive MPC for Uncertain Discrete-Time LTI MIMO Systems with Incremental Input Constraints. *IFAC-PapersOnLine* **2018**, *51*, 329–334. [[CrossRef](#)]
35. De Prada, C.; Grossmann, I.; Sarabia, D.; Cristea, S. A Strategy for Predictive Control of a Mixed Continuous Batch Process. *J. Process Control* **2009**, *19*, 123–137. [[CrossRef](#)]
36. Karer, G.; Mušič, G.; Škrjanc, I.; Zupančič, B. Model Predictive Control of Nonlinear Hybrid Systems with Discrete Inputs Employing a Hybrid Fuzzy Model. *Nonlinear Anal. Hybrid Syst.* **2008**, *2*, 491–509. [[CrossRef](#)]
37. Rawlings, J.B.; Risbeck, M.J. Model Predictive Control with Discrete Actuators: Theory and Application. *Automatica* **2017**, *78*, 258–265. [[CrossRef](#)]
38. Takács, G.; Rohal’-Ilkiv, B. Stability and Feasibility of MPC. In *Model Predictive Vibration Control*; Springer: London, UK, 2012; pp. 253–285. ISBN 978-1-4471-2332-3.
39. Vazquez, S.; Marino, D.; Zafra, E.; Peña, M.D.V.; Rodríguez-Andina, J.J.; Franquelo, L.G.; Manic, M. An Artificial Intelligence Approach for Real-Time Tuning of Weighting Factors in FCS-MPC for Power Converters. *IEEE Trans. Ind. Electron.* **2022**, *69*, 11987–11998. [[CrossRef](#)]
40. Yamashita, A.S.; Martins, W.T.; Pinto, T.V.B.; Raffo, G.V.; Euzébio, T.A.M. Multiobjective Tuning Technique for MPC in Grinding Circuits. *IEEE Access* **2023**, *11*, 43041–43054. [[CrossRef](#)]
41. Huntington, D.E.; Lyrintzis, C.S. Improvements to and Limitations of Latin Hypercube Sampling. *Probabilistic Eng. Mech.* **1998**, *13*, 245–253. [[CrossRef](#)]
42. Mogro, A.E.; Huertas, J.I. Assessment of the Effect of Using Air Conditioning on the Vehicle’s Real Fuel Consumption. *Int. J. Interact. Des. Manuf.* **2021**, *15*, 271–285. [[CrossRef](#)]
43. Axehill, D.; Vandenberghe, L.; Hansson, A. Convex Relaxations for Mixed Integer Predictive Control. *Automatica* **2010**, *46*, 1540–1545. [[CrossRef](#)]
44. Yang, L.; Katnik, A.; Ozay, N. Quickly Finding Recursively Feasible Solutions for MPC with Discrete Variables. In Proceedings of the 2019 IEEE Conference on Control Technology and Applications (CCTA), Hong Kong, China, 19–21 August 2019; pp. 374–381.

**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.