

EINDHOVEN UNIVERSITY OF TECHNOLOGY

CONTROL SYSTEMS GROUP  
DEPARTMENT OF ELECTRICAL ENGINEERING

---

**Centralized and Distributed Identified Model Based  
Predictive Control for Museum Hermitage  
Amsterdam**

---

*Author:*  
XU CHEN

*Graduation Supervisors:*

dr. MIRCEA LAZAR  
dr.ir. JOBERT H.A. LUDLAGE

*Graduation Committee:*

CHAIR	prof.dr.	PAUL M.J.	VAN DEN HOF
VOTING MEMBER	dr.	A.A.J. (ERJEN)	LEFEBER
ADVISORY MEMBER	dr.	MIRCEA	LAZAR
ADVISORY MEMBER	dr.ir.	JOBERT H.A.	LUDLAGE



March, 2019

## Declaration concerning the TU/e Code of Scientific Conduct

I have read the TU/e Code of Scientific Conduct<sup>1</sup>.

In carrying out research, design and educational activities, I shall observe the five central values of scientific integrity, namely: trustworthiness, intellectual honesty, openness, independence and societal responsibility, as well as the norms and principles which follow from them.

Date

15-03-2019

Name

Xu Chen

ID-number

1034292

Signature

Xu Chen

*Submit the signed declaration to the student administration of your department.*

<sup>1</sup> See: <https://www.tue.nl/en/our-university/about-the-university/organization/integrity/scientific-integrity/>

The Netherlands Code of Conduct for Scientific Integrity, endorsed by 6 umbrella organizations, including the VSNU, can be found here also. More information about scientific integrity is published on the websites of TU/e and VSNU

# Centralized and Distributed Identified Model Based Predictive Control for Museum Hermitage Amsterdam

Xu Chen  
Control Systems Group  
Eindhoven University of Technology

**Abstract**—This thesis considers the design and implementation of centralized and distributed identified model based predictive control for the thermal dynamics of a museum. A grey-box model is identified to simulate temperature evolution of the museum based on the data generated by HAMBASE SIMULINK tool, and residual correlation test and goodness of fit are employed for validation purpose. The centralized MPC with integral action is designed and applied on the HAMBASE SIMULINK model, the simulation results indicate that MPC with integral action outperforms standard MPC and PI control with respect to offset rejection and energy saving. Additionally, the algorithm of multi-iteration distributed MPC is developed to reduce the computational complexity, and the convergence to the centralized MPC is observed along iterations of the distributed MPC. As one of the novel contributions, the algorithm of distributed MPC with integral action is developed to eliminate offset, and the convergence to the centralized MPC with integral action is also observed.

**Index Terms**—Thermal Model Identification, Model Predictive Control, Integral Action, Distributed Control

## I. INTRODUCTION

A proper indoor climate is highly required for some monumental buildings, not only for the comfort level of visitors, but for the preservation of many precious collections. The temperature evolution in such a building is one of the most complex and important part of the climate dynamics, and this complexity comes from the thermal interaction among different zones (rooms and outside) [1]. According to assumptions made in [2], maintaining a constant temperature will reduce the risk of deterioration of the collections. However, keeping such a steady temperature environment requires high energy costs, and the energy consumption of buildings accounts for around 41% of EU final energy consumption in 2017 [3]. Therefore, the investigation and implementation of advanced control techniques for building thermal systems are of great value.

Although there are many similar features that are

shared by the building thermal systems and other processing plants, several characteristics make the thermal control in buildings even more challenging, which are listed in the following:

- Time-varying system dynamics and disturbances;
- Large scale system;
- Strong interaction among system states.

Various standard control schemes, such as an on/off switching control, proportional–integral (PI) and proportional-integral–derivative (PID), are still extensively used in building thermal systems, because of their simplicity, but parameters tuning is time-consuming, and re-tuning is often required if the operating conditions vary [4, 5]. With development of techniques in information storage, computing and communication, an increasing attention has been attracted to model predictive control (MPC) during the last two decades. In MPC, a system model is utilized to calculate the optimal input sequences over a certain horizon, and the first element of the optimal sequence is applied to the system. The inherent advantages of MPC make it an eligible method in thermal control, as listed below [6]:

- Ability to handle constraints and uncertainties;
- Good performance with respect to disturbance rejection;
- Integration of climate forecasts;
- Integration of energy-saving schemes in the formulation of optimization problem;
- Integration of distributed control strategies, so that the computational load is significantly reduced.

MPC is utilized to control a HVAC plant in [7], and the simulation results therein show that the proposed Generalized Predictive Control has strong robustness to shifts in operating conditions. An evaluation of learning-based MPC and two-position control is provided in [8], experiments show a considerable reduction in energy consumption with the learning-based MPC scheme, and

more sophisticated estimation of climate and occupancy information would result in a further decrease. MPC is employed in a simulated variable air volume system to control the zone temperature in [9]. Compared with a PI controller, the MPC controller in [9] can only steer the temperature close to the set point within an acceptable range. Although the PI controller regulates the temperature precisely at the set point, the control effort produced by the PI controller was much larger than that of the MPC. Moreover, the PI controller signal showed more fluctuation, while the MPC generated a much smoother control signal. This observation is actually consistent with the results obtained in this paper.

The prediction models in MPC design can be either physical models based on first principles or statistical models based on measured data. Physical models make use of the thermal dynamics, and determine the coefficients based on manufacturer documentation or grey-box identification techniques [10, 11]. Statistical models fit mathematical functions to the measured data. The widely used models in thermal control include autoregressive with exogenous (ARX) model, autoregressive moving average (ARMA) model, finite impulse response (FIR) model, autoregressive moving average exogenous (ARMAX) model and output error (OE) model [12].

The typical centralized control structure can be difficult to implement for large scale systems due to the high computational load and maintenance cost. Moreover, the centralized scheme usually has limited control flexibility [13]. Hence, the non-centralized control algorithms, which can be categorized as decentralized MPC and distributed MPC, are developed to overcome these issues in [14]. The decentralized MPC does not allow for communication between local controllers, while distributed MPC requires information exchange between local controllers to enhance the accuracy of predictions. The construction of coupling models for each local system in this paper is inspired by the idea from [14]. Furthermore, if multiple iterations are introduced in the distributed MPC, the Nash Optimality can be obtained with a local cost function [15], or the Pareto Optimality can be obtained with a global cost function [16, 17].

Although the increased interest on MPC for building thermal systems is undeniable, most of the works validate the designed MPC controllers on elementary physical models or simple identified models [9, 14–17]. In this work, a HAMBASE SIMULINK model of Museum Hermitage Amsterdam is utilized to generate data for identification purposes. Similar to [18, 19], a network of resistors and capacitors is used to model the thermal dynamics of a multi-zone building, and the parameters in the grey-box model are determined

by least-squares estimation (LSE) [20]. The centralized MPC controller that is designed based on the identified model is applied on the HAMBASE SIMULINK model for validation. The HAMBASE model is a nonlinear and time-variant simulation model that is developed by Martin de Wit [21], which has been proven to accurately simulate the heat and vapour flows in a building [22]. Therefore, the HAMBASE model can be considered as the “reality” of the museum.

The first aim of this project is to exploit a Linear Time Invariant (LTI) model of the museum with physical meaning for the design of MPC, and perform identification to estimate the corresponding parameters. In comparison, the MPC controller in [14] is designed on a simple ARX model, and the model in [9, 14, 23] does not consider some dominant uncontrollable inputs (e.g., outdoor temperature or solar radiation). The second aim of this project is to design the centralized MPC controller based on the identified model. Integral action is employed to eliminate the offset, and as one of the main contributions of this project, the MPC controller with integral action is also applied on the HAMBASE SIMULINK model. In comparison, [9, 24, 25] provide the simulation results on the same model that is used in the controller design, and the integral action is not introduced in any of them. The third aim of this project is to design the multi-iteration distributed MPC for the identified thermal model, and develop the algorithm of distributed MPC with integral action, which is another novel contribution of this project. In comparison, to the author’s best knowledge, there is no available distributed MPC algorithm with integral action.

The remainder of this thesis is structured as follows. Section II briefly introduces the HAMBASE SIMULINK model and defines the parameters. The control problem is also formulated in this section. Section III develops the LTI model for the thermal dynamics in the museum. The parameters are determined by the LSE method, and model validation is also covered in this section. Section IV investigates the options for cost function in the centralized MPC design, and both the MPC with and without integral action are designed and applied to the HAMBASE SIMULINK model. Section V constructs the interaction model for each local system, and designs the distributed MPC for each subsystem. The distributed MPC algorithm with the combination of integral action is also developed in this section. In Section VI, the conclusions and suggestions for future research are summarized.

## II. SYSTEM DESCRIPTION AND PROBLEM FORMULATION

Museum Hermitage Amsterdam serves as a case study for this work, and the constructed HAMBASE SIMULINK model in [21, 26] contains two temperature nodes: an air temperature node ( $T_a$ ) and a “resultant” temperature node ( $T_x$ ). The convective heat losses (ventilation, inter-zonal air flows) are calculated with the air temperature  $T_a$ , while the heat losses through the envelope are calculated with the “resultant” temperature  $T_x$ . The temperature evolution of 9 zones in the Hermitage Museum can be truthfully predicted, since most of factors that can affect the indoor temperature are taken into account, which include climate information (outdoor temperature, solar radiation, relative humidity, wind speed and direction, cloud cover, etc.); visitors’ profile; operation of the building’s doors, windows and venetian blinds; and heat transfer through radiation and convection.

All heat inputs are split into convective and radiant parts by means of convection factors, and the interaction between air temperature and resultant temperature can be summarized as

$$\rho_a c_p V_a \dot{T}_a = A_t h_{cv} \left(1 + \frac{h_{cv}}{h_r}\right) (T_x - T_a) - \sum \phi_{ab} + \phi_c - \frac{h_{cv}}{h_r} \phi_r, \quad (1)$$

where  $\rho_a$  is the density of air,  $c_p$  is the specific heat of air,  $V_a$  is the volume of air of a zone,  $A_t$  is the total interior surface area of a zone,  $h_r$  is the surface heat transfer coefficient for radiation,  $h_{cv}$  is the surface heat transfer coefficient for convection,  $\phi_{ab}$  is the heat flow caused by air entering the zone with an air temperature  $T_b$ ,  $\phi_c$  is the total convective heat input (short-wave and emitted thermal radiation from casual and solar gains, heating and cooling), and  $\phi_r$  is the total radiant heat input. Note that  $\phi_{ab}$ ,  $\phi_c$  and  $\phi_r$  are nonlinear and time-variant function of all kinds of heat flow. Let the heat capacity  $C_a = \rho_a c_p V_a$ , and coupling coefficient  $L_{xa} = A_t h_{cv} \left(1 + \frac{h_{cv}}{h_r}\right)$ . The equation above becomes

$$C_a \dot{T}_a = L_{xa} (T_x - T_a) - \sum \phi_{ab} + \phi_c - \frac{h_{cv}}{h_r} \phi_r. \quad (2)$$

Since this detailed building model yields tedious modeling, long simulation time, strong nonlinearity and time variability [19], a simplified thermal model that has a similar structure with equation (2) should be developed for use in MPC. To investigate and illustrate the MPC algorithms under an appropriate complexity, this paper only focuses on the thermal dynamics of the main exhibition area, which is the most crucial 3 zones in

the museum.

According to the aforementioned system’s properties, our objectives are formulated and described as below:

**Problem 1** Develop a LTI model base on physical insight to model the thermal dynamics of 3 zones in Hermitage museum, and estimate its parameters. Validate the identified model by goodness of fit and residual correlation test.

**Problem 2** Design a centralized MPC algorithm based on the identified model, and include integral action to eliminate offset. The standard MPC controller, MPC controller with integral action and conventional PI controller should be validated on the HAMBASE SIMULINK model, and the advantages and limitations of these controllers should be analyzed.

**Problem 3** Design the multi-iteration distributed MPC on the identified model, and develop the algorithm of distributed MPC with integral action. Investigate the convergence along the iterations of distributed MPC.

## III. THERMAL MODEL DEVELOPMENT

This section discusses the modeling of the temperature evolution in the 3 zones of Hermitage museum. In section III-A, a simplified physical model of the thermal dynamics is derived under the extensively accepted assumptions. Section III-B describes the identification process of the determination of corresponding parameters. The goodness of fit and residual correlation are checked in Section III-C. The identified model is employed in the prediction model of MPC design in Section IV and Section V.

### A. Physical Modeling

A LTI model with physical parameters is introduced in [19]. Therein, it was proven that this LTI model is able to reproduce the climate change of the HAMBASE model; however, this simplified model is a single zone model for free floating buildings, i.e., buildings that are not heated or cooled. Thavlov et al. [27] and Agbi et al. [18] introduced a network of resistors and capacitors that can be used to model the thermal dynamics of a multi-zone building with a heat power system. The model developed in this work is a combination of the models from these three papers. The simplified model is developed with the following assumption:

- 1) The air in each zone is perfectly mixed, so each zone only has one uniform air temperature.
- 2) The surface coefficients for convection and radiation are constant.
- 3) All heat inputs are distributed in such a way that all surfaces absorb the same amount per unit.

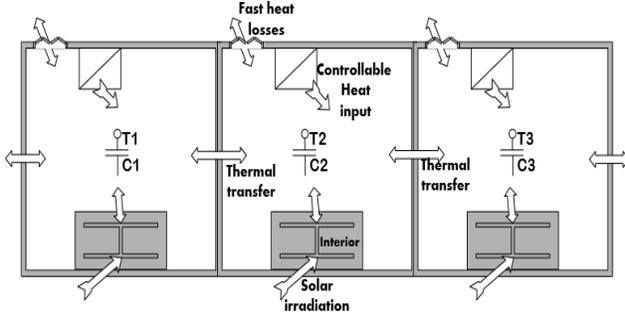


Fig. 1: Thermal model of three zones in Hermitage museum.

- 4) The natural ventilation of buildings has a relatively low speed.

Based on these assumptions, the thermal model of the three zones is then depicted in Fig. 1, in which the fast heat losses (e.g. ventilation and transmission through glazing) are considered as an uncontrollable input, and the other dominant uncontrollable input solar radiation is considered mainly absorbed by an imaginary interior node.

$T_2$  represents the air temperature of the main exhibition zone (Zone 6 in the SIMULINK model), while  $T_1$  (Zone 4 in the SIMULINK model) and  $T_3$  (Zone 5 in the SIMULINK model) represent two adjacent zones. Then the interaction between different temperature nodes can be modelled with resistance (conductance) and heat capacity, and the physical meaning of all related parameters are given in Table I.

TABLE I: Parameters in the simplified thermal model

Symbol	Description	Unit
$T_j$	Indoor air temperature for zone $j$	$^{\circ}\text{C}$
$T_e$	Outdoor temperature	$^{\circ}\text{C}$
$T_{int_j}$	Interior temperature for zone $j$	$^{\circ}\text{C}$
$R_{fa_j}$	Thermal resistance of fast heat losses for zone $j$	$^{\circ}\text{C}/\text{W}$
$R_{w_j}$	Thermal resistance between air temperature nodes	$^{\circ}\text{C}/\text{W}$
$R_{int_j}$	Thermal resistance between interior temperature and indoor air	$^{\circ}\text{C}/\text{W}$
$C_j$	Heat capacity of indoor air for zone $j$	$\text{J}/^{\circ}\text{C}$
$C_{int_j}$	Heat capacity of interior for zone $j$	$\text{J}/^{\circ}\text{C}$
$q_j$	Heat input from heat power system for zone $j$	$\text{W}$
$Rad$	Solar radiation	$\text{W}/\text{m}^2$
$fI_j$	Effective radiance area for zone $j$	$\text{m}^2$
	$j = 1,2,3.$	

The heat transfer between two nodes by conduction or convection can be considered proportional to the temperature difference between the two nodes involved. The RC network of three zones that describe the thermal

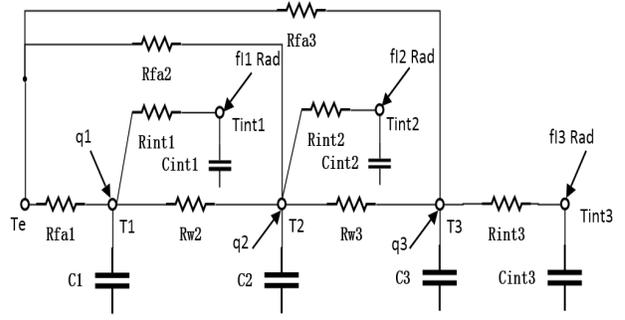


Fig. 2: Simplified thermal model.

model based on Fig. 1 is shown in Fig. 2.

For illustration purpose, the linear differential equations of Zone 6 based on this circuit are derived as

$$C_2 \dot{T}_2 = \frac{1}{R_{fa2}} (T_e - T_2) + \frac{1}{R_{int2}} (T_{int2} - T_2) + \frac{1}{R_{w2}} (T_1 - T_2) + \frac{1}{R_{w3}} (T_3 - T_2) + q_2 \quad (3)$$

$$C_{int2} \dot{T}_{int2} = \frac{1}{R_{int2}} (T_2 - T_{int2}) + fI_2 \cdot Rad \quad (4)$$

The differential equations for the three zones are then aggregated in state space form

$$\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u(t) + F_c w(t), \\ y(t) &= C_c x(t), \end{aligned} \quad (5)$$

where  $x(t) = [T_1 \ T_{int1} \ T_2 \ T_{int2} \ T_3 \ T_{int3}]^T$ ; the vector  $u(t)$  represents the input vector comprising of controllable inputs  $u(t) = [q_1 \ q_2 \ q_3]^T$ ; and the vector  $w(t)$  denotes uncontrollable inputs, including solar radiation and fast heat losses  $w(t) = [T_e \ Rad]^T$ . The matrices  $A_c, B_c, F_c$  and  $C_c$  represent the system matrices of this simplified thermal model, which have the following structure:

$$A_c = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & 0 & 0 & 0 \\ \theta_4 & \theta_5 & 0 & 0 & 0 & 0 \\ \theta_6 & 0 & \theta_7 & \theta_8 & \theta_9 & 0 \\ 0 & 0 & \theta_{10} & \theta_{11} & 0 & 0 \\ 0 & 0 & \theta_{12} & 0 & \theta_{13} & \theta_{14} \\ 0 & 0 & 0 & 0 & \theta_{15} & \theta_{16} \end{bmatrix}, \quad (6)$$

$$B_c = \begin{bmatrix} \theta_{17} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & \theta_{18} & 0 \\ 0 & 0 & 0 \\ 0 & 0 & \theta_{19} \\ 0 & 0 & 0 \end{bmatrix}, F_c = \begin{bmatrix} \theta_{20} & 0 \\ 0 & \theta_{21} \\ \theta_{22} & 0 \\ 0 & \theta_{23} \\ \theta_{24} & 0 \\ 0 & \theta_{25} \end{bmatrix}, \quad (7)$$

$$C_c = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}, \quad (8)$$

where  $\theta_1 - \theta_{25}$  are the redefined parameters that need to be identified, and the expressions of  $\theta_1 - \theta_{25}$  are provided in Appendix A. Since the purpose of modeling in this project is to predict the future output for controller design, certain physical relations between parameters are ignored in the parameters estimation for a better fitting performance.

The continuous-time model needs to be discretized with a sampling period  $T_s$ , since the system is typically observed and controlled in discrete-time. The Euler forward discretization method is implemented, and a discretized state space model is then given by

$$\begin{aligned} x(k+1) &= A(T_s)x(k) + B(T_s)u(k) + F(T_s)w(k), \\ y(k) &= Cx(k), \end{aligned} \quad (9)$$

where the discrete-time system matrices  $A, B, F$  and  $C$  are determined by

$$\begin{aligned} A(\theta) &= e^{A_c T_s} = I + A_c T_s + \frac{A_c^2 T_s^2}{2} + \dots \\ &\approx I + A_c(\theta)T_s, \end{aligned} \quad (10)$$

$$B(\theta) = \int_0^{T_s} e^{A\tau} d\tau B_c \approx \int_0^{T_s} I d\tau B_c = T_s B_c(\theta), \quad (11)$$

$$F(\theta) = \int_0^{T_s} e^{A\tau} d\tau F_c \approx \int_0^{T_s} I d\tau F_c = T_s F_c(\theta), \quad (12)$$

$$C(\theta) = C_c(\theta). \quad (13)$$

### B. Parameters Estimation

The goal is to identify the developed thermal model to be used in the MPC design subsequently. The experimental data is generated by the HAMBASE SIMULINK model in open loop with a Pseudorandom Binary Sequence (PRBS) as the excitation signal. The PRBS signal is a deterministic signal with white-noise-like properties that shifts between two values, which is also a widely used choice for linear model identification. To be specific, the input (experimental) data for the identification procedure are listed below:

- Heat input  $u(k)$  for each zone (i.e., PRBS in this case),
- Outdoor (ambient) temperature  $T_e(k)$ ,
- Solar radiation  $Rad(k)$ ,
- Air temperature  $T_j(k)$  of each zone.

The uncontrollable input sequences (i.e., outdoor temperature and solar radiation) are obtained from the national climate institution KNMI, and the small sequences are employed by the HAMBASE SIMULINK model during the experiment. The data is measured with sampling period of 5 minutes, which is sufficient for

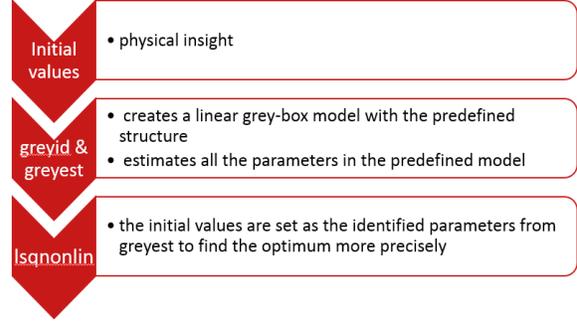


Fig. 3: Optimization progress.

thermal systems like our case due to their slow dynamics [28]. The experiments is implemented for 4 days, and the collected data samples are around 30 times of the number of parameters that need to be estimated.

Given the  $N$  samples of input  $u_k$ , uncontrollable input  $w_k$  and output  $y_k$ , the parameters are determined by Least Squares Estimation (LSE). With the parameterized system matrices  $A(\theta)$ ,  $B(\theta)$ ,  $C(\theta)$  and  $F(\theta)$  according to equations (3)-(13), the output of the developed thermal model

$$\hat{x}_{k+1}(\theta) = A(\theta)\hat{x}_k(\theta) + B(\theta)u_k + F(\theta)w_k, \quad (14)$$

$$\hat{y}_k(\theta) = C(\theta)\hat{x}_k(\theta), \quad (15)$$

should match the real output  $y_k$  from the HAMBASE SIMULINK model sufficiently well. The LSE method then leads to an optimization problem that can be expressed as

$$\theta^* = \arg \min_{\theta} \frac{1}{2} \varepsilon(\theta)^T \varepsilon(\theta), \quad (16)$$

where the output-error vector is determined by

$$\varepsilon(\theta) = Y - \hat{Y}(\theta) \quad (17)$$

with

$$Y = [y_1 \ \dots \ y_N]^T \quad (18)$$

$$\hat{Y}(\theta) = [\hat{y}_1(\theta) \ \dots \ \hat{y}_N(\theta)]^T \quad (19)$$

The initial condition  $x_0$  is also parameterized as  $x_0 = x_0(\theta)$  and estimated. This formulation results in a non-linear and non-convex optimization problem, because of the non-linear dependence of  $y_k(\theta)$  on  $\theta$ , so a good initial estimate is of great importance. As it is indicated in Fig. 3 the optimization problem is sequentially executed by two different solvers. The initial values are first determined according to their physical insight, then the optimization problem is solved by the MATLAB function `idgrey` and `greyst` with those

initial guesses. Specifically, `idgrey` allow us to create a linear grey-box model with the predefined structure and identifiable parameters, then the created system is sent to `greyest` together with the time-series data to estimate all the parameters in the predefined model. The identified parameter values are then set as the new initial values and sent to the MATLAB function `lsqnonlin`, in which the `trust-region-reflective` algorithm is applied to estimate all parameters again. According to the results of model validation in Section III-C, the thermal model that is identified by this optimization method can predict the air temperature evolution in HAMBASE SIMULINK model with sufficient accuracy.

### C. Model Validation

The experimental data is divided into fit part and validation part. Fit part constitutes 75% of the available data and validation part constitutes 25% of the available data. The simulation accuracy of the simplified thermal model is validated based on three criteria:

- Graphical Analysis;
- Numerical Analysis;
- Residual correlation test.

Graphical Analysis is based on a visual inspection of how the simulated output depicts the experimental available data. To show a wide range of relationships between the developed model and experimental data, the entire data set is viewed. The graph of the fitting performance is enclosed in Appendix B, which indicate that both the slow dynamics and fast dynamics are captured and reproduced with feasible accuracy.

Numerical analysis usually compresses the fitting performance into a single number. The Mean Square Error (MSE), which is the mean of the square of the deviation between simulated data and experimental data, is considered in this work.

The values of MSE are non-negative, and values that are close to zero indicate a good fit. The MSE of this fit for zone 4, 5 and 6 are 0.7498, 0.5552 and 0.5290, respectively. Compared with the MSE of the identified thermal model in [19], the MSE of this simplified model is relatively small and implies a sufficient accuracy.

Residual analysis consists of two tests: the whiteness test and the independence test, which is analyzed by auto-correlation and cross-correlation of the residuals  $e_k$ , which is the deviation between simulated outputs and experimental data. The auto-correlation is utilized to inspect whether the model is over simplified or inputs are improperly ignored, and the cross-correlation is used

to inspect the correctness of model structure, and their expressions are given by

$$R_{e^i}(\tau) = \frac{E[(e_{k+\tau}^i - \mu_{e^i})(e_k^i - \mu_{e^i})]}{\sigma_{e^i}^2}, \quad (20)$$

$$R_{e^i I^j}(\tau) = \frac{E[(e_{k+\tau}^i - \mu_{e^i})(I_k^j - \mu_{I^j})]}{\sigma_{e^i} \sigma_{I^j}}, \quad (21)$$

$i = 1, 2, 3; \quad j = 1, 2, 3, 4, 5,$

where  $\tau$  are the lagged samples of the auto-correlation and cross-correlation.  $R_{e^i}(\tau)$  represent the auto-correlation of 3 different zones, and  $R_{e^i I^j}(\tau)$  represent the cross-correlation between 3 zones and 5 inputs (include 2 uncontrollable inputs).  $e_k^i$  is the residual of simulated outputs and measured outputs for  $i$ -th zone at time instant  $k$ , and  $I_k^j$  is the  $j$ -th input at time instant  $k$ ,  $\mu$  and  $\sigma$  are the mean operator and standard deviation operator, respectively.

The results based on the 99% confidence interval is also enclosed in Appendix B. The 99% confidence region marking statistically insignificant correlations displays as a shaded region around zero. The results show that all auto-correlation exceed the confidence interval, while all cross-correlation stay within the confidence interval. Kramer et al. [19] and Ljung [29] state that the model should pass both the whiteness and the independence tests except output-error models, and less attention should be paid on the auto-correlation function if no error model is included. The cross-correlation shows that the model structure is correct and it describes the influence from inputs to outputs correctly.

In conclusion, the identified thermal model satisfies all three criteria of the model validation, so it is capable of reproducing temperature dynamics of the three zones and can be subsequently used for MPC design in Section IV.

## IV. CENTRALIZED MPC OF HERMITAGE MUSEUM

This section discusses the design of centralized MPC based on the thermal model developed in Section III. The centralized MPC should track a designed static reference or repeating stair sequence according to the required indoor temperature of each zone. The stability is guaranteed by terminal penalty for the standard MPC design, so this problem amount to a reference tracking MPC with stability guarantees.

In Section IV-A, the formulation of the MPC problem is derived, and the commonly used options for cost function are investigated. The MPC design with integral action is formulated in Section IV-B to remove offset. In Section IV-C, the designed standard MPC, MPC

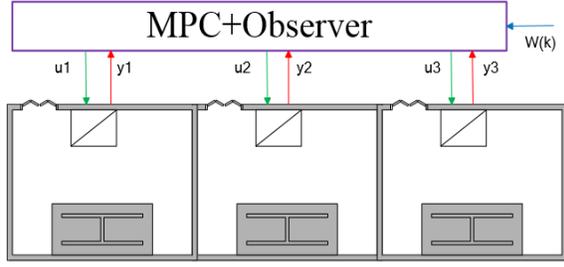


Fig. 4: Centralized MPC scheme.

with integral action and a conventional PI controller are validated on the HAMBASE SIMULINK model, and an assessment of the results is provided as well.

#### A. Problem formulation and selection of cost function

In the centralized MPC, the temperatures of three zones are controlled by one MPC controller, and a typical Luenberger observer is utilized to estimate the states of the prediction model (Fig. 4).

Consider the identified model in Section III:

$$x(k+1) = Ax(k) + Bu(k) + Fw(k), \quad (22)$$

$$y(k) = Cx(k), \quad (23)$$

where  $x(k) \in \mathbb{R}^n$ ,  $u(k) \in \mathbb{R}^m$ ,  $y(k) \in \mathbb{R}^m$  and  $w(k) \in \mathbb{R}^q$ . The constrained MPC for reference tracking is defined in the standard form

$$\min_{U_k} J(x_{0|k}, U_k) \quad (24a)$$

$$s.t. \quad x_{i+1|k} = Ax_{i|k} + Bu_{i|k} + Fw(k+i) \quad (24b)$$

$$x_{i|k} \in \mathbb{X} \subseteq \mathbb{R}^n \quad (24c)$$

$$u_{i|k} \in \mathbb{U} \subseteq \mathbb{R}^m \quad (24d)$$

$$x_{N|k} \in \mathbb{X}_T \subseteq \mathbb{R}^n \quad (24e)$$

where  $x_{i|k}$  and  $u_{i|k}$  denotes the predicted state and input at time instant  $k+i$ ,  $i \geq 0$ , based on data at time  $k$ , respectively,  $\mathbb{X}$  and  $\mathbb{U}$  represents the set of constraints of the state and input, respectively. With the prediction horizon denoted as  $N$ , the prediction model in the MPC design can be represented as

$$X_k = \Phi x(k) + \Gamma U_k + \Lambda W_k \quad (25)$$

where

$$X_k = [x_{1|k}^T \quad x_{2|k}^T \quad \cdots \quad x_{N|k}^T]^T,$$

$$U_k = [u_{0|k}^T \quad u_{1|k}^T \quad \cdots \quad u_{N-1|k}^T]^T,$$

$$W_k = [w(k)^T \quad w(k+1)^T \quad \cdots \quad w(k+N-1)^T]^T$$

$$\Phi = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^N \end{bmatrix}, \quad \Gamma = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{bmatrix},$$

$$\Lambda = \begin{bmatrix} F & 0 & \cdots & 0 \\ AF & F & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}F & A^{N-2}F & \cdots & F \end{bmatrix}.$$

a) *Option 1 of the cost function:* The first commonly used option for the cost function of reference tracking MPC is considered to penalize the distance to the reference state and input, i.e.,

$$J(x(k), U_k) = (x(k) - x_{0|k}^{ss})^T Q (x(k) - x_{0|k}^{ss}) + (X_k - X_k^{ss})^T \Omega (X_k - X_k^{ss}) + (U_k - U_k^{ss})^T \Psi (U_k - U_k^{ss}), \quad (26)$$

$$s.t. \quad X_k \in \mathbb{X}^{n \times N} \subseteq \mathbb{R}^{n \times N}$$

$$U_k \in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$$

where  $\Psi$  and  $\Omega$  are diagonal matrices that consist of penalty matrices  $Q$ ,  $R$  and  $P$ .  $Q$  and  $R$  are positive definite weighting matrices that penalize states and inputs, respectively, and  $P$  is the terminal penalty that is calculated based on the algebraic Riccati equation to guarantee the stability.  $U_k^{ss}$  and  $X_k^{ss}$  are the vector of reference state and input, respectively, which is given by

$$X_k^{ss} = \begin{bmatrix} x_{1|k}^{ss} \\ x_{2|k}^{ss} \\ \vdots \\ x_{N|k}^{ss} \end{bmatrix}, \quad U_k^{ss} = \begin{bmatrix} u_{0|k}^{ss} \\ u_{1|k}^{ss} \\ \vdots \\ u_{N-1|k}^{ss} \end{bmatrix}.$$

$U_k^{ss}$  and  $X_k^{ss}$  are computed via the prediction model  $x_{i+1|k}^{ss} = Ax_{i|k}^{ss} + Bu_{i|k}^{ss} + Fw(k+i)$  and the equality  $Cx_{i+1|k}^{ss} = r_{i|k}$ . To guarantee an unique solution (the number of equations should equal to the number of unknowns),  $x_{N|k}^{ss}$  is considered as an equilibrium state, i.e.,  $x_{N|k}^{ss} = Ax_{N|k}^{ss} + Bu_{N|k}^{ss} + Fw(k+N)$ . The equations can be further written in a compact form:

$$M_{SH} \begin{bmatrix} x_{0|k}^{ss} \\ X_k^{ss} \\ U_k^{ss} \\ u_{N|k}^{ss} \end{bmatrix} = M_R \quad (27)$$

where

$$M_{SH} = \begin{bmatrix} A & -I_n & 0 & B & 0 & 0 \\ & \ddots & \ddots & & \ddots & \\ 0 & A & -I_n & 0 & B & 0 \\ 0 & 0 & A - I_n & 0 & 0 & B \\ C_r & 0 & 0 & 0 & 0 & 0 \\ & \ddots & & & \ddots & \\ 0 & C_r & 0 & 0 & 0 & 0 \\ 0 & 0 & C_r & 0 & 0 & 0 \end{bmatrix},$$

$$M_R = \begin{bmatrix} -Fw(k) \\ \vdots \\ -Fw(k+N) \\ R_k \end{bmatrix}, \quad R_k = \begin{bmatrix} r_{0|k} \\ r_{1|k} \\ \vdots \\ r_{N|k} \end{bmatrix}.$$

Substituting  $X_k$ ,  $U_k$ ,  $X_k^{ss}$  and  $U_k^{ss}$  in (26) based on (25) and (27), the cost function proposed in equation (26) then amounts to a standard quadratic programming cost as below:

$$J(x(k), U_k) = \frac{1}{2} U_k^T G U_k + U_k^T F x(k) + U_k^T F_w W_k + U_k^T Y M_{SH}^{-1} M_R + \text{constant}, \quad (28)$$

where

$$G = 2(\Psi + \Gamma^T \Omega \Gamma), \quad F = 2\Gamma^T \Omega \Phi, \quad F_w = 2\Gamma^T \Omega \Lambda$$

$$Y = -2 \begin{bmatrix} \Gamma^T \Omega & \Psi \end{bmatrix} \begin{bmatrix} 0_{(n+m)N \times n} & I_{(n+m)N} & 0_{(n+m)N \times p} \end{bmatrix}$$

However, the temperature response under this formulation is unreasonable (Fig. 5). Note that the tracking performances before 280 minutes are actually good for all three zones, but they become terrible when solar radiation (one of the uncontrollable input to the system) start to rapidly increase (Fig. 6) around 340 minutes. Consider the prediction horizon is 12 (60 minutes), the increase of solar radiation around 340 minutes is available for the MPC controller around 280 minutes. This observation indicates that this terrible response may influenced by the change of the uncontrollable input.

To further investigate this terrible response, a simple two-dimensional linear system with an uncontrollable input  $w(k)$  is defined in the Appendix C. Several tests are implemented to explore the influence of  $w(k)$  on the calculation of  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27). The results indicate that  $X_k^{ss}$  and  $U_k^{ss}$  can show zigzag dynamics with quite large varying range if abrupt changes occur in the uncontrollable input, and the violently changed  $X_k^{ss}$  and  $U_k^{ss}$  in the cost function would result in poor tracking performance of the output. These results are consistent with observations on this thermal

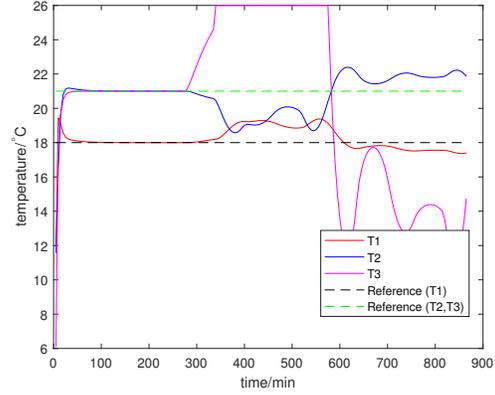


Fig. 5: Temperature response using cost function (26)

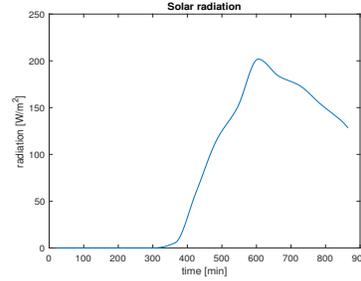


Fig. 6: Solar radiation

model, for instance, the computed reference input 3 varies violently when one of the uncontrollable inputs starts to change dramatically (Fig. 7). In summary, the considerably increase of one uncontrollable input results in violently changed  $X_k^{ss}$  and  $U_k^{ss}$ , so the computation of  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27) in this situation maybe unreliable. Hence, the MPC controller that penalizes the distance to the violently changed  $X_k^{ss}$  and  $U_k^{ss}$ , which are unreliable, would induce terrible response.

In conclusion, one should be careful when designing the MPC controller based on the cost function (26)

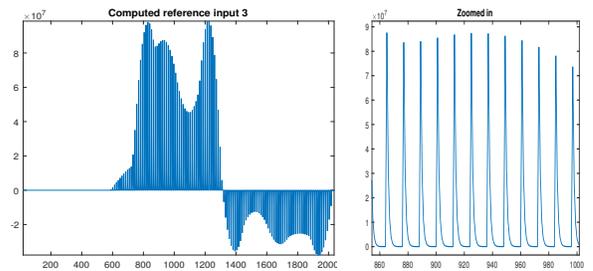


Fig. 7: 3rd reference input in  $U_k^{ss}$

if the model includes uncontrollable inputs that may considerably change within one prediction horizon.

b) *Option 2 of the cost function:* The MPC design in the rest of this paper considers the other extensively used cost function to track references, which penalizes the deviation of the outputs from the corresponding references and the increments of the inputs, i.e.,

$$\begin{aligned} J(x(k), \Delta U_k) &= (y(k) - r(k))^T Q (y(k) - r(k)) \\ &+ (Y_k - R_k)^T \Omega (Y_k - R_k) + \Delta U_k^T \Psi \Delta U_k, \quad (29) \\ \text{s.t. } Y_k &\in \mathbb{Y}^{q \times N} \subseteq \mathbb{R}^{q \times N} \\ \Delta U_k &\in \Delta \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N} \\ U_k &\in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N} \end{aligned}$$

Note that the constraints on  $U_k$  should be translated to the constraints on  $\Delta U_k$  based on equation (30), and the outputs should also be expressed with the increments of the inputs  $\Delta U_k$  instead of inputs  $U_k$ .

$$U_k = T u^*(k-1) + H \Delta U_k \quad (30)$$

where  $u^*(k)$  is the optimal control at time instant  $k$ ,

$$T = \begin{bmatrix} I_m \\ I_m \\ \vdots \\ I_m \end{bmatrix}, \quad H = \begin{bmatrix} I_m & 0_m & \cdots & 0_m \\ I_m & I_m & \cdots & 0_m \\ \vdots & \vdots & \ddots & \vdots \\ I_m & I_m & \cdots & I_m \end{bmatrix}.$$

The prediction model for this cost function is then represented as

$$Y_k = \tilde{\Phi} x(k) + \tilde{\Gamma} (T u^*(k-1) + H \Delta U_k) + \tilde{\Lambda} W_k, \quad (31)$$

where

$$\begin{aligned} Y_k &= [y_{1|k}^T \quad y_{2|k}^T \quad \cdots \quad y_{N|k}^T]^T, \\ \Delta U_k &= [\Delta u_{0|k}^T \quad \Delta u_{1|k}^T \quad \cdots \quad \Delta u_{N-1|k}^T]^T, \\ \tilde{\Phi} &= \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^N \end{bmatrix}, \quad \tilde{\Gamma} = \begin{bmatrix} CB & 0 & \cdots & 0 \\ CAB & CB & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}B & CA^{N-2}B & \cdots & CB \end{bmatrix} \\ \tilde{\Lambda} &= \begin{bmatrix} CF & 0 & \cdots & 0 \\ CAF & CF & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ CA^{N-1}F & CA^{N-2}F & \cdots & CF \end{bmatrix}. \end{aligned}$$

The cost function proposed in equation (29) then leads to the standard form of quadratic programming:

$$\begin{aligned} J(x(k), \Delta U_k) &= \text{constant} + \frac{1}{2} \Delta U_k^T G \Delta U_k \\ &+ \Delta U_k^T (F_x x(k) + F_w W_k + F_u u(k-1) + F_r R_k), \quad (32) \end{aligned}$$

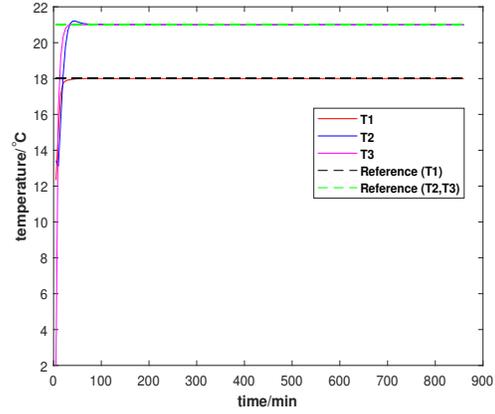


Fig. 8: Temperature response using cost function (29)

where

$$\begin{aligned} G &= 2(\Psi + H^T \tilde{\Gamma}^T \Omega \tilde{\Gamma} H), \\ F_x &= 2H^T \tilde{\Gamma}^T \Omega \tilde{\Phi}, \quad F_w = 2H^T \tilde{\Gamma}^T \Omega \tilde{\Lambda}, \\ F_u &= 2H^T \tilde{\Gamma}^T \Omega \tilde{\Gamma}^T, \quad F_r = -2H^T \tilde{\Gamma}^T \Omega \end{aligned}$$

The penalty matrices is determined by making trade-off between the control performance in each zone, Fig. 8 shows the temperature response with the penalty are tuned as  $Q = \text{blkdiag}(50e7, 50e6, 10e7)$ ,  $R = \text{blkdiag}(3, 2, 5)$ , and  $N = 12$ .

It can be observed that a good tracking performance is achieved on the simulation of the simplified model. In conclusion, the cost function proposed in equation (29), which penalizes the deviation of the outputs from references and the increments of the inputs, is more trustable in the MPC design on the model that includes considerably changed uncontrollable inputs.

### B. Centralized MPC with integral action

Different methods have been presented to achieve offset free control against unknown disturbance in MPC algorithms [30, 31]. Integral action is one effective method to achieve this purpose by taking the difference of both sides of the original model (22)-(23), i.e.,

$$\Delta x(k+1) = A \Delta x(k) + B \Delta u(k) + F \Delta w(k), \quad (33)$$

$$\begin{aligned} \Delta y(k+1) &= C \Delta x(k+1) \\ &= CA \Delta x(k) + CB \Delta u(k) + CF \Delta w(k), \quad (34) \end{aligned}$$

where

$$\Delta x(k) = x(k) - x(k-1), \quad (35)$$

$$\Delta u(k) = u(k) - u(k-1), \quad (36)$$

$$\Delta y(k) = y(k) - y(k-1). \quad (37)$$

With the redefined state  $x_I = [\Delta x(k) \ y(k)]^T$ , an enlarged state space model is constructed based on equation (33)-(34) as

$$x_I(k+1) = A_I x_I(k) + B_I \Delta u(k) + F_I \Delta w(k), \quad (38)$$

$$y(k) = C_I x_I(k), \quad (39)$$

where

$$A_I = \begin{bmatrix} A & O_{n_p \times q} \\ CA & I_q \end{bmatrix}, \quad B_I = \begin{bmatrix} B \\ CB \end{bmatrix},$$

$$F_I = \begin{bmatrix} F \\ CF \end{bmatrix}, \quad C_I = [O_{q \times n_p} \quad I_q].$$

The prediction model for the enlarged system can be written in compact form as

$$Y_k = \tilde{\Phi}_I x(k) + \tilde{\Gamma}_I \Delta U_k + \tilde{\Lambda}_I W_k, \quad (40)$$

where  $\tilde{\Phi}_I$ ,  $\tilde{\Gamma}_I$  and  $\tilde{\Lambda}_I$  are coefficient matrices consisting of  $A_I$ ,  $B_I$ ,  $F_I^j$  and  $C_I^j$ .

The cost function is still designed in the same form of equation (29) to penalize deviation of the outputs from the corresponding references and the increments of the inputs, so the quadratic cost can be rewritten compactly as

$$J(x(k), \Delta U_k) = \text{constant} + \frac{1}{2} \Delta U_k^T G \Delta U_k$$

$$+ \Delta U_k^T (F_x x_I(k) + F_w W_k + F_r R_k), \quad (41)$$

$$\text{s.t. } Y_k \in \mathbb{Y}^{q \times N} \subseteq \mathbb{R}^{q \times N}$$

$$\Delta U_k \in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$$

$$U_k \in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$$

where

$$G = 2(\Psi + (\tilde{\Gamma}_I)^T \Omega \tilde{\Gamma}_I), \quad F_x = 2(\tilde{\Gamma}_I)^T \Omega \tilde{\Phi}_I,$$

$$F_w = 2(\tilde{\Gamma}_I)^T \Omega \tilde{\Lambda}_I, \quad F_r = -2(\tilde{\Gamma}_I)^T \Omega.$$

Note that the constraints on  $U_k$  should also be translated to the constraints on  $\Delta U_k$  based on equation (30) in the quadratic programming. The redefined state  $x_I(k)$  need to be estimated at each current instant, so a typical Luenberger observer is designed as

$$\hat{x}_I(k+1) = A_I \hat{x}_I(k) + B_I \Delta u(k) + F_I \Delta w(k)$$

$$+ L(y(k) - C_I \hat{x}_I(k)), \quad (42)$$

where  $L$  is the observer gain to improve convergence.

To validate the correctness of integral action, linear increased process noise and measurement noise are added to the identified model, and the standard MPC designed in Section IV-A and MPC with integral action are implemented on the model with the same noise. Fig. 9 and Fig. 10 show the temperature response from standard MPC and MPC with integral action, respectively. It can

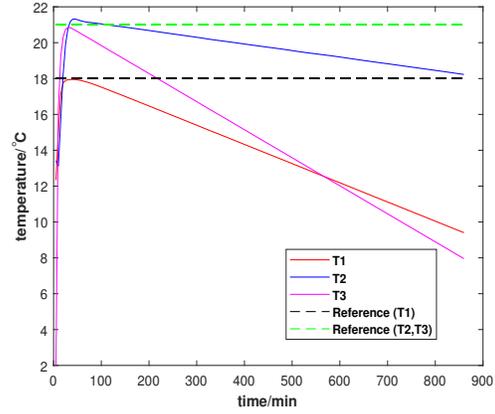


Fig. 9: Standard MPC on the model with disturbance

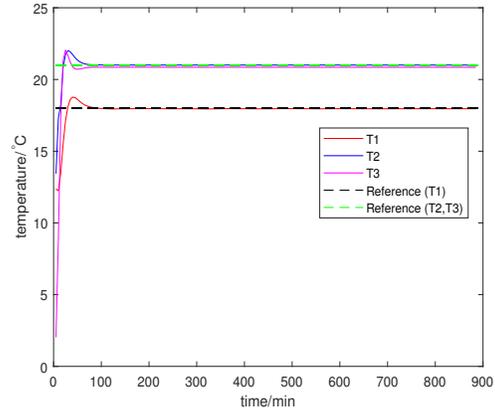


Fig. 10: MPC with integral action on the model with disturbance

be observed that the MPC with integral action is capable of removing the influence of slowly changed disturbance.

### C. Validation on the HAMBASE SIMULINK model

To validate the control algorithm on the real application, the designed MPC in Section IV-A and Section IV-B are compiled in the MATLAB Function block of the SIMULINK environment, where the formulated MPC problem is solved by the `mpcqp solver`. A Luenberger Observer block is also employed to estimate the current state for both standard MPC and MPC with integral action, since only the output (air temperature of each zone) is available from the HAMBASE Building model block. The observer gain should also be carefully tuned in this real application case, since low observer gain results in slow convergence to

the system states, but high observer gain leads to a peaking phenomenon in which initial estimator error can be prohibitively large. Either situation would make the MPC problem infeasible.

The retuning of MPC is also required in this real application case, the penalties for standard MPC are tuned as  $Q = \text{blkdiag}(56e8, 182e7, 64e6)$ ,  $R = \text{blkdiag}(1482, 60, 30)$ , and  $N = 12$ , and the penalties for MPC with integral action are tuned as  $Q = \text{blkdiag}(56e8, 100e7, 64e6)$ ,  $R = \text{blkdiag}(882, 50, 30)$ , and  $N = 12$ .

Note that the control performance of MPC is sensitive to the accuracy of the identified model, Appendix D shows that the control performance is very poor if the prediction model that MPC uses has relatively lower accuracy than the model identified in Section III.

To have a better comparison of the control methods, a simple PI controller is also designed for the HAMBASE SIMULINK model. Fig. 11 shows the temperature response with PI controller, standard MPC controller and MPC controller with integral action. Fig. 12 shows the optimized heat input, which can be considered as an indicator of energy consumption.

It can be observed that offset occur in the response of standard MPC, while offsets are removed by the MPC with integral action. Although the profile of energy consumption is close for these three controllers, MPC actually consumes less energy. This difference can be observed more clearly if the reference varies during the experiment. A new reference is designed as stair sequences, and the MPC with integral action and PI controllers are designed to track the new references. The temperature response and energy consumption can be seen in Fig. 13-14 and Table II.

Zone	PI	MPC with integral action
4	3.7638e+09 J	3.6200e+09 J
5	1.5803e+09 J	1.4991e+09 J
6	2.1127e+09 J	2.0103e+09 J
sum	7.4567e+09 J	7.1294e+09 J

TABLE II: Energy Consumption

It can be observed that MPC with integral action consumes around 5% less energy than PI controllers, since the total energy consumption for the museum is large, 5% saving can be considered valuable, and more proportion of energy is expected to be saved if the references are more frequently changed. Furthermore, as it can be seen in Fig. 14, the violent change in the heat input generated by the PI is not allowed or will damage the heat power system, while the heat input generated by MPC is mild.

In conclusion, the control performance of MPC relies

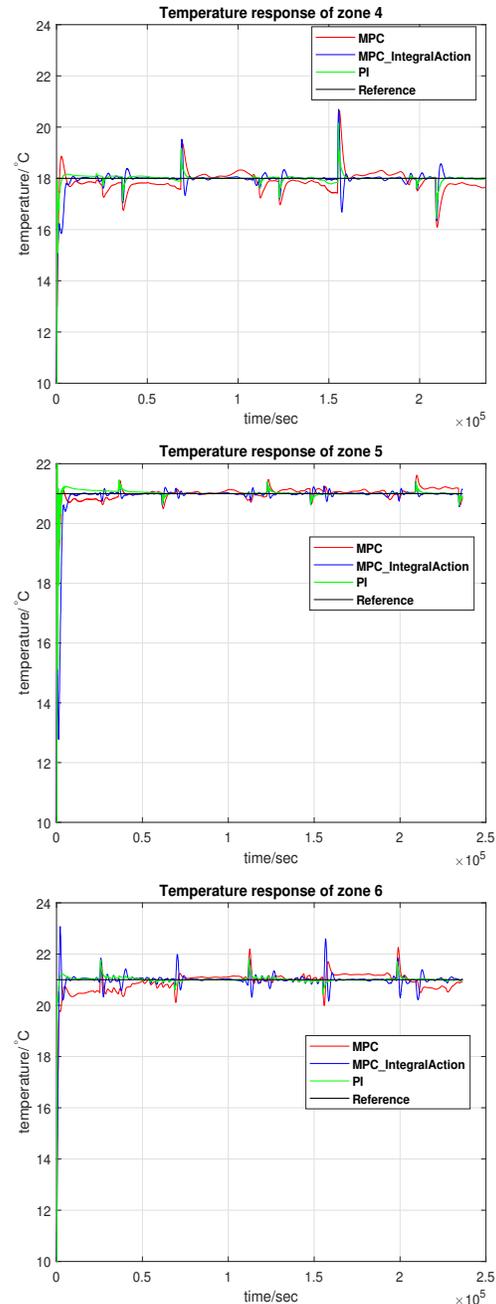


Fig. 11: Temperature response of the HAMBASE model

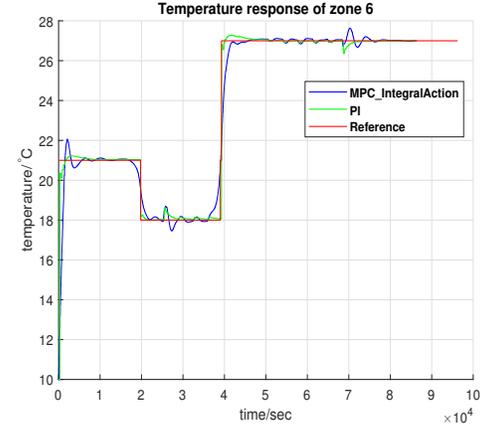
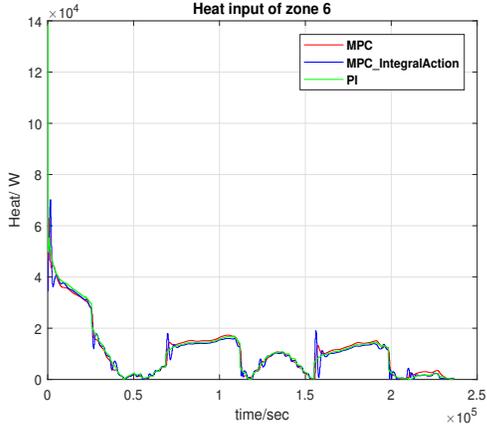
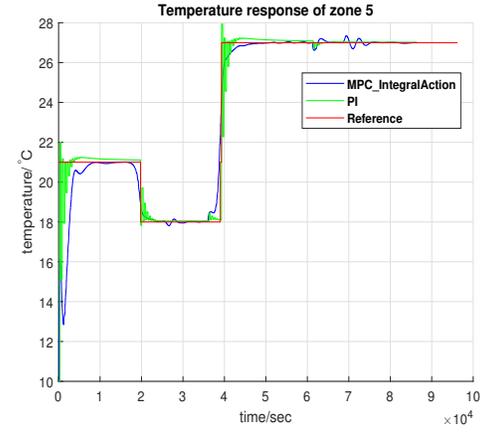
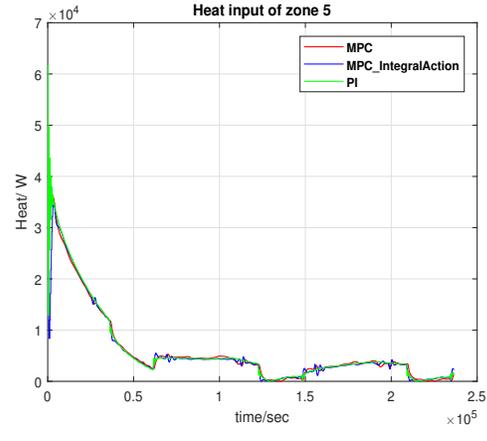
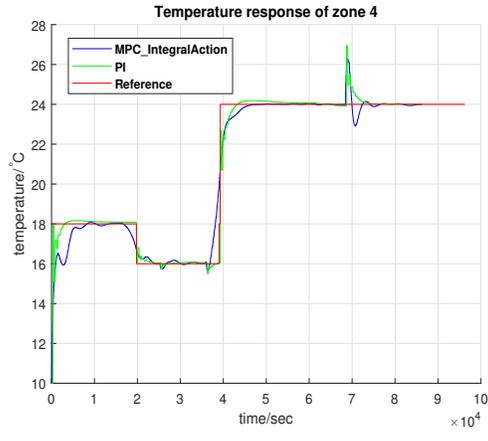
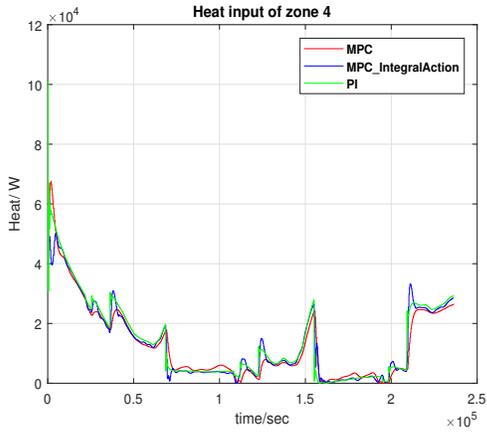


Fig. 12: Energy consumption of the HAMBASE model      Fig. 13: Temperature response of the HAMBASE model

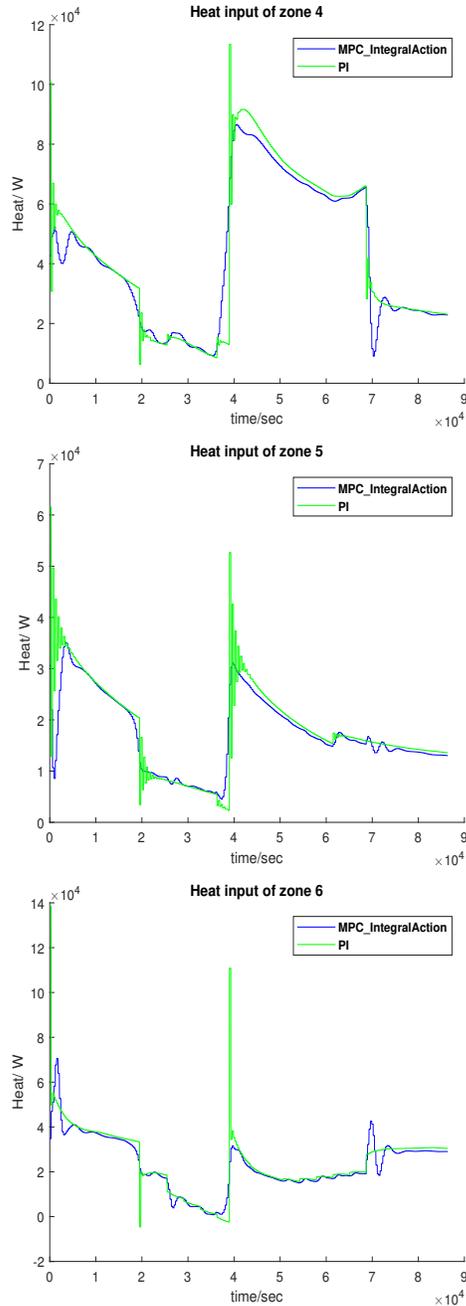


Fig. 14: Energy consumption of the HAMBASE model

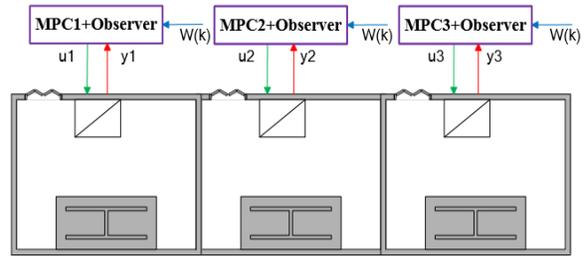


Fig. 15: Decentralized MPC scheme.

on the accuracy of the prediction model; MPC with integral action shows advantage of offset rejection compared with standard MPC, and outperforms the conventional PI controller with respect to energy saving. Although the control performances are good, the computational load of the centralized MPC exponentially increase with the system size. Therefore, the investigation of non-centralized MPC algorithms is motivated.

## V. NON-CENTRALIZED MPC

The centralized MPC has limited control flexibility when deal with different objectives for different zones, and the implementation can be prohibitively slow due to the high computational load for large scale systems (e.g., thermal systems with multiple zones). Hence, the non-centralized control structure is usually advocated to overcome these problems. The non-centralized MPC algorithms can be categorized as decentralized MPC and distributed MPC. In Section V-A, the decentralized MPC is formulated and applied on the developed model from Section III. Section V-B decomposes the overall system model into appropriate subsystem models, in which interaction between each zone is considered. The distributed MPC controllers with multi-iteration are designed based on each local model in this Section. Section V-C develops the algorithm of multi-iteration distributed MPC with the combination of integral action.

### A. Decentralized MPC

Decentralized MPC regulates the temperature of each zone by several independent controllers (Fig. 15), so the computational complexity would be reduced. Intuitively, the performance of temperature control may not be guaranteed, since the interactions between each zone are directly ignored in the prediction model.

The thermal model proposed for the entire three zones

is split into three independent models in the following form:

$$\begin{aligned} x^j(k+1) &= A^j x^j(k) + B^j u^j(k) + F^j w(k), \\ y^j(k) &= C^j x^j(k), \end{aligned} \quad (43)$$

$j = 1, 2, 3,$

where  $x^j$ ,  $u^j$  and  $y^j$  are the state, input and output of  $j$ -th zone, respectively,  $A^j$ ,  $B^j$ ,  $C^j$  and  $F^j$  are the corresponding system matrices as given below:

$$A^j = S^j A (S^j)^T, \quad (44)$$

$$B^j = S^j B Z^j, \quad (45)$$

$$F^j = S^j F, \quad (46)$$

$$C^j = [1 \ 0], \quad (47)$$

with

$$S^j = \begin{cases} \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}, & j = 1, \\ \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}, & j = 2, \\ \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, & j = 3, \end{cases}$$

$$Z^j = \begin{cases} [1 \ 0 \ 0]^T, & j = 1, \\ [0 \ 1 \ 0]^T, & j = 2, \\ [0 \ 0 \ 1]^T, & j = 3. \end{cases}$$

The input sequence for each zone is solved by minimizing a local cost function

$$\begin{aligned} J^j(x^j(k), \Delta U_k^j) &= (y^j(k) - r^j(k))^T Q (y^j(k) - r^j(k)) \\ &+ (Y_k^j - R_k^j)^T \Omega (Y_k^j - R_k^j) + (\Delta U_k^j)^T \Psi \Delta U_k^j, \end{aligned} \quad (48)$$

*s.t.*  $Y_k \in \mathbb{Y}^{q \times N} \subseteq \mathbb{R}^{q \times N}$   
 $\Delta U_k \in \Delta \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$   
 $U_k \in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$

in parallel and independently. Fig. 16 shows the simulation results on the thermal system identified in Section III. It can be observed that the decentralized MPC fails to achieve a good control performance, which implies that the thermal coupling between zones is significant, and a communication-based distributed control scheme is necessary.

### B. Distributed MPC with multiple iterations

Distributed MPC algorithms are the state of the art in control of large-scale dynamically coupled systems. Distributed MPC also uses local controllers for each subsystem, which is similar to decentralized MPC, while interactions between each subsystem are considered and

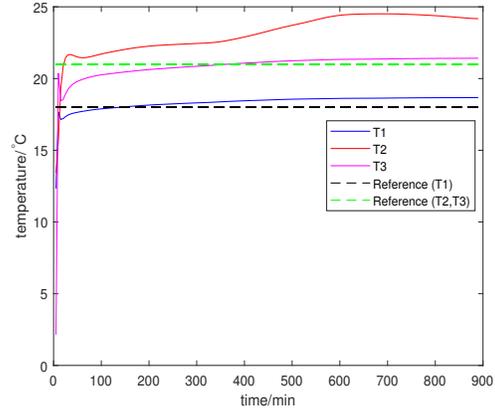


Fig. 16: Temperature response with decentralized MPC

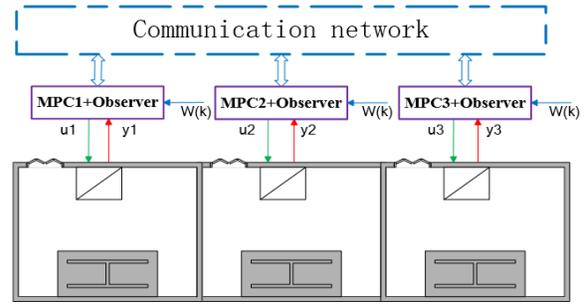


Fig. 17: Distributed MPC scheme.

the predicted output sequences are exchanged by local controllers via a communication network (Fig. 17).

Distributed MPC relies on decomposing the overall system model into appropriate subsystem models. In the case of thermal system, the coupling elements between each zone usually are the output of the adjacent zones (the air temperature). Hence, the local thermal model including the influence with the adjacent zones can be expressed as

$$\begin{aligned} x^j(k+1) &= A^j x^j(k) + B^j u^j(k) + F^j w(k) \\ &\quad + B^{sj} y^{sj}(k), \\ y^j(k) &= C^j x^j(k), \end{aligned} \quad (49)$$

$j = 1, 2, 3,$

where the local system matrices  $A^j$ ,  $B^j$ ,  $F^j$  and  $C^j$  are determined in the same way with equation (44)-(47), and

the interaction terms are given by

$$B^{sj} = \begin{cases} \begin{bmatrix} A_{13} \\ 0 \end{bmatrix}, & j = 1, \\ \begin{bmatrix} A_{31} & A_{35} \\ 0 & 0 \end{bmatrix}, & j = 2, \\ \begin{bmatrix} A_{53} \\ 0 \end{bmatrix}, & j = 3. \end{cases} \quad (50)$$

$$y^{sj} = \begin{cases} y^2, & j = 1, 3 \\ \begin{bmatrix} y^1 \\ y^3 \end{bmatrix}, & j = 2, \end{cases} \quad (51)$$

where  $A_{ab}$  in equation (50) represents the row  $a$  column  $b$  of  $A$ . The prediction model for each local controller then can be constructed in a compact form

$$Y_k^j = \tilde{\Phi}^j x^j(k) + \tilde{\Gamma}^j (T u^{j*}(k-1) + H \Delta U_k^j) + \tilde{\Lambda}^j W(k) + \tilde{\Lambda}^{sj} Y_k^{sj}, \quad (52)$$

where  $u^*(k-1)$  is the optimal control at time instant  $k-1$ ,

$$Y_k^j = \begin{bmatrix} y_{1|k}^j & \cdots & y_{N|k}^j \end{bmatrix}^T, \quad (53)$$

$$\Delta U_k^j = \begin{bmatrix} \Delta u_{0|k}^j & \cdots & \Delta u_{N-1|k}^j \end{bmatrix}^T, \quad (54)$$

$$Y_k^{sj} = \begin{bmatrix} y^{sj}(k)^T & y_{2|k-1}^{sj}{}^T & \cdots & y_{N|k-1}^{sj}{}^T \end{bmatrix}^T, \quad (55)$$

and  $\tilde{\Phi}^j$ ,  $\tilde{\Gamma}^j$ ,  $\tilde{\Lambda}^j$  and  $\tilde{\Lambda}^{sj}$  are coefficient matrices consisting of  $A^j$ ,  $B^j$ ,  $F^j$ ,  $C^j$  and  $B^{sj}$ , and note that  $y_{1|k-1}^{sj}$  (the first element in  $Y_k^{sj}$ ) is replaced by measured output  $y^{sj}(k)$ .

Note that  $Y_k^{sj}$  is the future output prediction, which are the variables that the local controllers would send to and receive from the global communication network, and the expression in equation (55) is only used for the non-iteration DMPC or the first iteration of the multi-iteration DMPC. To converge to a Nash equilibrium point [15], the future prediction  $Y_k^{sj}$  is updated with an optimized control sequence at each iteration  $p$ , the  $Y_k^{j,p+1}$  is updated first as

$$Y_k^{j,p+1} = \tilde{\Phi}^j x^j(k) + \tilde{\Gamma}^j (T u^{j*}(k-1) + H \Delta U_k^{j,p*}) + \tilde{\Lambda}^j W(k) + \tilde{\Lambda}^{sj} Y_k^{sj,p}, \quad (56)$$

where  $\Delta U_k^{j,p*}$  is the optimized input sequence at iteration  $p$ , and  $Y_k^{sj,p+1}$  is then determined based on  $Y_k^{j,p+1}$

and their relation (51), so the  $Y_k^{sj,p+1}$  is iteratively updated as below:

$$Y_k^{sj,1} = \begin{bmatrix} y^{sj}(k)^T & (y_{2|k-1}^{sj})^T & \cdots & (y_{N|k-1}^{sj})^T \end{bmatrix}^T$$

$$Y_k^{sj,2} = \begin{bmatrix} y^{sj}(k)^T & (y_{1|k}^{sj,2})^T & \cdots & (y_{N-1|k}^{sj,2})^T \end{bmatrix}^T$$

$$Y_k^{sj,3} = \begin{bmatrix} y^{sj}(k)^T & (y_{1|k}^{sj,3})^T & \cdots & (y_{N-1|k}^{sj,3})^T \end{bmatrix}^T$$

$$\cdots$$

The distributed MPC problem of each local controller  $j$  can be defined as: At time  $k \in \mathbb{Z}^+$  and iteration  $p \in \mathbb{Z}^+$ , compute the input increments sequence by minimizing the local cost function

$$J^{j,p}(x^j(k), Y_k^{sj,p}, \Delta U_k^{j,p}) = (y^{j,p}(k) - r^j(k))^T Q (y^{j,p}(k) - r^j(k)) + (Y_k^{j,p} - R_k^j)^T \Omega (Y_k^{j,p} - R_k^j) + (\Delta U_k^{j,p})^T \Psi \Delta U_k^{j,p}, \quad (57)$$

$$s.t. \quad Y_k^{j,p} \in \mathbb{Y}^{q \times N} \subseteq \mathbb{R}^{q \times N}$$

$$\Delta U_k^{j,p} \in \Delta \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$$

$$U_k^{j,p} \in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N}$$

The iterative procedure for the distributed MPC within one time instant is provided in Algorithm 1 given the parameters  $\epsilon > 0$ ,  $p_{max} \in \mathbb{Z}^+$ ,  $w^j \in \mathbb{R}_{(0,1)}$ , and a stop condition based on 1-norm in the following form:

$$\|\Delta U_k^{j,p} - \Delta U_k^{j,p-1}\| \leq \epsilon \quad (58)$$

Fig. 18 shows a good tracking performance can be obtained with the penalty matrices  $Q_1 = 50e7$ ,  $R_1 = 3$ ;  $Q_2 = 50e6$ ,  $R_2 = 2$ ;  $Q_3 = 10e7$ ,  $R_3 = 5$ , even if the number of iterations is fixed to only 2. The convergence of the multiple iteration distributed algorithm is inspected by comparing the sum of three local cost function value with the cost function value of centralized MPC under the same penalty, i.e.,  $Q = blkdiag(50e7, 50e6, 10e7)$ ,  $R = blkdiag(3, 2, 5)$ . Fig. 19 and 20 show the convergence of the multi-iteration distributed algorithm at two time instants. It can be observed that a fast convergence is achieved.

Although the distributed algorithm does not converge to the global centralized solution exactly, the control performances from the distributed MPC (Fig. 18) are pretty good and close to the performances that the centralized MPC obtained in Fig. 8. The evolution of the value of the cost function for distributed (sum of all three local cost function) and centralized algorithms during the simulation period is also given in Fig. 21, and a good match can be observed.

From the computational point of view, the mean time

**Algorithm 1** iterative procedure for the distributed MPC within one time instant

- 1: Initialize the iteration counter  $p = 1$ .
- 2: Initialize the local inputs increments  $\Delta \hat{U}_k^{j,0} = [\Delta \hat{u}_{2|k-1}^{j*} \cdots \Delta \hat{u}_{N-1|k-1}^{j*} \Delta \hat{u}_{N-1|k-1}^{j*}]^T$ ,  $j = 1, 2, 3$ .
- 3: Obtain the current outputs and the estimated states from observer.
- 4: **while**  $\rho^j > \epsilon$  for some  $j = 1, 2, 3$  and  $p \leq p_{max}$  **do**
- 5: Construct the interaction vector  $Y_k^{sj}$  based on  $Y_k^j$ ,  $j = 1, 2, 3$  from the communication network
- 6:  $\forall j = 1, 2, 3$ , compute the optimized input increments by minimizing local cost function (57).
- 7: Let  $\Delta \hat{U}_k^{j,p}$  be the optimized sequence.
- 8: Set  $\Delta U_k^{j,p} = w^j \Delta \hat{U}_k^{j,p} + (1 - w^j) \Delta \hat{U}_k^{j,p-1}$
- 9: Set  $\rho^j = \|\Delta \hat{U}_k^{j,p} - \Delta \hat{U}_k^{j,p-1}\|$
- 10: Update  $Y_k^j$  based on  $\Delta U_k^{j,p}$  and send it to the communication network.
- 11: Increase the iteration counter by one:  $p = p + 1$ .
- 12: **end while**

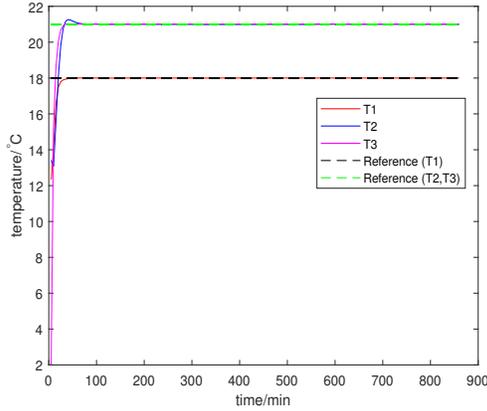


Fig. 18: Temperature response with distributed MPC

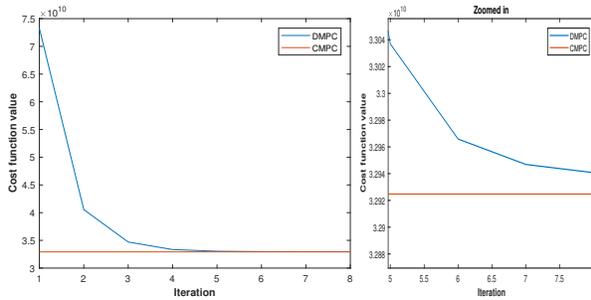


Fig. 19: Convergence of the distributed algorithm in instant 5th min

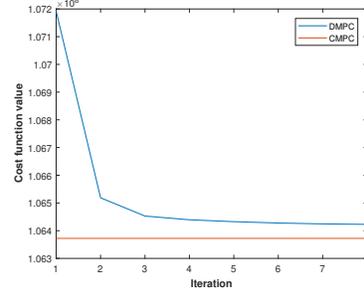


Fig. 20: Convergence of the distributed algorithm in instant 45th min

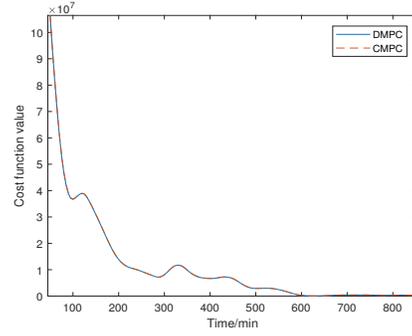


Fig. 21: Cost function evolution for centralized and distributed structures

spent by each distributed controller over 2 iterations to solve the local optimization problem is 2.72s using a Dual CPU at 2.50 GHz, while the centralized controller needs 4.81s. Note this difference is expected to be larger if more subsystems are included, since the computation load mainly depends on the number of manipulated variables. In general, the number of manipulated variables linearly grows with increase of subsystems, and the computational load grows even faster than linearly, while the number of manipulated variables is usually invariant for distributed control. Hence, the development of distributed MPC is strongly motivated.

### C. Multi-iteration distributed MPC with integral action

According to the results obtained in Section IV, MPC with integral action shows superiority with respect to offset rejection, so developing the distributed MPC algorithm with the combination of integral action is valuable.

Similar to Section IV, an enlarged plant is formed for

each subsystem as

$$x_I^j(k+1) = A_I^j x_I^j(k) + B_I^j \Delta u^j(k) + F_I^j \Delta w(k) + B_I^{sj} \Delta y^{sj}(k), \quad (59)$$

$$y^j(k) = C_I^j x_I^j(k), \quad (60)$$

$j = 1, 2, 3,$

where

$$\begin{aligned} x_I^j(k) &= \begin{bmatrix} \Delta x^j(k) \\ y^j(k) \end{bmatrix}, & A_I^j &= \begin{bmatrix} A^j & O_{n \times q} \\ C^j A^j & I_q \end{bmatrix}, \\ B_I^j &= \begin{bmatrix} B^j \\ C^j B^j \end{bmatrix}, & F_I^j &= \begin{bmatrix} F^j \\ C^j F^j \end{bmatrix}, \\ B_I^{sj} &= \begin{bmatrix} B^{sj} \\ C^j B^{sj} \end{bmatrix}, & C_I^j &= [O_{q \times n} \quad I_q]. \end{aligned}$$

The prediction model for the local controllers then can be constructed in the following form:

$$Y_k^j = \tilde{\Phi}_I^j x_I^j(k) + \tilde{\Gamma}_I^j \Delta U_k^j + \tilde{\Lambda}_I^j \Delta W_k + \tilde{\Lambda}_I^{sj} \Delta Y_k^{sj}, \quad (61)$$

where

$$Y_k^j = \begin{bmatrix} y_{1|k}^j & \cdots & y_{N|k}^j \end{bmatrix}^T, \quad (62)$$

$$\Delta U_k^j = \begin{bmatrix} \Delta u_{0|k}^j & \cdots & \Delta u_{N-1|k}^j \end{bmatrix}^T, \quad (63)$$

$$Y_k^{sj} = \begin{bmatrix} y^{sj}(k)^T & y_{2|k-1}^{sj}{}^T & \cdots & y_{N|k-1}^{sj}{}^T \end{bmatrix}^T, \quad (64)$$

$$\Delta Y_k^{sj} = \begin{bmatrix} y^{sj}(k) - y^{sj}(k-1) \\ y_{2|k-1}^{sj} - y^{sj}(k) \\ \vdots \\ y_{N|k-1}^{sj} - y_{N-1|k-1}^{sj} \end{bmatrix}, \quad (65)$$

and  $\tilde{\Phi}_I^j$ ,  $\tilde{\Gamma}_I^j$ ,  $\tilde{\Lambda}_I^j$  and  $\tilde{\Lambda}_I^{sj}$  are coefficient matrices consisting of  $A_I^j$ ,  $B_I^j$ ,  $F_I^j$ ,  $C_I^j$  and  $B_I^{sj}$ .  $\Delta Y_k^{sj}$  is constructed based on the  $Y_{k-1}^j$ ,  $y^j(k)$  and  $y^j(k-1)$  of the adjacent subsystem, which means more variables are required from the communication network compared with distributed MPC without integral action. For a multiple iteration distribute MPC,  $\Delta Y_k^{sj}$  is updated with optimized control sequence at each iteration  $p$ . First,  $Y_k^{j,p+1}$  is updated as

$$Y_k^{j,p+1} = \tilde{\Phi}_I^j x_I^j(k) + \tilde{\Gamma}_I^j \Delta U_k^{j,p*} + \tilde{\Lambda}_I^j \Delta W_k + \tilde{\Lambda}_I^{sj} \Delta Y_k^{sj,p}, \quad (66)$$

where  $\Delta U_k^{j,p*}$  is the optimized input sequence at iteration  $p$ , and  $Y_k^{sj,p+1}$  is determined next based on  $Y_k^{j,p+1}$  and their relation (51). Finally,  $\Delta Y_k^{sj,p+1}$  is iteratively

updated based on  $Y_k^{sj,p+1}$ ,  $y^{sj}(k)$  and  $y^{sj}(k-1)$  as below:

$$\begin{aligned} \Delta Y_k^{sj,1} &= \begin{bmatrix} y^{sj}(k) - y^{sj}(k-1) \\ y_{2|k-1}^{sj} - y^{sj}(k) \\ \vdots \\ y_{N|k-1}^{sj} - y_{N-1|k-1}^{sj} \end{bmatrix}, \\ \Delta Y_k^{sj,2} &= \begin{bmatrix} y^{sj}(k) - y^{sj}(k-1) \\ y_{1|k}^{sj,2} - y^{sj}(k) \\ \vdots \\ y_{N-1|k}^{sj,2} - y_{N-2|k}^{sj,2} \end{bmatrix}, \quad \dots \end{aligned}$$

The distributed MPC problem with integral action is defined as: At time  $k \in \mathbb{Z}^+$  and iteration  $p \in \mathbb{Z}^+$ , compute the input increments sequence by minimizing the local cost function

$$\begin{aligned} J^{j,p}(x^j(k), \Delta Y_k^{sj,p}, \Delta U_k^{j,p}) &= (y^{j,p}(k) - r^j(k))^T Q (y^{j,p}(k) - r^j(k)) \\ &+ (Y_k^{j,p} - R_k^j)^T \Omega (Y_k^{j,p} - R_k^j) + (\Delta U_k^{j,p})^T \Psi \Delta U_k^{j,p}, \end{aligned} \quad (67)$$

$$\begin{aligned} s.t. \quad Y_k^{j,p} &\in \mathbb{Y}^{q \times N} \subseteq \mathbb{R}^{q \times N} \\ \Delta U_k^{j,p} &\in \Delta \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N} \\ U_k^{j,p} &\in \mathbb{U}^{m \times N} \subseteq \mathbb{R}^{m \times N} \end{aligned}$$

The algorithm for distributed MPC with integral action is provided in Algorithm 2 given the parameters  $\epsilon > 0$ ,  $p_{max} \in \mathbb{Z}^+$  and  $w^j \in \mathbb{R}_{(0,1)}$ .

To validate the correctness of this algorithm, the distributed MPC designed in Section IV-B and distributed MPC with integral action are applied on a system with linear increased disturbance with small slope. Fig. 22 and 23 show that the offset is rejected by distributed MPC with integral action.

The convergence to the centralized MPC with integral action at one time instant can be observed in Fig. 24, and the evolution of the value of the cost function for distributed and centralized algorithms shows a good match during the simulation period (Fig. 25).

To summarize, decentralized MPC maybe an acceptable strategy when physical interaction between subsystems is insignificant, which is not the case for Hermitage museum, while distributed MPC is advocated to guarantee a good control performance even for the system with strong coupling, and a fast convergence to the centralized MPC is observed through communication. Slowly varying disturbance can also be rejected in the distributed MPC when integral action is employed. Note that integral action can be designed on certain zones that offset occurs, and the local MPC controller with integral

---

**Algorithm 2** Distributed MPC with integral action
 

---

- 1: Construct the augmented system matrices according to the equations (59)-(60).
  - 2: Compute the prediction matrices  $\tilde{\Phi}_I^j$ ,  $\tilde{\Gamma}_I^j$ ,  $\tilde{\Lambda}_I^j$  and  $\tilde{\Lambda}_I^{sj}$  for  $j = 1, 2, 3$ .
  - 3: Initialize the simulation counter  $k = 1$ .
  - 4: **while**  $k < k_{sim}$  **do**
  - 5:   Initialize the iteration counter  $p = 1$ .
  - 6:   Initialize the local inputs increments  $\Delta \hat{u}_{t|k}^{j,0} = \Delta \hat{u}_{t+1|k-1}^{j,*}$  for  $t = 1, \dots, N-1$  and  $j = 1, 2, 3$ .
  - 7:   Obtain the current output  $y^j(k)$  and the estimated states from observer,  $j = 1, 2, 3$
  - 8:   Send  $y^j(k)$  to the communication network for all  $j = 1, 2, 3$
  - 9:   Construct the coupling vector  $Y_k^{sj}$ ,  $y^{sj}(k)$  and  $y^{sj}(k-1)$  based on the information from the communication network,  $j = 1, 2, 3$ .
  - 10:   **while**  $\rho^j > \epsilon$  for some  $j = 1, 2, 3$  and  $p \leq p_{max}$  **do**
  - 11:     Update the coupling increments vector  $\Delta Y_k^{sj}$  based on  $Y_k^{sj}$ ,  $y^{sj}(k)$  and  $y^{sj}(k-1)$ ,  $j = 1, 2, 3$
  - 12:      $\forall j = 1, 2, 3$ , compute the optimized input increments by minimizing local cost function (67).
  - 13:     Let  $\Delta \hat{U}_k^{j,p}$  be the optimized sequence.
  - 14:     Set  $\Delta U_k^{j,p} = w^j \Delta \hat{U}_k^{j,p} + (1 - w^j) \Delta \hat{U}_k^{j,p-1}$
  - 15:     Set  $\rho^j = \|\Delta \hat{U}_k^{j,p} - \Delta \hat{U}_k^{j,p-1}\|$
  - 16:     Calculate the predicted output sequence  $Y_k^{j,*}$  with  $\Delta U_k^{j,p}$ ,  $j = 1, 2, 3$ .
  - 17:     Send  $Y_k^{j,*}$  to the communication network for all  $j = 1, 2, 3$ .
  - 18:     Update  $Y_k^{sj}$  based on  $y_k^j$  and  $Y_k^{j,*}$  from the communication network,  $j = 1, 2, 3$ .
  - 19:     Increase the iteration counter by one:  $p = p + 1$ .
  - 20:   **end while**
  - 21:   Increase the simulation counter by one:  $k = k + 1$ .
  - 22: **end while**
- 

action can cooperate with local MPC controller without integral action, which shows the high control flexibility in the distributed structure.

## VI. CONCLUSIONS

In this work, centralized and non-centralized identified model based predictive control is designed for the thermal control of Hermitage museum. A simplified grey-box model is developed to simulate temperature evolution of the museum, and its parameters are identified based on the data generated by HAMBASE SIMULINK tool. The results from model validation indicate that the

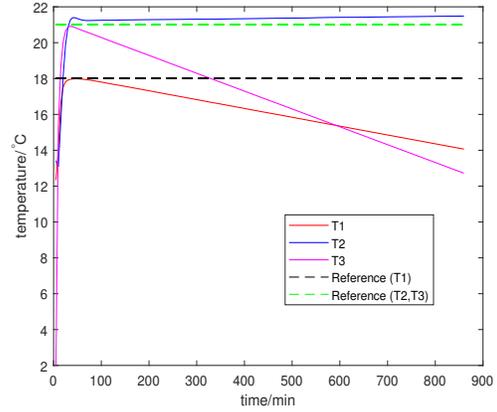


Fig. 22: Distributed MPC on the system with disturbance

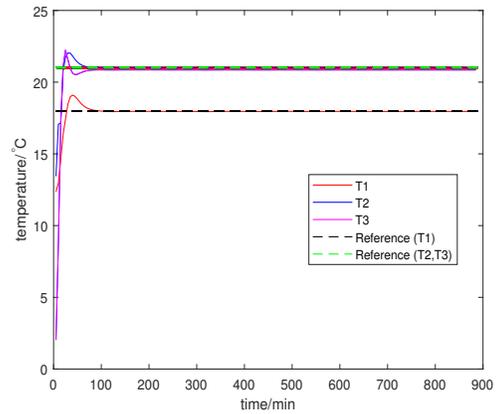


Fig. 23: Distributed MPC with integral action on the system with disturbance

presented model is capable of reproducing the thermal dynamics with sufficient accuracy and can be subsequently used for MPC design.

In the design of centralized MPC, two commonly used cost functions for reference tracking are investigated. The MPC with cost function that penalizes the distance to the reference states and inputs (Option 1) has risk of poor control performance when the model includes uncontrollable inputs that may considerably change within one prediction horizon, while the MPC that penalizes the deviation of the outputs from references and the increments of the inputs (Option 2) is more trustable to achieve a good tracking performance. Hence, the cost function is constructed in the form of Option 2 for the MPC design in the rest of this work. The centralized MPC with integral action is also designed to remove offsets with an augmented system. For validation pur-

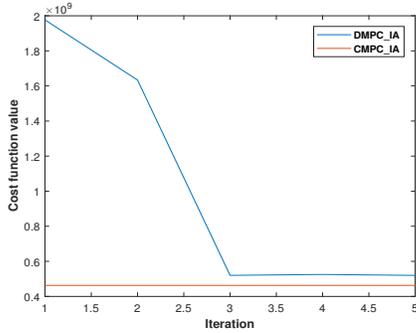


Fig. 24: Convergence of the distributed algorithm in instant 45th min

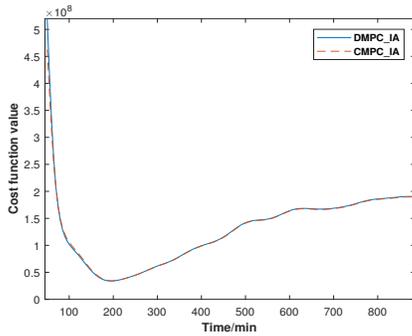


Fig. 25: Cost function evolution for centralized and distributed structures

pose, the standard MPC controller, MPC controller with integral action and a simple PI controller are applied on the HAMBASE SIMULINK tool. The simulation results imply that MPC with integral action shows advantage of offset rejection compared with standard MPC, and outperforms the conventional PI control with respect to energy saving.

Although the centralized MPC can guarantee a good control performance, the computational load for centralized algorithms exponentially increases with the system size. The non-centralized MPC algorithms, which can be categorized as decentralized MPC and distributed MPC, are designed for the reduction of computational load. The simulation results show that decentralized MPC fails to achieve a good control performance for the thermal system of Hermitage museum with significant coupling, while distributed MPC guarantees a good tracking performance and requires less computational time compared with centralized MPC, and a higher control flexibility is also expected in the distributed structure. The convergence to the centralized MPC is observed along iterations of the distributed MPC. Furthermore, the algorithm of distributed MPC with integral action is developed for

offset rejection, and the convergence to the centralized MPC with integral action is also observed.

#### ACKNOWLEDGMENT

The author would like to thank dr. Mircea Lazar and dr. Jobert H.A. Ludlage for their guidance and patience during this work, the results obtained in this thesis would have not been possible without their constant help. The author also would like to thank dr. Rick Kramer for providing the HAMBASE SIMULINK model and his valuable suggestions on the simplified model development.

#### REFERENCES

- [1] K. Kompatscher, S. Seuren, R. Kramer, J. Van Schijndel, and H. Schellen, "Energy efficient HVAC control in historical buildings: A case study for the Amsterdam Museum," *Energy Procedia*, vol. 132, pp. 891–896, 2017.
- [2] M. Martens, "Climate risk assessment in museums: degradation risks determined from temperature and relative humidity data," *Building and Environment*, vol. 2005, no. 2012, pp. 41–44, 2012.
- [3] S. Thomas, "Drivers of recent energy consumption trends across sectors in EU28," Tech. Rep. May, European Commission, Directorate General for Energy, Energy Efficiency Unit, 2018.
- [4] I. Kaya, N. Tan, and D. P. Atherton, "Improved cascade control structure for enhanced performance," *Journal of Process Control*, vol. 17, no. 1, pp. 3–16, 2007.
- [5] Z. Jun and Z. Kanyu, "A particle swarm optimization approach for optimal design of PID controller for temperature control in HVAC," *Proceedings - 3rd International Conference on Measuring Technology and Mechatronics Automation, ICMTMA 2011*, vol. 1, no. 2, pp. 230–233, 2011.
- [6] A. Afram and F. Janabi-Sharifi, "Theory and applications of HVAC control systems - A review of model predictive control (MPC)," *Building and Environment*, vol. 72, pp. 343–355, 2014.
- [7] M. Xu and S. Li, "Practical generalized predictive control with decentralized identification approach to HVAC systems," *Energy Conversion and Management*, vol. 48, pp. 292–299, 2007.
- [8] J. Taneja, S. M. Ieee, D. Culler, F. Ieee, C. Tomlin, and F. Ieee, "Reducing Transient and Steady State Electricity Consumption in HVAC Using Learning-Based Model-Predictive Control," *Proceedings of the IEEE*, vol. 100, no. 1, pp. 240–253, 2012.
- [9] G. Huang, "Model predictive control of VAV zone thermal systems concerning bi-linearity and gain nonlinearity," *Control Engineering Practice*, vol. 19, no. 7, pp. 700–710, 2011.
- [10] A. Beghi, L. Cecchinato, F. Paggiaro, and M. Rampazzo, "VAVAC Systems Modeling and Simulation for FDD Applications," *2011 9th IEEE International Conference on Control and Automation (ICCA)*, pp. 800–805, 2011.
- [11] T. Baumelt, "Distributed Building Identification," Tech. Rep. May, Czech Technical University in Prague, 2016.
- [12] C. Yiu, *Statistical modelling and forecasting schemes for air-conditioning system*. PhD thesis, The Hong Kong Polytechnic University, 2008.
- [13] V. Chandan and A. G. Alleyne, "Optimal Control Architecture Selection for Thermal Control of Buildings," *Proceedings of the 2011 American Control Conference*, pp. 2071–2076, 2011.
- [14] P. D. Moroşan, R. Bourdais, D. Dumur, and J. Buisson, "Building temperature regulation using a distributed model predictive control," *Energy and Buildings*, vol. 42, no. 9, pp. 1445–1452, 2010.
- [15] S. Li, Y. Zhang, and Q. Zhu, "Nash-optimization enhanced distributed model predictive control applied to the Shell benchmark

problem,” *Information Sciences*, vol. 170, no. 2-4, pp. 329–349, 2005.

- [16] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright, “Distributed MPC strategies for automatic generation control,” pp. 383–388, IFAC Symposium on Power Plants and Power Systems Control, 2006.
- [17] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright, “Distributed MPC Strategies With Application to Power System Automatic Generation Control,” *IEEE Transactions on Control Systems Technology*, vol. 16, no. 6, pp. 1192–1206, 2008.
- [18] C. Agbi, Z. Song, and B. Krogh, “Parameter identifiability for multi-zone building models,” in *Proceedings of the IEEE Conference on Decision and Control*, pp. 6951–6956, 2012.
- [19] R. Kramer, J. van Schijndel, and H. Schellen, “Inverse modeling of simplified hygrothermal building models to predict and characterize indoor climates,” *Building and Environment*, vol. 68, pp. 87–99, 2013.
- [20] T. McKelvey and A. Helmersson, “A dynamical minimal parametrization of multivariable linear systems and its application to optimization and system identification,” in *14th World Congress of IFAC of IFAC*, pp. 3778–3783, Elsevier, 1999.
- [21] M. de Wit, *Hambase: heat, air and moisture model for building and systems evaluation*, vol. 100. Eindhoven University Press, Eindhoven, the Netherlands, 2019.
- [22] A. W. M. Schijndel, van, *Integrated heat air and moisture modeling and simulation*. No. 2007, Eindhoven University Press, Eindhoven, the Netherlands, 2019.
- [23] S. S. Walker, W. Lombardi, S. Lesecq, and S. Roshany-Yamchi, “Application of Distributed Model Predictive Approaches to Temperature and CO2 Concentration Control in Buildings,” *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 2589–2594, 2017.
- [24] S. Yuan and R. Perez, “Multiple-zone ventilation and temperature control of a single-duct VAV system using model predictive strategy,” vol. 38, pp. 1248–1261, 2006.
- [25] J. Ma, J. Qin, T. Salsbury, and P. Xu, “Demand reduction in building energy systems based on economic model predictive control,” *Chemical Engineering Science*, vol. 67, no. 1, pp. 92–100, 2012.
- [26] R. Kramer, A. van Schijndel, and H. Schellen, “The importance of integrally simulating the building, HVAC and control systems, and occupants’ impact for energy predictions of buildings including temperature and humidity control,” *Journal of Building Performance Simulation ISSN:*, vol. 10, pp. 272–293, 2017.
- [27] A. Thavlov and H. W. Bindner, “Thermal Models for Intelligent Heating of Buildings,” *Proceedings of the International Conference on Applied Energy, ICAE 2012*, p. A10591, 2012.
- [28] M. Y. Lamoudi, *Distributed model predictive control for energy management in buildings*. PhD thesis, Université de Grenoble, 2013.
- [29] L. Ljung, *System Identification Theory for User*, vol. 25. PTR Prentice Hall Upper Saddle River NJ, 1987.
- [30] U. Maeder, F. Borrelli, and M. Morari, “Automatica Linear offset-free Model Predictive Control,” *Automatica*, vol. 45, no. 10, pp. 2214–2222, 2009.
- [31] D. D. Ruscio, “Model Predictive Control with Integral Action : A simple MPC algorithm,” *Modeling, Identification and Control*, vol. 34, no. 3, pp. 119–129, 2013.

#### APPENDIX A

##### EXPRESSIONS OF PARAMETERS IN THE STATE SPACE MODEL

According to the differential equations derived in Section III-A, the expressions of parameters in the state

space model are given as

$$\begin{aligned}\theta_1 &= -\frac{1}{C_1}\left(\frac{1}{R_{fa1}} + \frac{1}{R_{int1}} + \frac{1}{R_{w2}}\right), \\ \theta_2 &= \frac{1}{C_1 R_{int1}}; \theta_3 = \frac{1}{C_1 R_{w2}}; \theta_4 = \frac{1}{C_{int1} R_{int1}}; \\ \theta_5 &= -\frac{1}{C_{int1} R_{int1}}; \theta_6 = \frac{1}{C_2 R_{w2}}; \\ \theta_7 &= -\frac{1}{C_2}\left(\frac{1}{R_{w2}} + \frac{1}{R_{int2}} + \frac{1}{R_{fa2}} + \frac{1}{R_{w3}}\right); \\ \theta_8 &= \frac{1}{C_2 C_{int2}}; \theta_9 = \frac{1}{C_2 R_{w3}}; \theta_{10} = \frac{1}{C_{int2} R_{int2}}; \\ \theta_{11} &= -\frac{1}{C_{int2} R_{int2}}; \theta_{12} = \frac{1}{C_3 R_{w3}}; \\ \theta_{13} &= -\frac{1}{C_3}\left(\frac{1}{R_{fa4}} + \frac{1}{R_{int3}} + \frac{1}{R_{w3}}\right); \\ \theta_{14} &= \frac{1}{C_3 R_{int3}}; \theta_{15} = \frac{1}{C_{int3} R_{int3}}; \\ \theta_{16} &= -\frac{1}{C_{int3} R_{int3}}; \theta_{17} = \frac{1}{C_1}; \theta_{18} = \frac{1}{C_2}; \theta_{19} = \frac{1}{C_3}; \\ \theta_{20} &= \frac{1}{C_1 R_{fa1}}; \theta_{21} = \frac{fI_1}{C_{int1}}; \theta_{22} = \frac{1}{C_2 R_{fa2}}; \\ \theta_{23} &= \frac{fI_2}{C_{int2}}; \theta_{24} = \frac{1}{C_3 R_{fa3}}; \theta_{25} = \frac{fI_3}{C_{int3}};\end{aligned}$$

#### APPENDIX B

##### GRAPHS OF MODEL VALIDATION

The fitting performance of the identified model for graphical analysis is given in Fig. 26. The results of residual correlation test based on 99% confidence interval is shown in Fig. 27.

#### APPENDIX C

##### TEST FOR INVESTIGATION OF THE COMPUTATION OF $X_k^{ss}$ AND $U_k^{ss}$

As it is indicated in Section IV-A, the  $X_k^{ss}$  and  $U_k^{ss}$  that computed based on equation (27) maybe unreasonable if the considerably changes occur in the uncontrollable input of the system, so the MPC that penalize these unreasonable  $X_k^{ss}$  and  $U_k^{ss}$  could result in terrible response. This section investigate the computation of  $X_k^{ss}$  and  $U_k^{ss}$  under the influence of uncontrollable input  $w(k)$ . For the convenience of illustration, a simpler double integrator with additional uncontrollable input is used, i.e,

$$\begin{aligned}x(k+1) &= Ax(k) + Bu(k) + w(k), \\ y(k) &= Cx(k),\end{aligned}\tag{68}$$

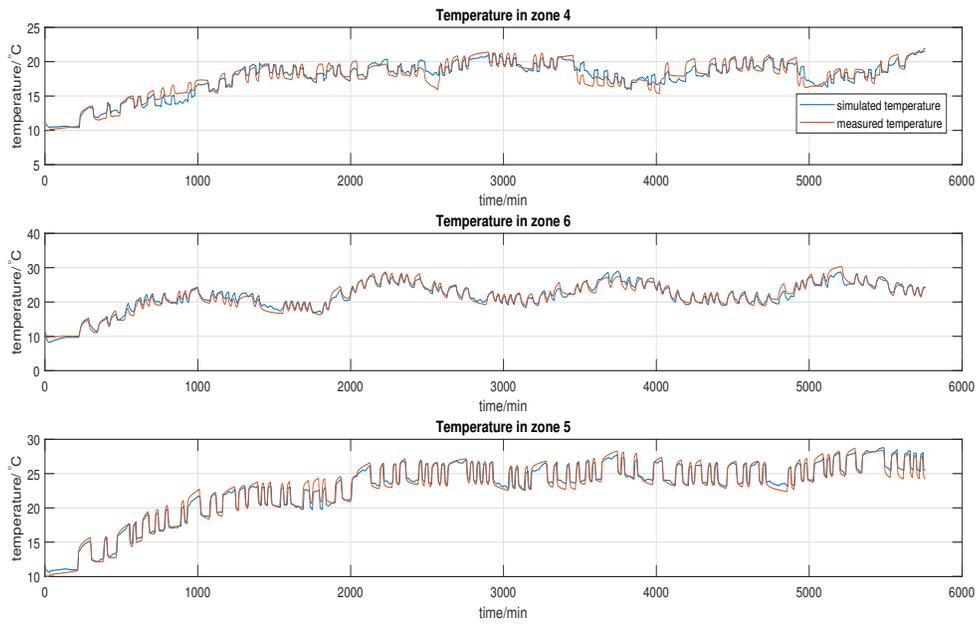


Fig. 26: Graphical analysis of the identified model.

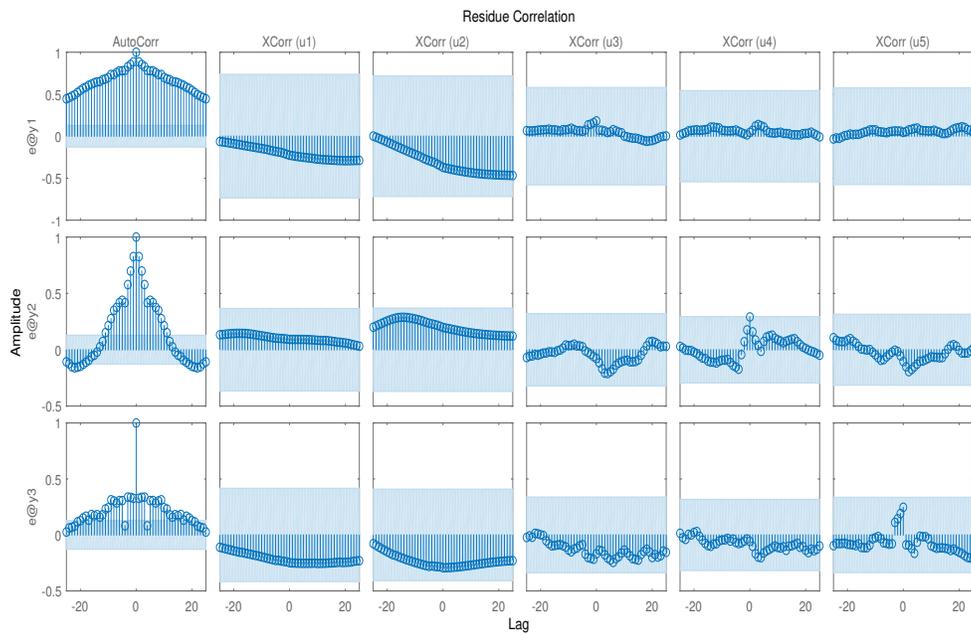


Fig. 27: Residual correlation test of the identified model.

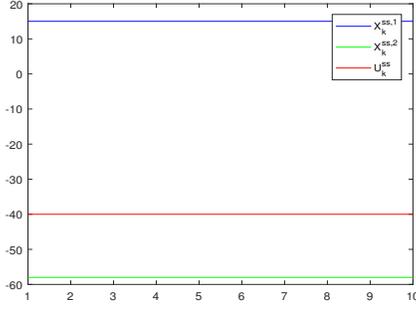


Fig. 28: Reference state and input based on static  $W_k$

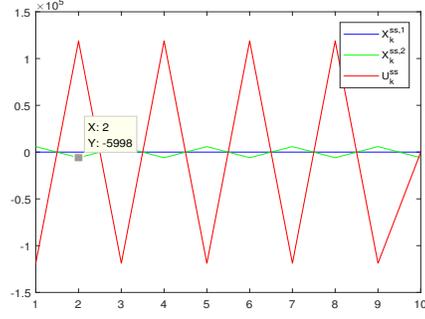


Fig. 29: Reference state and input based on abruptly changed  $W_k$

where

$$A = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix}, B = \begin{bmatrix} 0.5T_s^2 \\ T_s \end{bmatrix}, C = [1 \quad 0], T_s = 0.1. \quad (69)$$

The reference  $R_k$  is also considered as static sequence (e.g.,  $r_{i|k} = 15, \forall j = 1, 2, \dots, N$ ), and the prediction horizon  $N$  is set as 10. The computation of  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27) is checked by tuning of  $W_k$ .

(i). Consider  $w(k)$  is static in the prediction horizon, e.g.,

$$W_k = \begin{bmatrix} 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \end{bmatrix}$$

The computed  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27) is depicted in Fig. 28. It can be observed that the calculated reference state and input are all constant.

(ii). Assume that an abrupt change occurs on  $w(k)$  at the end of prediction horizon, e.g.,

$$W_k = \begin{bmatrix} 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 6 & 600 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \end{bmatrix}$$

The computed  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27) is depicted in Fig. 29. It can be observed that the calculated reference state and input show zigzag dynamics with quite large varying range.

(iii). To discover the dynamics in Fig. 29 are resulted by the abrupt change of  $w(k)$  or the large value of  $w(k)$ , assume  $W_k$  consisting of slowly changed large values, e.g.,

$$W_k = \begin{bmatrix} 591 & 592 & 593 & 594 & 595 & 596 & 597 & 598 & 599 & 600 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \end{bmatrix}$$

Fig. 30 shows the computed  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27). It can be observed that the calculated reference state and input does not violently vary, although  $W_k$  consist of large values.

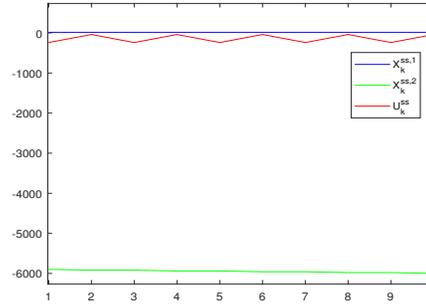


Fig. 30: Reference state and input based on  $W_k$  with large values

(iv). Assume that an abrupt change occurs on  $w(k)$  at the middle of prediction horizon, e.g.,

$$W_k = \begin{bmatrix} 3 & 4 & 5 & 6 & 600 & 7 & 8 & 9 & 10 & 11 \\ 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 & 4 \end{bmatrix}$$

Fig. 31 shows the computed  $X_k^{ss}$  and  $U_k^{ss}$  based on equation (27). It can be observed that the calculated reference state and input dramatically vary before the abrupt change occurs on  $w(k)$ .

To summarize, the reference state and input that computed based on equation (27) would show zigzag dynamics with huge varying range if abrupt changes occur in the uncontrollable input.

## APPENDIX D

### CONTROL PERFORMANCE OF MPC UNDER INFLUENCE OF MODEL ACCURACY

In this work, a thermal model with a slightly different structure is also developed and identified. The fitting performance to the experimental data is given in Fig. 33, and the MSE for zone 4,5 and 6 are 1.0838, 1.8276 and 1.7124, respectively. Although this model is relatively

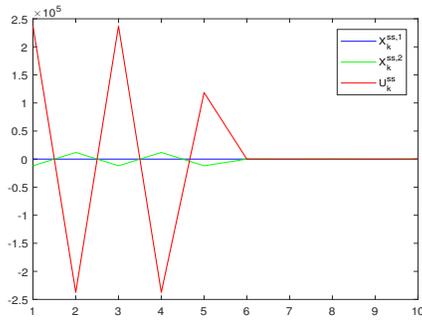


Fig. 31: Reference state and input based on  $W_k$  with abrupt change at the middle of prediction horizon

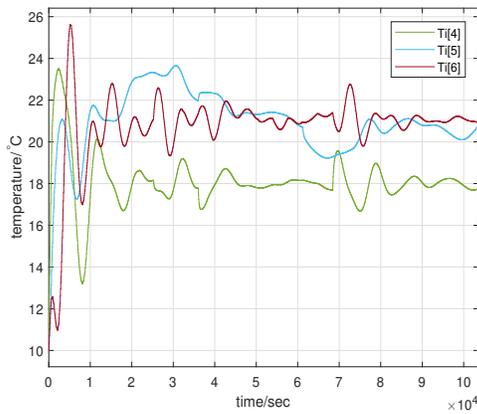


Fig. 32: Control performance achieved by centralized MPC with integral action based on the second model

inaccurate compared with the model identified in Section III, it can be observed that most of thermal dynamics are also reproduced. Compared with the MSE of the model identified in Section III, the MSE of this fit are larger, but not totally unreasonable.

However, the centralized MPC based on this model achieves a poor performance on HAMBASE SIMULINK model, and this poor performance cannot be improved by integral action. Fig. 32 shows the temperature responses achieved by the centralized MPC with integral action. It can be concluded that the control performance of MPC is sensitive to the accuracy of the model identification.

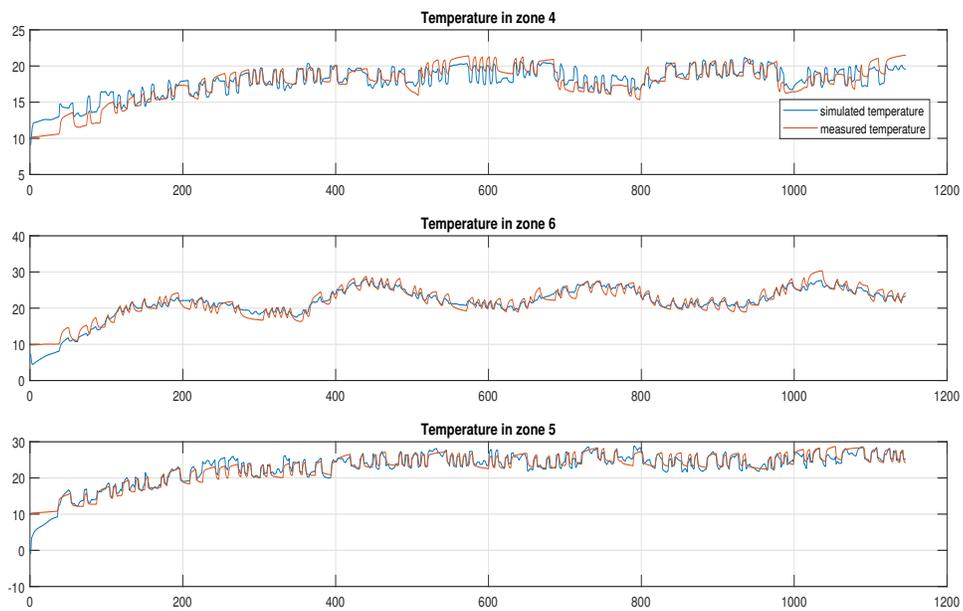


Fig. 33: Graphical analysis of the second model.