Distributed and cooperative control with applications to building HVAC systems

Long, Yushen

2017

Long, Y. (2017). Distributed and cooperative control with applications to building HVAC systems. Doctoral thesis, Nanyang Technological University, Singapore.

<http://hdl.handle.net/10356/69494>

<https://doi.org/10.32657/10356/69494>

Distributed and Cooperative Control With Applications to Building HVAC Systems

Long Yushen

School of Electrical & Electronic Engineering

A thesis submitted to the Nanyang Technological University
in fulfillment of the requirements for the degree of
Doctor of Philosophy

2016

Statement of Originality

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other University or Institution.

.....

Date

.....

Long Yushen

Acknowledgements

First, I owe my deepest gratitude to my supervisor, Professor Xie Lihua, for his invaluable guidance and continuous encouragement during my Ph.D. studies. It was his insightful guidance that brought me into such an interesting research field and enabled me to develop an understanding of the subject. What I appreciated most is the many opportunities and the freedom he has offered that allowed me to find the research topics I am most interested in. Without his great enthusiastic help, this thesis would not have been possible.

I also would like to thank Professor Karl Henrik Johansson (Royal Institute of Technology, Sweden) and Dr. Liu Shuai for their collaborations and fruitful discussions. Their insightful comments and suggestions have greatly inspired my research.

Thanks also go to the postgraduates, staff members and visiting scholars at Internet of Things Lab, Nanyang Technological University and other friends who helped and supported me in these years.

I must also acknowledge the research scholarship I received from the Singapore-Berkeley Building Efficiency and Sustainability in the Tropics (SinBerBEST) Program.

Finally, I would like to thank my parents for their eternal love.

Abstract

According to the environment programme report [1] published by the United Nations, buildings account for 40% of energy consumption and resources and one third of greenhouse gas emissions. Thus to improve the energy efficiency of buildings becomes more and more attractive. Heating, ventilation and air-conditioning (HVAC) system composes one third to half of building energy use [2] so to optimize the energy efficiency of it is one of the most promising directions. To this end, developing energy-efficient control strategies for building HVAC systems is necessary for both economy and environment protection.

Model Predictive Control (MPC) has attracted more and more attention to improve the energy efficiency of HVAC systems. Because of its optimality nature and ability to handle input and output constraints, MPC usually can achieve better performance compared with traditional control strategies. In recent years, a lot of research works have been published on using MPC approaches to reduce the energy cost of HVAC systems while keeping the temperature in a comfortable region, see, e.g. [3], [4] and so on.

There are two difficulties when MPC is applied to building HVAC systems. The first one comes from the large scale of an HVAC system in a building. The number of states of the system depends on the number of zones whose temperatures need to be maintained in a comfortable region. The number of decision variables in the corresponding optimization problems for MPC grows fast when a long prediction horizon is considered. Therefore, the optimization problem which should be solved

in every sample interval becomes large and even computationally intractable. A traditional centralized controller is no longer suitable for building HVAC systems. The other difficulty is uncertainties such as weather condition and occupancy. To use model predictive control, one needs to predict evolution of the system to find the optimal control input. However, because of the uncertainties, it is impossible to get a precise prediction. It has been pointed out in [5] and [6] that the robustness of model predictive control is a non-trivial topic. Even arbitrarily small disturbance may destabilize a nominally stable system. Therefore, it is necessary to propose some novel strategies which are robust with respect to external disturbance and/or model uncertainty.

In this thesis, we present research results on the design of distributed and robust MPC with their applications on building HVAC systems. First, we introduce a scenario-based distributed stochastic MPC algorithm. We consider the weather condition and occupancy as stochastic disturbance and use probabilistic constraints to confine the temperature of each zone within a comfortable region with a pre-defined probability. Since the evaluation of the probabilistic constraints requires the computation of multi-integral, the resulted stochastic optimization problem becomes intractable. Furthermore, the distributions of the weather condition and the occupancy are hard to model. To overcome these issues, a scenario-based approach is used to approximate the probabilistic constraints by a group of deterministic ones such that the stochastic optimization problem is approximated by a deterministic one. After that, the distribution of the uncertainties is not required any more. A Lagrange dual problem is formulated and it is decomposed by taking advantage of its separable structure. A sub-gradient algorithm is used to solve the dual problem in a distributed way.

Though the scenario-based approach can handle stochastic uncertainty efficiently, it is hard to have guarantee on recursive feasibility and convergence. To overcome this issue, we propose a distributed robust MPC based on reduced order models. Considering the complex nature of the heat transfer process, first order model usually is over simplified for a zone with furniture and walls which have different heat

capacity. A model based on computational fluid dynamics, though it is accurate enough and widely used in civil and environmental engineering [7], [8], is too complex for controller design and analysis. In the proposed algorithm, the off-line design is based on higher order model such that the controller can capture the system property as much as possible. In the on-line implementation phase, the optimal control problem is based on a reduced order model so that the computational burden is alleviated and the implementation becomes practical. The modelling error introduced by the model reduction is considered as a bounded disturbance and a tube-based approach is used to shape the system behaviour. The on-line optimization generates an open-loop control action which steers the nominal system to the equilibrium while a linear feedback controller which is designed off-line is used to compensate the uncertainties.

After introducing two MPC based on linear discrete time models, we present research results based on nonlinear continuous-time models. Though convex optimization problems can be formulated by using linear discrete time models and efficient numerical solvers are available, system behaviour between sampling intervals cannot be handled and the nonlinearity of the building system is not considered. On the other hand, due to the development of nonlinear programming [9], nonlinear optimal control problem is no longer intractable. Thus, to use nonlinear continuous-time model provides more potentials on the improvement of the system performance without too much additional computational burden.

By using nonlinear continuous-time model, we first present a distributed MPC algorithm based on contraction property. Most of the existing nonlinear continuous-time MPC is based on the Lipschitz constant of the system. Due to the inherent conservativeness, the resulted algorithms are usually impractically conservative. To overcome this issue, we consider a special class of nonlinear systems with contraction property [10]. Instead of using the Lipschitz constants of the local systems, contraction coefficients are used to estimate the prediction error, which greatly improve the conservativeness. Sufficient conditions on feasibility and stability are also derived.

Next, we apply the proposed contraction theory based distributed MPC to building HVAC systems. We design a hierarchical structure to handle the system in two levels: the building level and the zone level. The upper layer collects the global information which includes the temperature measurements of each room, the occupancy and the weather prediction, and it generates temperature and control references for each zone. The lower layer, on the contrary, only makes use of local information and the reference signal to optimize the local performance. Two layers work with different frequencies. The upper layer updates slowly with a long prediction horizon while the lower layers update more frequently with a short prediction horizon.

In the previous control schemes we assume perfect communication between controllers and sensors/actuators. However, in smart buildings, digital wireless communication is widely used and channel capacity is limited so that perfect communication is impossible. Therefore, we also study a networked MPC design problem. To equip the valves and the dampers of HVAC systems with advanced controllers is not practical because of the limited space of duct. A more practical strategy is that control signals are computed by remote controllers and transmitted to the actuators which can adjust the valves and dampers. Due to the limited bandwidth of digital communication channel, we study the quantization effect on the control signals and the state measurements. Sufficient conditions are provided to guarantee feasibility and stability.

Symbols and Acronyms

Symbols

$\mathbb{R}$	set of real numbers
$\mathbb{R}^N$	set of N dimensional real vectors
$\mathbb{R}^{m \times n}$	$m \times n$ dimensional Euclidean space
$\mathbb{N}$	set of natural numbers
$\mathbb{Z}^N$	set of N dimensional integer vectors
A^T	transpose of A
$A_{i,j}$	i, j th block of A
$\ x\ $	2-norm of vector x
$\ x\ _P$	P-norm of vector x
$A \oplus B$	Minkowski sum of sets A and B
$A \ominus B$	Pontryagin difference of sets A and B
$s[a : b]$	$(s_a, s_{a+1}, \dots, s_b)$ for discrete signal s_k
$\text{box}(\mathbb{X})$	the smallest hyperrectangle containing $\mathbb{X}$ with faces perpendicular to the cartesian axis
$\lambda_{\max}(A)$	the largest eigenvalue of a positive definite matrix A
$\lambda_{\min}(A)$	the smallest eigenvalue of a positive definite matrix A

Acronyms

AC	Air Conditioning
----	------------------

ACB	Active Chilled Beam
AHU	Air Handling Unit
DLC	Direct Load Control
GPC	Generalized Model Predictive Control
HVAC	Heating, Ventilation and Air-conditioning
LDDS	Liquid Desiccant Dehumidification System
LMI	Linear Matrix Inequality
LTI	Linear Time-Invariant
MIMO	Multi-Input Multi-Output
MPC	Model Predictive Control
NLP	Nonlinear Programming
PI	Proportional Integral
PID	Proportional Integral Derivative
PWM	Pulse-Width Modulation
RC	Resistive-Capacitive
RMPC	Randomized Model Predictive Control
SMPC	Stochastic Model Predictive Control
VAV	Variable Air Volume

Contents

Acknowledgements	i
Abstract	iii
Symbols and Acronyms	vii
List of Contents	ix
List of Figures	xiii
1 Introduction	1
1.1 Motivation and Objectives	1
1.2 Main Contributions	3
1.3 Outline of the Thesis	4
2 Literature Review	7
2.1 HVAC System Modelling	7
2.2 Control Strategies for HVAC Systems	11
2.2.1 Conventional Control	11
2.2.2 Adaptive Control	12
2.2.3 Robust Control	14
2.2.4 Model Predictive Control	14
2.3 Conclusion	25

3	A Scenario-Based Distributed Stochastic MPC for Building Temperature Regulation	27
3.1	System Modeling	28
3.2	Randomized MPC formulation	30
3.3	Distributed optimization algorithm	35
3.4	Simulation results	38
3.5	Conclusions	40
4	Distributed Non-Cooperative Robust MPC Based on Reduced-Order Models	45
4.1	Problem Formulation	46
4.2	Controller Design	47
4.2.1	Bounded Unmeasurable State	47
4.2.2	Distributed Robust MPC	48
4.3	Implementation	51
4.3.1	Feedback Gains and Weighted Matrices	52
4.3.2	Computation of Sets	52
4.3.3	Initial Feasible Reference Trajectories	53
4.4	Numerical Example	54
4.5	Conclusion	58

5	Distributed Nonlinear Model Predictive Control Based on Contraction Theory	63
5.1	Problem Formulation	64
5.2	Controller Design and Implementation	66
5.2.1	Offline design	66
5.2.2	Initialization	67
5.2.3	Online implementation	68
5.3	Analysis	70
5.3.1	Feasibility	70
5.3.2	Stability	74
5.4	Numerical Example	77
5.5	Conclusion	81
6	A Hierarchical Distributed MPC for HVAC Systems	83
6.1	Controller design	84
6.2	Simulation and Discussion	90
6.3	Conclusion	93
7	Quantized Self-Triggered MPC	97
7.1	Problem formulation	98
7.1.1	System Definition	98
7.1.2	Quantizer	99
7.2	Self-triggered MPC design	100
7.2.1	Optimal control problem	100
7.2.2	Self-triggered condition	101
7.3	Analysis	102
7.3.1	Feasibility	102
7.3.2	Stability	105
7.4	Numerical Examples	108
7.5	Conclusion	111

8	Conclusions and Future Work	113
8.1	Conclusion	113
8.2	Future Work	115
	Author's Publications	117
	Bibliography	119

List of Figures

2.1	Schematic diagram of the HVAC system	8
2.2	Schematic diagram of AHU	8
2.3	Schematics of on-off control	11
2.4	Principle of MPC	15
3.1	Building configuration	40
3.2	Control profile under centralized MPC	41
3.3	Indoor temperature controlled by centralized MPC	41
3.4	Indoor temperature controlled by distributed MPC with fully connected topology	42
3.5	Indoor temperature controlled by distributed MPC with simplified topology	42
3.6	Indoor temperature controlled by PID	43
3.7	Control profile generated by decentralized PID	43
4.1	Schematic representation of a building with two apartments	58
4.2	State and input trajectory of distributed algorithm. Black dashed lines represent room A and C while gray solid lines represent room B and D.	59

4.3	State and input trajectory of decentralized algorithm. Black dashed lines represent room A and C while gray solid lines represent room B and D.	59
4.4	State and input trajectory of centralized full order algorithm. Black dashed lines represent room A and C while gray solid lines represent room B and D.	60
4.5	Temperature deviation between distributed (black dashed line) and decentralized (gray solid line) algorithm (room A and B). In this example, the air-conditioning system is working on cooling mode. Therefore the distributed algorithm has faster response than the decentralized one since the former one has lower temperature.	60
4.6	Temperature deviation between distributed (black dashed line) and decentralized (gray solid line) algorithm (room C and D)	61
4.7	Temperature deviation between distributed (black dashed line) and centralized (gray solid line) algorithm (room A and B)	61
4.8	Temperature deviation between distributed (black dashed line) and centralized (gray solid line) algorithm (room C and D)	62
5.1	Initial feasible state trajectory and control profile	77
5.2	Comparison of the state trajectories and control profiles under centralized and distributed algorithms	78
5.3	State trajectories and control profiles under decentralized PI controller	79
6.1	Control scheme	86
6.2	Environment temperature	92
6.3	Thermal load	92
6.4	Temperature response and control input under the proposed algorithm	94

6.5	Temperature response and control input under decentralized PI algorithm	95
7.1	Scheme of the networked control system	98
7.2	Comparison of the state trajectories under different control schemes .	109
7.3	Comparison of the control inputs under different control schemes . . .	110
7.4	Comparison of the state trajectories under different control schemes .	111
7.5	Comparison of the control inputs under different control schemes . . .	112

Chapter 1

Introduction

1.1 Motivation and Objectives

A number of reports have discussed energy consumption problems in buildings and especially the HVAC systems in buildings. According to an environment programme report [1] by the United Nations , 40% of energy consumption and resources and one third of greenhouse gas emissions are caused by buildings. According to [11] and [12], buildings account for 76% and 73% of electricity consumption in Europe and the United States. In the United States, nearly 10% of greenhouse gas emissions and 25% of the electricity is due to the HVAC systems in buildings [13]. In Singapore, more than 30% of energy is consumed by buildings while air-conditioning systems are responsible for more than half of the total energy consumed by buildings [14]. Therefore, to reduce the energy cost of buildings becomes more and more attractive and one of the most promising directions is to improve the energy efficiency of HVAC systems.

There are two possible ways to achieve this goal. The first one is to design novel HVAC devices with more efficient mechanical structures. For example, ACB is a kind of novel HVAC structure to substitute the traditional VAV systems [15, 16]. However, it will cost much time and money to reinstall new HVAC devices in existing

buildings. Thus, our research works mainly focus on the second way, which is to design novel control algorithms, to improve the efficiency of the HVAC systems.

To design novel control algorithms for HVAC systems, several problems should be considered. First of all, the indoor environment condition should be kept in a certain comfortable region. In our research works, we mainly focus on the indoor temperature and the first aim of the control algorithm we will design is to guarantee that the indoor temperature is kept in a predefined region. Secondly, the energy consumption should be reduced by using those novel control algorithms. Furthermore, it should be noticed that there exist many physical constraints because of the mechanical structures such as the limited air flow rate, the maximal cooling and heating power, the minimal ventilation level and so on.

To handle the mentioned issues, MPC could be a suitable choice. The control signal of MPC is generated by repeatedly solving a constrained finite horizon optimal control problem based on the measurement of the system. Therefore, the energy efficiency could be contained in the objective function explicitly. Besides that, the physical limitations of the valves and dampers can be expressed as the constraints of the optimization problem as well.

As we know, a common commercial building usually contains more than ten stories and hundreds of rooms or zones and the resulted dynamic system will contain a large number of states which makes the implementation and maintenance of a centralized controller difficult. As the dimension the system grows, the optimization problem of MPC may become difficult to solve. On the other hand, due to the development of network technique, distributed control based on sensor and actuator network has attracted a lot of attentions in recent years. Unlike those centralized control strategy, distributed control is to use a group of controllers each of which only takes care of a local subsystem and cooperates with each other to handle the whole large system. In this way, the complex centralized controller is substituted by a group of simple controllers which are much easier to be installed and maintained.

The research in this thesis includes the design and analysis of different distributed and networked MPC algorithms which are based on different types of model-

s (discrete-time/continuous-time, linear/nonlinear) and different interpretations of uncertainties (stochastic/deterministic). Several numerical examples are also given to illustrate the effectiveness of the proposed algorithms.

1.2 Main Contributions

The main contributions of the thesis are listed as follows:

1. We design a distributed stochastic MPC algorithm for HVAC systems in buildings. The system model is simplified to a linear discrete-time one and the uncertainties are considered as random variables. A scenario-based approach is used to approximate the stochastic optimization problem. Sub-gradient method is implemented to solve the Lagrange dual problem in a distributed way and a vanishing step size is chosen to guarantee convergence.
2. We design a distributed robust MPC algorithm based on reduced order models. We consider a linear discrete-time model and the uncertainties are modelled as bounded disturbances. A high order model is used for off-line design to capture the system behaviour while a reduced order model is used for on-line optimization to reduce the computational burden. A neighbour-to-neighbour communication protocol is used to exchange the predicted trajectory between a local system and its neighbours. The computation of the decentralized auxiliary controllers and the invariant sets are discussed.
3. We design a distributed MPC algorithm for nonlinear continuous-time systems. We assume that each subsystem has contraction property. Then we make use of this property to estimate the discrepancy between the predicted trajectory and the disturbed one. A tightened constraint is designed to ensure that the real trajectory evolves around the predicted one. Then we show that with properly chosen parameters, the tightened constraint can guarantee feasibility and stability.

4. We apply the contraction based algorithm to HVAC systems in buildings. We design a hierarchical structure to control the HVAC system in different levels. Upper layer controller collects global information and solves an optimal control problem to generate reference signals with a long prediction horizon and updates with a lower frequency. Lower layer controllers use local information to track reference signals from the upper layer and optimize local performance indexes. Sufficient conditions on feasibility of the lower layer controllers are given under the assumption that the upper layer problem is always feasible.
5. We consider a networked control problem under the MPC framework. It is assumed that sensors and actuators are connected with the controller through a digital communication channel. We design a self-triggered scheme in order to reduce communication and computational burden. The quantization effect caused by the limited bandwidth of the digital communication channel is studied. Some sufficient conditions on feasibility and stability are given in terms of the prediction horizon, triggering levels, bit numbers and other system parameters.
6. We apply the proposed algorithms to HVAC control problems. The proposed algorithms can be implemented through well-developed sensor and actuator networks. We assume that indoor temperature can be measured by temperature sensors and damper position can be adjusted by electrical motor. The proposed algorithms are executed by processors based on measured temperature and predictive information to calculate control signal for dampers. Numerical examples and simulations are given to validate our results.

1.3 Outline of the Thesis

In Chapter 2, a literature review on the developments and state-of-the-art results of MPC and its applications on HVAC systems in buildings is given.

In Chapter 3, we propose a scenario-based distributed stochastic MPC algorithm. First of all, we derive a control oriented linear discrete-time model from a simplified nonlinear continuous-time one. By considering the weather conditions, occupancies and other disturbances as random variables, a stochastic optimization problem is formulated. Based on the scenario approach proposed in [17], [18], [19], the probabilistic constraints are substituted by randomized realizations and the stochastic optimization problem is approximated by a deterministic one. A sub-gradient based algorithm is used to solve the Lagrange dual problem in a distributed where a vanishing step size is chosen to guarantee convergence. Finally, a numerical example is given to illustrate the proposed algorithm.

In Chapter 4, we propose a robust distributed MPC based on reduced order models. We use a high order model in the off-line design to capture the system property as much as possible while a simplified model is used for on-line optimization to reduce the computational burden. A neighbour-to-neighbour communication protocol is designed such that the dynamical coupling can be compensated. We also discuss the computation of the decentralized auxiliary controllers and the invariant sets. Finally, we apply the proposed algorithm to a four-room HVAC system.

In Chapter 5, we design a distributed MPC based on nonlinear continuous-time models. Contraction property of a local subsystem is used to estimate the discrepancy between the nominal trajectory and the disturbed one. Based on this estimation, a tightened constraint is designed to guarantee that the real trajectory evolves around the nominal one. Feasibility and stability conditions are proved under some proper assumptions.

In Chapter 6, we apply the distributed algorithm developed in Chapter 5 to a HVAC system in building. A hierarchical control scheme is proposed to handle the complex structure of the building. We design the upper layer controller to generate reference temperature trajectories based on global information for lower layer controllers. The lower layer controllers are designed to track the reference trajectories while optimizing their own performance index. We show that the optimization problems

associated with the lower layer controllers are always feasible provided a feasible upper layer problem. Then we use a numerical example to show the effectiveness of the proposed algorithm.

In Chapter 7, a nonlinear continuous-time MPC is designed for networked control systems. The actuator and the sensor are connected with the controller through a digital communication network and because of the limited bandwidth, only quantized signal can be transmitted through the network. To reduce the communication and computation burden, a self-triggered scheme is designed. Some sufficient conditions on feasibility and stability are provided. Finally, we apply the proposed algorithm to two numerical examples.

In Chapter 8, we conclude the thesis and give some future research directions.

Chapter 2

Literature Review

2.1 HVAC System Modelling

To design an effective control algorithm for HVAC systems and analyze its performance, the model of an HVAC system is necessary. As in other engineering areas, the model of HVAC system is a compromise between accuracy and simplicity. To directly build a model for an HVAC system, from the basic equation of heat conduction

$$\frac{\partial \tau}{\partial t} = \alpha \left(\frac{\partial^2 \tau}{\partial x^2} + \frac{\partial^2 \tau}{\partial y^2} + \frac{\partial^2 \tau}{\partial z^2} \right)$$

where τ is the temperature of point (x, y, z) at time t , it can be observed that the thermodynamic system should be considered as a distributed parameter system which is very difficult to analyze. Note that the above equation is already a simplified one since no radiation and convection is considered. Also in real application, the air flow is inevitable. Therefore, the above equation cannot be directly used to analyse and design HVAC control systems even though it captures detailed physical feature of the heat transfer process.

A commonly used simplified schematic diagram for HVAC systems is shown in Fig 2.1 and the schematic diagram of AHU is given in Fig. 2.2. Fresh air is introduced from outside to guarantee the ventilation level. After being mixed with

Figure 2.1: Schematic diagram of the HVAC system

Figure 2.2: Schematic diagram of AHU

some return air, it passes a cooling coil which is usually driven by cold water, and then the air temperature decreases if cooling is demanded. After that, by using a fan, the cooled air is distributed to VAV boxes and then to each zone. The dampers in AHU can be used to adjust the portion of the return air and the fresh air while the zone dampers are used to adjust the air flow rate into each zone. Obviously, the open ratio of these dampers will affect the energy cost of the HVAC system. For example, if the portion of the fresh air is large, more energy will be consumed in the cooling coil or the heating coils to make the temperature of the air suitable.

Based on Fig 2.1, a lot of simplified models have been proposed. Based on different degree of simplification, these models can be divided into three classes: theoretical model, empirical model and hybrid model. Theoretical model is built from basic heat and mass balance equations. Though it is simplified from partial differential equations to ordinary differential equations, it still contains a large number of parameters to be identified and requires detailed information of duct structure, physical constants of fluid and so on. And it provides satisfactory performance for only a narrow range of geometric parameters [20].

On the other hand, empirical model requires less knowledge of the geometric specification of devices. In [21], optimal operation of a large cooling system is studied based empirical model of condensers, pumps and chillers. Several polynomials are used to build empirical models and parameters are identified by using operating data.

Hybrid model, which balances the complexity and accuracy, is more attractive than the previous two model for control design. Its model structures are derived from basic thermodynamic principles while their parameters are determined by real data. In [22], the author proposes a linear dynamic model for a duct. The duct is divided into a number of sections. For each section, a first order dynamic system is used to describe the temperature of the air in the duct. In [23], a simplified model for the cooling coil is proposed. Based on the energy balance and heat transfer equation, two-parameter and three-parameter dynamic models are proposed. Then

linear and nonlinear parameter identification methods can be used to identify the parameters. Another model for the cooling coil is given in [24]. The model is validated under Proportional, PI and PID control algorithms. A nonlinear model for a hot-water-to-air heat exchanger loop is built in [25] and a nonlinear control law is proposed. This model could be used to predict the effect of inlet air temperature, air flow rate and inlet water temperature to the output air temperature by using water flow rate as a control input. In [26], a hybrid static model for LDDS is proposed. This model is based on energy and mass balance principles and simplified by lumping the geometric parameters and fluids thermodynamic coefficients as constants. The resulted two equations contain seven parameters which could be identified via Levenberg-Marquardt method. By modelling each zone as a box with 6 faces, dynamic models for zone temperature are proposed in [27]. The author uses an analogue electric circuit to represent the building elements. Resistor (R) is used to represent the heat resistance of walls and air while capacitor (C) is used to represent the heat capacity. By extending the R-C model to R-C networks, in [28], the authors propose a multi-zone temperature model. By minimizing the error between the predict temperature and the real measurement, the R and C values are estimated. Finally, an experiment shows that the accuracy of this model is better than a conduction-only model. Then the authors extend their results to nonlinear case in [29], where model reduction techniques are also used to simplify the nonlinear term. In [30], a series of HVAC system components including a zone, heating coil, cooling and dehumidifying coil, humidifier, ductwork, fan and mixing box are modelled as linear first order systems. By using Ziegler-Nichols rule, the parameter of PID controller is tuned and the simulation results show good performance.

Figure 2.3: Schematics of on-off control

2.2 Control Strategies for HVAC Systems

2.2.1 Conventional Control

On-off control is the simplest control strategy in HVAC systems. The controller is a relay feedback controller with hysteresis and the control scheme is shown in Fig: 2.3. The air-conditioning device switches between on and off modes and temperature varies in a given temperature band. The upper and lower bound of the temperature band can be designed. On one hand, the band width should not be too large because of thermal comfort. On the other hand, the band width should not be too small otherwise devices switch between on and off frequently and will be worn quickly. In [31], the authors propose an optimization-based approach to determine the upper and lower bound of the temperature band to balance thermal comfort and on-off cycling frequency. An MPC based approach is proposed in [2] where the duty circle of air-conditioning system is determined by solving an open-loop optimal control problem repeatedly. Compared with [31], controller in [2] takes action before temperature reaches the given boundary so temperature variation is significantly reduced.

As in other process engineering areas, PID is the most commonly used control strategy for HVAC systems because of its simple structure which can be easily understood and implemented. A PID controller has three tuning parameters: proportional gain, integral time and derivative time. Performance of the closed-loop system is mainly dependent on the choice of the three parameters but tuning procedure is known as a time-consuming and expensive task. Furthermore, due to nonlinearity of HVAC system and time-varying working load over the year, retuning is usually necessary to

ensure satisfactory performance. Therefore, researchers mainly focus on self-tuning PID instead of standard PID. Some early works include [32] where system parameters are estimated from oscillation excited by a relay feedback. Another self-tuning PID algorithm is proposed in [33], where the authors use second-order plus dead time model to design PID controller and experiments show better performance than conventional first order approaches. Another self-tuning PID controller, which is also based on second-order plus dead time model, is proposed in [34]. In [35], self-tuning PID controllers are designed for both single- and multi-variable model of the HVAC system. In [36], a first-order plus dead time model is used and to compensate time delay, a Smith-predictor is designed.

Due to the low cost of design and implementation, the aforementioned HVAC control techniques are still widely used. However, in the long run, these controllers are not economic because usually they cannot adjust the control input properly according to time-varying thermal load. The inherent nonlinearity of the HVAC system also leads to undesired performance. To improve the efficiency of the HVAC system, a lot of advanced control strategies are proposed in many literatures.

2.2.2 Adaptive Control

Adaptive control is a kind of control strategy which contains some parameter estimations scheme and adjusts itself by using these estimations during the operation of the system. It can be used to control systems with uncertain and time-varying parameters which are commonly found in the model of HVAC systems because the parameters of the HVAC systems are usually dependent on environment temperature and weather condition which are uncertain and time-varying.

In [37], two parameter-adaptive (self-tuning) deadbeat control algorithms are proposed to control the temperature and humidity by adjusting water flows in the heat exchanger and humidifier. Due to the nonlinearity, parameters of the linearized model of the HVAC system are dependent on the operating point which is time-varying. By introducing a parameter estimation scheme, the controllers could be self-tuning

when the operating point is changing. Another adaptive controller which is used to adjust the flow rate of the mixed air and the water vapor is proposed in [38]. This controller is determined by minimizing the control increments and the square of the error between output and the reference. In [39], a pattern recognition adaptive controller which could adjust its gain and integral time is designed for first-order plus dead-time systems. Near optimal performance is guaranteed for a range of system parameters and noises. In [40], a model for a fan-coil heating and two thermal zones is built as a bilinear system with fifth-order dynamic disturbance. Then a robust adaptive controller is designed where the control gain matrices are determined by an algebraic Riccati equation. In [41], the adaptive controller is combined with a predictive scheme to compensate the thermal lag of the room to achieve better performance compared with traditional PI and on-off controllers. Another adaptive predictive controller is proposed in [42]. By using a recursive least squares algorithm combined with a z -domain fitting method, the parameters are estimated on-line. Based on the estimated dead time, a Smith predictor is used to compensate the time delay in the HVAC system and a self-tuning PI controller is proposed based on the predicted error and estimated parameters. With the assumption of passivity of the HVAC system, an adaptive feedforward controller is proposed based on the estimation of the steady state solution in [43]. [44] further extends the results into time-varying ambient temperature case. Due to the passivity of the HVAC system, the controller provides robust stability to the actuator saturation and environment changes. Furthermore, a passivity based optimal controller is designed to balance the thermal comfort and energy consumption. In recent years, by combining with fuzzy logic, neural network and adaptive control strategy, a lot of intelligent controllers are proposed, see, e.g. [45], [46], [47] and [48]. These hybrid controllers, on the one hand, inherit the benefits from both adaptive and intelligent controllers such as the ability to deal with complex and highly nonlinear systems. On the other hand, they also inherit many drawbacks such as difficulty of designing an adaptive law and lack of analytic results for the performance.

2.2.3 Robust Control

Robust control is another kind of control strategy which can be used to control systems with uncertain and time-varying parameters. Unlike an adaptive controller, which changes itself to adapt the varying parameter, a robust controller is fixed and designed well to guarantee that if changes are in given bounds, it will work well [49, 50].

In [51], the HVAC system is modelled as a nonlinear system. A traditional PID controller is designed for the linearized model around the operating point. Due to the changing of operating point, the performance of the PID controller is not satisfactory. In [52], a robust controller is designed based on the standard Riccati approach proposed by [53] and simulation results show better performance compared with the optimized PID controller. However, in this work, even though the model is simplified as a first-order plus time-delay model, the order of the controller is five which makes it difficult to implement. The high frequency disturbance is also ignored in this work. A robust control strategy using 'freezing', gain scheduling, integral term reset and feedback transition control is developed in [54] and [55] to take care of the control instability during the transition processes between different control modes. By economizer control and demand controlled ventilation control, the indoor air quality could achieve a better level while the coil energy consumption is reduced. In [56], an experimental HVAC system which consists of five subsystems including external and return air dampers, a variable speed blower and a heating coil is studied. An MIMO model is built for this system. A robust controller which is used to adjust the discharge air temperature and air flow rate is designed by μ -synthesis method. The four key control variables in the discharge air system are the input air and the water temperatures and the corresponding flow rates.

2.2.4 Model Predictive Control

In MPC, an optimization problem will be formulated based on the system dynamic model and current measurement at each time step. The working principle of MPC

Figure 2.4: Principle of MPC

is shown in Fig: 2.4. By solving this open-loop optimal control problem, an optimal control sequence is obtained. Only the first control action will be implemented and the optimization problem will be reformulated and solved again when the next measurement comes [57–62]. It should be noted that though at each time step an open-loop optimal control problem is solved, the whole system is closed loop since the optimal solution is a function of current measurement. It is superior to traditional control strategies due to its optimal nature and the ability to handle input, state and output constraints which come from the limitation of actuator and safety requirement. For example, in chemical engineering the open ratio of a valve cannot exceed 100% and should be no less than 0 while pressure and temperature should be kept in safe ranges. Though traditional PID controllers can be modified with a saturation to ensure that the control input constraints are satisfied, MPC can foresee and compensate before reaching the limitations and the performance is improved. During the past decades, great achievement has been made on the theoretical development and real application of both centralized and distributed MPC.

2.2.4.1 Centralized Implementation

Like most other control strategies, MPC is proposed as a centralized scheme at the beginning. The idea of predicting and optimizing system behavior by using internal model is included in [63], [64] and [65], where no constraint is considered and in [66], where constraint on control input is considered. A lot of successful applications of MPC have been found before mature analysis technique is developed. For example, in [67], an MPC algorithm is tested for the temperature control of a heat exchanger and the level/temperature control of a stirred tank. It is shown that compared with traditional control technique, MPC has superior performance, is easier to tune and has robustness with respect to time delay and nonlinearities. In [68], the authors apply MPC algorithm to a pilot plant consisting of two coupled distillation columns and superior performance has been shown when compared with single-loop PID controllers combined with a dynamic decoupler. In [69], MPC algorithm is compared with single loop and cascade PID by applications to a heat exchanger and an industrial autoclave.

However, due to the finite horizon formation of MPC, it cannot guarantee closed-loop stability if the optimization problem is not designed carefully, especially when the system is nonlinear and subject to constraints. For continuous-time systems, one significant result is given in [70], where a control problem is formulated with an infinite prediction horizon and a finite control horizon. For an unstable linear plant, the stability is guaranteed by a terminal equality constraint, which requires that the state of the system can be driven to the origin in finite steps. The terminal equality constraint is also used in [71] where a nonlinear system is considered and the stability is established by proving the differentiability of the value function with the help of many strong assumptions. Then those assumptions are relaxed in [72] where only local Lipschitz continuity of the value function is required. However, though some suboptimal alternative method is proposed in [73], to satisfy an equality constraint usually needs infinite numbers of iteration for a numerical solver, which limits the application of the aforementioned results. Besides that, a termi-

nal equality constraint also restricts the attraction region of the nonlinear system. Considering these issues, the result in [71] is extended to [74], where a terminal inequality constraint is used to only require that the state of the system enters a neighbourhood of the origin in finite time. In this paper, a dual-mode control scheme is also proposed, which means that when the state of the system is outside of the terminal set, a receding horizon scheme is applied and a local linear feedback controller is implemented once the state enters the terminal set. It is shown that if the state of the system is outside of the terminal set, then it will enter the set in finite time. After that, the local linear feedback controller can stabilize the system to the origin. In [75], another type of MPC is proposed where the prediction horizon is also a decision variable. The robust stability is guaranteed by a contraction constraint which requires the terminal state should shrink a little compared with the initial state. However, the recursive feasibility cannot be guaranteed and it is made as an assumption in this paper. In [76], a quasi-infinite horizon approach is proposed which inherits a lot of features from [74]. In this paper, the authors use a quadratic term to approximate the infinite cost of the system. A terminal constraint is designed such that inside the set the quadratic cost function is an upper bound of the infinite cost. Then by minimizing the quadratic term, the infinite cost can also be ‘minimized’ in some sense. The algorithms on the design of the terminal set, the quadratic cost function and the corresponding linear feedback controller are also discussed. Similar to [75], in [77], the authors also use a contraction constraint to ensure stability while feasibility is imposed as an assumption. A piece-wise constant MPC for nonlinear continuous-time system is proposed in [78] where the prediction model is continuous-time while the computed control signal is implemented in a sample-and-hold way. For discrete-time systems, in [79], an infinite horizon optimal control problem is approximated by using a receding horizon approach and the stability is proved based on a controllability assumption which is difficult to verify. In [80], a similar idea of [76] is adopted and the control horizon is chosen to be smaller than the prediction horizon to reduce the computational complexity.

It has been pointed out in [5] and [6] that the robustness of MPC is a nontrivial

topic when there are constraints on state and control input. By considering the uncertainties as bounded disturbances, one of the earlier approaches to guarantee robustness is min-max MPC [74], [81], [82], [83], [84] where the disturbances are also considered as decision variables and the control profile is chosen to minimize the maximal value for all possible realization of disturbances. In [85], state feedback controller is designed based on a min-max control formulation and robustness is achieved against additive bounded input disturbances and/or polyhedral parametric uncertainties in the state space matrices. However, due to the open-loop nature of this scheme, the result is usually too conservative and the computational complexity also limits its application. To overcome these issues, two different but related approaches are proposed in [86] and [87] for linear discrete-time systems. The control inputs in both literatures are a linear combination of a linear feedback term which is designed off-line and a disturbance term which is computed on-line by solving an optimization problem. The state and the control constraints are also properly tightened by using invariant sets. In [86], the feedback term is used to regulate the system and the additional term is for disturbance attenuation, while in [87], on the contrary, the additional term drives the nominal trajectory to the origin and the feedback term is used to correct the error caused by disturbances. However, these approaches are difficult to be generalized to nonlinear cases except a few works, e.g., [88], [89]. The other approach to deal with uncertainties is to treat them as random variable and this leads to stochastic MPC formulation where the constraints are formulated in a probabilistic sense. In [90], the authors formulate a stochastic optimal control problem and a nonlinear programming method is proposed to solve this problem. However, this paper only considers an open-loop optimization and feasibility and stability analysis are not given. In [91], recursive feasibility and stability are guaranteed with the help of a terminal equality constraint. Then the result is extended in [92] where the terminal equality constraint is relaxed to an inequality constraint. Except for additive uncertainties, multiplicative uncertainties are also studied in some literatures, e.g., [93], [94], [95] and [96] where the distributions of the uncertainties are used explicitly to tighten the constraints.

In recent years, inspired by randomized optimization technique, scenario-based approach [17–19] is used to solve stochastic MPC problem approximately without knowing the distribution of the uncertainties. For example, in [97, 98], the authors consider a linear discrete-time system where the uncertainties influence the system in a general way. No assumption on the distribution of the uncertainties is made and stability is proved in some probabilistic sense. However, feasibility is not considered in both papers. More recently, the scenario-based approach is extended to non-convex cases and some results on nonlinear stochastic MPC is obtained in [99].

2.2.4.2 Distributed Implementation

Though centralized MPC has been developed well in past years, for large scale systems such as traffic networks [100], power systems [101], biological systems [102] and building systems [103], a centralized implementation is often impractical. Firstly, it is usually not possible to collect full information of the whole system to realize a centralized controller. Secondly, as the dimension of the system grows, the corresponding optimal control problem becomes intractable and cannot be solved within a reasonable period of time. Distributed MPC has attracted more and more attention in recent years and different types of distributed algorithms have been proposed for different types of couplings and communication requirements.

Earlier works include [104] and [105] where decentralized computation schemes are proposed to solve a centralized MPC for water distribution systems. However, feasibility and stability issues are not studied. In [106], a distributed MPC is proposed for dynamically coupled linear discrete-time systems. For each subsystem, the prediction is based on local state and predicted trajectories of neighbours obtained at the previous time instant. Feasibility and stability are ensured by using a contraction constraint and controllability assumption but the state and input constraints are not included. In [107], a cooperative distributed MPC, which is based on iterative communications among subsystems between sampling time instants, is proposed for linear discrete-time systems and it is compared with centralized and decentralized approaches. However, the subsystems are only coupled through control input.

Then it is extended to output feedback cases [108] and applied to power systems [109] where dynamical couplings are also included but the analysis in [109] is mainly restricted to open-loop stable systems and for unstable systems, a centralized terminal controller is required which compromises the distributed nature of the proposed algorithm. A similar cooperative distributed MPC algorithm is proposed in [110] where a terminal equality constraint is used to stabilize the unstable modes of the subsystems. The result is extended to nonlinear systems in [111]. Another type of cooperative distributed MPC for nonlinear systems is proposed for a two-input system in [112] where a feasible stabilizing controller is assumed to be known before the design of MPC controller. A Lyapunov-based constraint is used to ensure the decreasing of the Lyapunov functional candidate and stability. Multi-input case is dealt with in [113]. Sequential and iterative algorithms are also designed for different communication requirements. Furthermore, asynchronous and delayed measurements are handled in [114] and [115]. Though the performance of these cooperative distributed algorithms can reach global optimum when the number of iterations is sufficiently large [109, 110] or can be better than a given feedback controller [113], each controller requires the model of the whole plant. This may be a drawback which will limit their applications to large-scale systems. To overcome this issue, several dual-decomposition based algorithms are proposed for dynamically coupled linear discrete-time systems. In [116] a dual-decomposition algorithm is proposed with sub-gradient updates of the Lagrange multipliers. A stopping criterion based on relaxed dynamic programming [117] is used to determine the number of iterations in the dual-decomposition algorithm. However, though state constraints are imposed in the primary problem, they may not be satisfied by using the control input obtained from dual problem. This issue is resolved in [118] where an adaptive constraint tightening technique is introduced to ensure that a feasible solution for the tightened dual problem is also feasible for the original primary problem. In [119] and [120], an accelerated sub-gradient algorithm is proposed to increase the convergence rate of the dual-decomposition algorithm from $\mathcal{O}(\frac{1}{k})$ to $\mathcal{O}(\frac{1}{k^2})$. However, these approaches are difficult to generalize to nonlinear systems since convexity is a key

assumption for dual-decomposition. Furthermore, robustness with respect to external disturbances is hard to ensure since the propagation of disturbances will destroy the sparsity of the system matrix. Following the idea of [106], some algorithms which only require partial information of the system while having some robustness to uncertainties are proposed. In [121], the author considers a group of interconnected nonlinear continuous-time systems. Each subsystem predicts its future behavior based on its neighbors' prediction obtained at the previous sampling time instant. A consistent constraint is imposed to ensure that the real trajectory is not far away from the predicted one. In this approach, stability does not depend on the type of the cost function therefore it can be chosen freely. In [122], a similar idea is used for interconnected linear discrete-time systems. However, different from [121], instead of using a consistent constraint, a tube-based approach [87] is used to keep the real trajectory evolving around the predicted one. Then this algorithm is extended to sample-data systems [123] and continuous-time systems [124]. A decentralized MPC algorithm is proposed for nonlinear discrete-time systems in [125] where the dynamical couplings are treated as external disturbances. More recently, a non-cooperative algorithm is proposed for weakly coupled nonlinear continuous-time systems in [126] where a robust constraint from [127] is imposed to each local optimization problem to bound the prediction trajectory. This is also a decentralized algorithm since no information is exchanged among agents.

For dynamically decoupled systems, the couplings can occur in the objective functions to achieve some degree of cooperation such as formation, tracking and so on. In [128], the authors design a distributed MPC algorithm for multi-vehicle formation problem. Vehicles are modeled by nonlinear continuous-time systems and the formation is enforced by using coupled objective function. Similar to [121], a compatibility constraint is imposed on each local optimization problem to enforce the consistency between what a vehicle plans to do and what its neighbors believe that it will plan to do. A similar problem is considered in [129] for a group of dynamically decoupled nonlinear discrete-time systems. Though no compatibility constraint is imposed in the optimization problem, the authors use an assumption to bound the

difference between prediction trajectories. In [130], a group of linear discrete-time systems is considered and a similar assumption to that in [129] is used to ensure that real trajectory is not far away from the assumed one. More recently, a robustness constraint, which explicitly forces the trajectory of each agent to converge to the origin, is used in [127]. The objective function is chosen as the Lyapunov functional candidate and the decreasing property is shown by treating the discrepancy between the real trajectory and the assumed one as disturbances. The extension of this work to time-delay cases can be found in [131] and [127]. Instead of using the objective function as a Lyapunov functional candidate, in [132] the authors use the approach in [86], where the control input is parametrized by a feedback term and a disturbed term, to design the local optimization problem. A contractive constraint is used to make the disturbed term vanish and the control input converges to a linear feedback law which is chosen to stabilize the system in a local region. In [133], the method in [134] is extended to a group of time-varying linear discrete-time systems whose objectives are tracking a reference signal cooperatively. The control input of each subsystem is parametrized as the output of a dynamical system. The parameter of the system can be designed off-line and the on-line decision variable only contains the initial value of this dynamical system and a distributed sub-gradient method is used to update the consensus signal. Vehicle platoon problem is addressed in [135] where a group of vehicles are modelled as dynamically decoupled nonlinear continuous-time systems and coupled objective functions are designed to make vehicles track a leader stably and cooperatively. Two different constraints are used to restrict the tracking error and two types of string stability are achieved. However, neither state nor input constraints are considered. This problem is also studied in [136] where a different constraint is proposed to restrict the tracking error and γ -gain stability is ensured and state and control constraints are satisfied.

Coupling can also appear in constraints, for example, driving a group of vehicles to a target set while maintaining relative formation and/or avoiding collisions. In [137], a distributed MPC algorithm with a sequential communication structure is proposed to solve such a problem. Each agent solves its local optimal control problem

based on its predecessor's solution such that the coupled constraints can be satisfied. In [138], a parallel structure is proposed for solving distributed MPC with coupled constraints. Different from [137], in this algorithm, each agent updates its control input in a parallel way but at each time instant only one agent solves the optimal control problem while other agents continue to use the feasible control inputs obtained at previous time instants. In [139], the coupled constraints are formulated in a probabilistic manner and a similar update strategy from [138] is used to implement the stochastic MPC algorithm in a distributed way.

2.2.4.3 Applications to HVAC systems

In [140], a constrained multivariable predictive control is developed in order to obtain a new DLC program. This control algorithm is used in the residential HVAC system to determine the forced duty-cycles for these devices in order to modify the global load curve according to a prefixed target load curve. The controller is implemented in a real HVAC system in the University of Cartagena and the experiment results show better performance in comparison with other methodologies. A novel model predictive controller is executed in [141] by using a three-step procedure including short-term weather prediction, optimization of control strategy by using building model and post-processing of the optimal strategy for the test facility. In [142], a GPC based PID controller is designed based on the identification model for a cooling coil unit in the HVAC systems. The GPC-PID controller with hierarchical structure is used to tune conventional PID controller parameters to deal with a wide range of operating conditions and uncertainties and better performance is achieved than conventional methods. In [143], a model predictive controller is used to control temperature and ventilation of multiple zones in a building. This controller predicts the temperature of the zones via using an MIMO linear model and solves a constrained convex quadratic optimization problem to obtain control actions to guarantee ventilation level and temperature comfort. A simple identification approach is proposed through closed loop step response data in [144]. Through this procedure, the MIMO

model of the HVAC system is decoupled into four independent single open loop processes. Based on this identification model, an MIMO GPC strategy is designed and simulation results show accuracy and robustness in the presence of noise. In [145], a decentralized MPC architecture for multiple evaporator vapor compression systems is proposed. The cooling capacity of each evaporator is controlled by an MIMO MPC controller and standard PI controllers are used to regulate system pressures by modulating compressor speed and discharge valve opening. This architecture is modified further in [146] by adding a global controller to generate the set points for the local MPC controllers. This controller balances the goals of cooling zone temperature tracking with optimal energy consumption. A building with a water tank used for actively storing cold water produced by a series of chillers is investigated in [3]. A dual-mode nonlinear model is built for the water tank because it might work on charging mode or discharging mode. Dual stage optimization and moving window blocking strategies are used to reduce the computational complexity. Algorithm in [147] is used to compute the control invariant set to guarantee persistent feasibility. Simulation results show that the daily electricity bill can be reduced by 24.5% compared to the current heuristic manual control strategy. Due to the inherent nonlinearity, the resulted optimization problem in MPC should be non-convex and local optima would occur by using nonlinear programming solvers based on sequential quadratic programming [148] or interior point methods [149]. In [150], the set of locally optimal solutions is thoroughly explored by repeatedly running a NLP solver using a large number of initial guesses which are distributed throughout the optimization space. Two different HVAC system configurations are studied and five locally optimal solutions are found for each configuration. Physical explanations are given for each local solution.

However, approaches in the references mentioned above are subjected to two problems. The first one is the assumption of bound for the disturbance. The other is that these controllers are designed to take care of the worst case. These problems lead conservativeness for the implementation of these robust MPC controllers. One possible way to reduce the conservativeness is to formulate SMPC instead of the

robust one. In [151], the SMPC problem is proposed to minimize the expected cost instead of the maximal possible cost in future N steps. Then the original problem is simplified by discretizing the continuous random variables into discrete ones. SMPC problems are formulated for the control of HVAC systems in [152–154]. In these articles, the weather and occupancy uncertainties are modelled by Gaussian random variables which are unbounded. Also some violation situations of constraints are allowed with small probability which leads to probabilistic constraints. By using methods in [155] [156] and [157], those chance constrained SMPC problems are reformulated as deterministic ones and are solved by a sequential quadratic programming method. Sparsity of the subproblem is used to derive fast iteration algorithms and disturbance feedback is introduced to improve performance.

In real applications, uncertainties caused by weather condition, human occupancy and so on are not necessarily of Gaussian distribution. However, stochastic optimization problem cannot be solved explicitly [17] without specific distribution assumption. On the other hand, approximate solution can be found by using randomized algorithm without specific distribution assumption. This approach is based on a scenario-based optimization technique [17–19]. In this approach, the involved random variables are substituted by samples of them so that the probabilistic constraints are converted into deterministic ones. The number of samples is determined by the dimension of the optimization problem, the confidence level of the desired solution and the violation probability of the constraints. Due to its ability to handle random variables with general distributions, RMPC has been used in building temperature and HVAC system control in recent years, see, e.g. [158], [159] and [160].

2.3 Conclusion

In this chapter, a comprehensive literature review has been given for the modelling and control of HVAC systems, especially for MPC and its application in the control of HVAC systems. Though a lot of detailed work has been done for the nominal MPC, the research on distributed MPC is still challenging and attractive. For example,

centralized stochastic MPC has been studied in existing literatures [153], [94], [95] and distributed stochastic MPC for dynamically decoupled systems are also studied in recent years [139]. However, for dynamically coupled systems, stochastic MPC is still an open problem. Another serious problem in distributed MPC for dynamically couple systems is conservativeness. For example, sufficient conditions given in [121] and [126] are even not satisfied by their examples. Such conservativeness limits the application of distributed MPC significantly.

Chapter 3

A Scenario-Based Distributed Stochastic MPC for Building Temperature Regulation

In this chapter, we consider the temperature regulation problem of buildings. By using the thermal dynamic model of buildings, occupancy schedule, and weather prediction and so on, a stochastic optimal control problem is formulated to keep the temperature of rooms within a comfortable range with a predefined probability while improving energy efficiency. Then by using several realizations of the uncertainties, which are called scenarios, we transform the corresponding stochastic optimization problem to a deterministic one. A sub-gradient-based dual decomposition method is used to solve the randomized optimization problem in a distributed manner such that the computational burden caused by the large number of rooms can be alleviated. A matrix norm based criterion is proposed to reduce the communication between controllers. Simulation results show the effectiveness of the proposed solution.

This chapter is organized as follows. Simplified control-oriented models for HVAC systems are introduced in Section 3.1. The centralized optimization problem is formulated in Section 3.2. A distributed algorithm is outlined in Section 3.3. Simulation results are given in Section 3.4 and conclusions are drawn in Section 3.5.

3.1 System Modeling

In this section we will introduce some notations from graph theory and derive simplified physical models for indoor temperature dynamics based on which control algorithms could be designed and analyzed in the following chapters.

A digraph is defined as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}_{\mathcal{G}}\}$, where $\mathcal{V} = \{1, \dots, N\}$ is the set of nodes, each of which represents a subsystem. $\mathcal{E}_{\mathcal{G}} \subseteq \mathcal{V} \times \mathcal{V}$ is the set of edges. If pair $(i, j) \in \mathcal{E}_{\mathcal{G}}$, the state of subsystem i is in the dynamics of subsystem j and subsystem i is referred to as an upstream neighbour of subsystem j while j is referred to as a downstream neighbour of i . The sets $\mathcal{N}_i^u = \{j \in \mathcal{V} | j \text{ is an upstream neighbour of } i\}$ and $\mathcal{N}_i^d = \{j \in \mathcal{V} | j \text{ is a downstream neighbour of } i\}$ collect all upstream neighbours and downstream neighbours of subsystem i , respectively. The neighbours of subsystem i is defined as $\mathcal{N}_i \triangleq \mathcal{N}_i^u \cup \mathcal{N}_i^d$.

Though a time-varying nonlinear partial differential equation can capture the detailed behavior of the indoor temperature dynamics, it is not suitable for control and optimization. Therefore, the following assumptions [161], [30], [4] are made to simplify the modeling.

- The temperature distributions of the air in each room and outdoor environment are uniform.
- The heat capacity of air is constant.

The indoor temperature dynamic model of each room is considered as an RC system [162] and that of the whole building is considered as a network of RC systems. In this network, each node represents a room and its state represents temperature of the room. Control input to each node is cooling/heating from HVAC system while weather condition and human activity are modeled as a random disturbance to each node.

Under the above assumptions, the following equation can be written as the dynamic model of indoor temperature:

$$C_i \dot{T}_i = Q_{out} + \sum_{j \in \mathcal{N}_i} Q_{ji} + Q_{cooling,i} + P_d^i, \quad i = 1, \dots, N, \quad (3.1)$$

where T_i is the temperature of room i , C_i is the lumped mass of room i , Q_{out} is the heat coming from outdoor environment which is random, Q_{ji} is the heat flow from room j to room i , $Q_{cooling,i}$ is the heat flow caused by the HVAC system to keep the indoor temperature of the i th room in a comfortable region, N is the number of all rooms and unmeasured disturbances in room i such as occupancy, usage of electronic devices and so on are collected in P_d^i . By using $R_{ij} = R_{ji}$ to model the heat resistance between room i and j , R_{oa}^i to model the heat resistance between room i and outdoor environment, the equation (3.1) can be written as follows:

$$C_i \dot{T}_i = \dot{m}_{vent,i} c_p (T_s - T_i) + \sum_{j \in \mathcal{N}_i} \frac{T_j - T_i}{R_{ij}} + \frac{T_{oa} - T_i}{R_{oa}^i} + P_d^i, \quad (3.2)$$

where $\dot{m}_{vent,i}$ is the ventilation mass flow of the i -th room, c_p is the heat capacity of air, T_s is the temperature of supply air and T_{oa} is environment temperature.

It can be observed that system (3.2) is bilinear in the state T_i and control input m_i . To facilitate controller design and performance analysis, linear discrete-time model may be used. To this end, we use u_i to substitute the nonlinear term $\dot{m}_{vent,i} c_p (T_s - T_i)$ and discretize the system by trapezoidal rule. Then we get a linear discrete-time system model as follows:

$$T_i(k+1) = a_{ii} T_i(k) + b_i u_i(k) + \sum_{j \in \mathcal{N}_i} a_{ji} T_j(k) + c_i w_i(k) \quad (3.3)$$

where a_{ii} , b_i , a_{ji} and c_i are constants related to C_i , R_{oa}^i , R_{ji} ; $w_i(k)$ models disturbance to room i from weather condition and human activity and The control variable $u_i(k)$ is duty circle and the PWM technique is used to convert the control $u_i(k)$ into a continuous signal that turns the zone dampers on or off ([2], [163]).

3.2 Randomized MPC formulation

In this section, the SMPC problem will be formulated based on LTI model (3.3). The disturbance $w_i(k)$, which collects weather condition and occupancy, is interpreted as random variables. In case that the number of historical data is not enough, we first estimate the distribution of weather condition and occupancy which can be used to generate samples. Copula is one of the most popular tools which can be used to give such estimation. Therefore, first of all, a brief introduction to copula is given.

A copula $C(y_1, \dots, y_n)$ is a multivariate distribution whose marginal distributions of $y_1, \dots, y_n$ are all uniform. Since the marginal distributions are fixed in a copula, it is used to describe the multivariate dependence structure. Given a random variable X with cumulative distribution F_X which is continuous, the random variable under a probability integral transform defined as $Y = F_X(X)$ has a uniform distribution on $[0, 1]$. Combined this fact with the concept of copula, we can estimate the marginal distributions and dependence structure of a group of random variables separately. Suppose that we want to estimate the multivariate distribution $F(x_1, \dots, x_n)$. The basic procedure is that:

- Collect empirical sample data from experiment.
- Estimate the empirical marginal distributions of $x_1, \dots, x_n$ as $F_{x_1}, \dots, F_{x_n}$.
- Use probability integral transforms $y_i = F_{x_i}(x_i)$, $i = 1, \dots, n$ to transform the collected data into points in $[0, 1]^n$.
- Estimate the dependence structure $C(y_1, \dots, y_n)$ of y_i , $i = 1, \dots, n$ by using some well-established copula families whose parameters describe the dependence among those random variables.
- Generate samples $(y_{1,j}, \dots, y_{n,j})$, $j = 1, \dots, N_s$ from the joint distribution $C(y_1, \dots, y_n)$ where N_s is the number of scenarios.
- Use inverse probability integral transforms $x_i = F^{-1}(y_i)$ to get scenarios $(x_{1,j}, \dots, x_{n,j})$, $j = 1, \dots, N_s$ from generated $(y_{1,j}, \dots, y_{n,j})$.

For more details of copula, please refer to [164], [165], [166] and references therein.

Let N_c be the prediction horizon. $T_i(k|t)$, $u_i(k|t)$ and $w_i(k|t)$ are used to denote the predicted temperature of room i at time t , the optimal control action to room i and the predicted disturbance to room i at the k -th time instant from t . Since the system is time invariant, in the following we assume the current time $t = 0$ and with some abuse of notations we use $T_i(k)$, $u_i(k)$ and $w_i(k)$ instead of $T_i(k|0)$, $u_i(k|0)$ and $w_i(k|0)$.

Let $T_i = [T_i(1), \dots, T_i(N_c)]^T$, $U_i = [u_i(0), \dots, u_i(N_c - 1)]^T$, $W_i = [w_i(0), \dots, w_i(N_c - 1)]^T$ and $W = [W_1^T, \dots, W_N^T]^T$. Then the future trajectory of the temperature of room i can be written as:

$$T_i = A_i T_i(0) + B_i U_i + C_i W_i + \sum_{j \in \mathcal{N}_i} A_{ji} T_j(0) + \sum_{j \in \mathcal{N}_i} D_{ji} T_j, \quad i = 1, \dots, N, \quad (3.4)$$

where matrices A_i , B_i , A_{ji} and D_{ji} can be easily constructed by using (3.3) recursively.

The above equations can be reorganized as:

$$\underbrace{\begin{bmatrix} I & -D_{21} & \dots & -D_{N1} \\ -D_{12} & I & \dots & -D_{N2} \\ \vdots & \vdots & \ddots & \vdots \\ -D_{1N} - D_{2N} & \dots & \dots & I \end{bmatrix}}_D \begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} = \begin{bmatrix} A_1 T_1(0) + B_1 U_1 + C_1 W_1 + \sum A_{j1} T_j(0) \\ A_2 T_2(0) + B_2 U_2 + C_2 W_2 + \sum A_{j2} T_j(0) \\ \vdots \\ A_N T_N(0) + B_N U_N + C_N W_N + \sum A_{jN} T_j(0) \end{bmatrix}$$

By taking the inverse of matrix D , the future temperature trajectory can be solved in the following equation:

$$\begin{bmatrix} T_1 \\ T_2 \\ \vdots \\ T_N \end{bmatrix} = \underbrace{\begin{bmatrix} K_{11} & K_{21} & \dots & K_{N1} \\ K_{12} & K_{22} & & \vdots \\ \vdots & & \ddots & \vdots \\ K_{1N} & \dots & \dots & K_{NN} \end{bmatrix}}_{K=D^{-1}} \times \begin{bmatrix} A_1 T_1(0) + B_1 U_1 + C_1 W_1 + \sum A_{j1} T_j(0) \\ A_2 T_2(0) + B_2 U_2 + C_2 W_2 + \sum A_{j2} T_j(0) \\ \vdots \\ A_N T_N(0) + B_N U_N + C_N W_N + \sum A_{jN} T_j(0) \end{bmatrix} \quad (3.5)$$

Let $T_{\max}^i$ and $T_{\min}^i \in \mathbb{R}^{N_c}$ denote comfort region of temperature in the i th room which serve as constraints in optimization. Denote $F = [I; -I]$, $s = [-s_{11}^T, \dots, -s_{1N}^T, s_{21}^T, \dots, s_{2N}^T]^T$ where s_{1i} and $s_{2i} \in \mathbb{R}^{N_c}$ are used to relax the upper and lower bounds of temperature of room i respectively, $T_{con} = [(T_{\max}^1)^T, \dots, (T_{\max}^N)^T, - (T_{\min}^1)^T, \dots, - (T_{\min}^N)^T]^T$ and $T = [T_1^T, T_2^T, \dots, T_N^T]^T$. The elements of all s_{1i} and s_{2i} are nonnegative. Note that s is used to avoid infeasible optimization problem and they do not have explicit physical meaning. The temperature constraints can be expressed as

$$FT \leq T_{con}$$

By substituting (3.5) into the above equation, we have

$$FK \begin{bmatrix} B_1 U_1 \\ B_2 U_2 \\ \vdots \\ B_N U_N \end{bmatrix} \leq T_{con} - FKQ$$

where

$$Q = \begin{bmatrix} A_1 T_1(0) + C_1 W_1 + \sum A_{j1} T_j(0) \\ A_2 T_2(0) + C_2 W_2 + \sum A_{j2} T_j(0) \\ \vdots \\ A_N T_N(0) + C_N W_N + \sum A_{jN} T_j(0) \end{bmatrix}$$

In cases solutions are infeasible, slack variable s is introduced to relax the temperature constraints further as follows:

$$FK \begin{bmatrix} B_1 U_1 \\ B_2 U_2 \\ \vdots \\ B_N U_N \end{bmatrix} + s \leq T_{con} - FKQ \quad (3.6)$$

Considering that the energy cost for air-conditioning of a group of rooms is the sum

of the energy cost in each room, we define the cost function as:

$$G(U, s) = \sum_{i=1}^N [c^T U_i + p^T (s_{1i} + s_{2i})]$$

where c contains the power of AC and the electricity charge, p is used to penalize the slack variables and $U = [U_1^T, \dots, U_N^T]^T$.

By denoting (3.6) as $g(U, s) \leq f(Q)$ where Q contains all the disturbance W_i , the probabilistic constraints optimization problem is formulated as follows:

Problem 3.1.

$$\begin{aligned} \min_{U, s} \quad & G(U, s) = \sum_{j=1}^N [c^T U_j + p^T (s_{1j} + s_{2j})] \\ \text{s.t.} \quad & P(g(U, s) \leq f(Q)) \geq 1 - \alpha, \\ & FU_j \leq U_{con}, \quad j = 1, \dots, N, \\ & 0 \leq s_{1j} \leq s_{\max}, \quad j = 1, \dots, N, \\ & 0 \leq s_{2j} \leq s_{\max}, \quad j = 1, \dots, N, \end{aligned}$$

where $U_{con} = [\mathbf{1}^T, \mathbf{0}^T]^T$ is the constraint for the control inputs and $1 - \alpha$ is the desired probability level for constraint satisfaction.

Assume that $W^i = [W_1^{iT}, \dots, W_N^{iT}]^T$, $i = 1, \dots, N_s$ are N_s scenarios generated from the estimated distribution of W and N_s satisfies that $N_s \geq \frac{2}{\alpha} (\ln \frac{1}{\beta} + d)$ where $\beta \in (0, 1)$ is the confidence level of the solution and d is the number of decision variables [19], [18]. It follows from [17] that Problem 3.1 can be approximated by the following deterministic linear programming:

Problem 3.2.

$$\min_{U, s} \quad G(U, s) = \sum_{j=1}^N [c^T U_j + p^T (s_{1j} + s_{2j})]$$

$$\begin{aligned}
s.t. \quad & g(U, s) \leq f(Q_i), \quad i = 1, \dots, N_s, \\
& FU_j \leq U_{con}, \quad j = 1, \dots, N, \\
& 0 \leq s_{1j} \leq s_{\max}, \quad j = 1, \dots, N, \\
& 0 \leq s_{2j} \leq s_{\max}, \quad j = 1, \dots, N,
\end{aligned}$$

where Q_i is constructed by using W^i to substitute W in Q .

Denote $Y = \min_i (T_{con} - FKQ_i) = [(Y_u^1)^T, \dots, (Y_u^N)^T, (Y_l^1)^T, \dots, (Y_l^N)^T]^T$ where $\min$ applies element-wise to a vector. Note that even though the matrix D is sparse due to the structure of the Laplacian matrix, its inverse K does not necessarily have such structure. It can be observed that to solve the future temperature trajectory of any room, the information of the whole building including all control sequences, the initial temperatures of all rooms and the disturbances to all rooms are needed. Therefore, to compute Y_u^i and Y_l^i precisely, one needs to know the temperature and disturbance information of all rooms. However, if those K_{ij} with small norms are set to be zero, partial information is enough to approximate Y_u^i and Y_l^i . The linear programming 3.2 can be rewritten as:

Problem 3.3.

$$\begin{aligned}
\min_{U, s} \quad & G(U, s) = \sum_{j=1}^N [c^T U_j + p^T (s_{1j} + s_{2j})] \\
s.t. \quad & FK \begin{bmatrix} B_1 U_1 \\ B_2 U_2 \\ \vdots \\ B_N U_N \end{bmatrix} + s \leq Y, \\
& FU_j \leq U_{con}, \quad j = 1, \dots, N, \\
& 0 \leq s_{1j} \leq s_{\max}, \quad j = 1, \dots, N, \\
& 0 \leq s_{2j} \leq s_{\max}, \quad j = 1, \dots, N
\end{aligned}$$

Denote $\tilde{K}$ as a matrix constructed from K by setting blocks K_{ij} as zero when the

norms of K_{ij} are less than ϵ_1 , $\tilde{Y} = Y + [\mathbf{0}^T, L^T]^T$ and

$$L^T = [\mathbf{1}^T B_1^T, \dots, \mathbf{1}^T B_N^T]^T (K - \tilde{K})^T.$$

The communication cost among optimizers will be reduced if Problem 3.3 is simplified as follows and the reason will be explained in Remark 3.3.

Problem 3.4.

$$\begin{aligned} \min_{U, s} \quad & G(U, s) = \sum_{j=1}^N [c^T U_j + p^T (s_{1j} + s_{2j})] \\ \text{s.t.} \quad & F\tilde{K} \begin{bmatrix} B_1 U_1 \\ B_2 U_2 \\ \vdots \\ B_N U_N \end{bmatrix} + s \leq \tilde{Y}, \\ & F U_j \leq U_{con}, \quad j = 1, \dots, N, \\ & 0 \leq s_{1j} \leq s_{\max}, \quad j = 1, \dots, N, \\ & 0 \leq s_{2j} \leq s_{\max}, \quad j = 1, \dots, N, \end{aligned}$$

In the centralized optimization Problem 3.4, there are $3N * N_c$ decision variables. The complexity of this problem will grow rapidly when the number of rooms and prediction horizons are large. Thus it is necessary to design some distributed algorithm to reduce the computational burden.

3.3 Distributed optimization algorithm

In this section, a distributed algorithm based on Lagrangian relaxation [167] and dual decomposition is proposed. By Lagrangian relaxation method, Problem 3.4 can be rewritten as:

Problem 3.5.

$$\begin{aligned}
& \max_{\lambda} \min_{U, s} \sum_{j=1}^N [c^T U_j + p^T (s_{1j} + s_{2j})] + \sum_{j=1}^N \lambda_j^T (H_j [U^T, s_{1j}^T, s_{2j}^T]^T - [\tilde{Y}_u^{jT}, \tilde{Y}_l^{jT}]^T) \\
& \text{s.t.} \quad FU_j \leq U_{con}, \quad j = 1, \dots, N, \\
& \quad 0 \leq s_{1j} \leq s_{\max}, \quad j = 1, \dots, N, \\
& \quad 0 \leq s_{2j} \leq s_{\max}, \quad j = 1, \dots, N, \\
& \quad \lambda_j \geq 0, \quad j = 1, \dots, N,
\end{aligned}$$

where

$$H_j = \begin{bmatrix} \tilde{K}_{1j} B_1 & \dots & \tilde{K}_{Nj} B_N & -I & 0 \\ -\tilde{K}_{1j} B_1 & \dots & -\tilde{K}_{Nj} B_N & 0 & I \end{bmatrix}$$

and $\lambda = [\lambda_1^T, \dots, \lambda_N^T]^T$.

$H_j [U^T, s_{1j}^T, s_{2j}^T]^T - [\tilde{Y}_u^{jT}, \tilde{Y}_l^{jT}]^T$ is the subgradient of λ_j . Note that for fixed Lagrangian variable λ , Problem 3.5 is separable. By taking advantage of this special structure, Problem 3.5 is decomposed into N subproblems and the j th subproblem is:

Problem 3.6.

$$\begin{aligned}
& \min_{U_j, s_{1j}, s_{2j}} M_j [U_j^T, s_{1j}^T, s_{2j}^T]^T \\
& \text{s.t.} \quad FU_j \leq U_{con}, \\
& \quad 0 \leq s_{1j} \leq s_{\max}, \\
& \quad 0 \leq s_{2j} \leq s_{\max},
\end{aligned}$$

where

$$M_j = \left[c^T + \sum_{i=1}^N \lambda_i^T E_{ij} p^T + \lambda_j^T \begin{bmatrix} -I \\ 0 \end{bmatrix} p^T + \lambda_j^T \begin{bmatrix} 0 \\ I \end{bmatrix} \right]$$

$$\text{and } E_{ij} = \begin{bmatrix} \tilde{K}_{ji} B_j \\ -\tilde{K}_{ji} B_j \end{bmatrix}.$$

When the original problem has a nonlinear convex objective function and constraints, then it can be solved by subgradient decomposition method efficiently, e.g. [103]. However, in this linear programming case, even though the optimal dual variables are found, the optimal primal variable cannot be constructed directly due to the linearity of the objective function. The solutions of the subproblems 3.6 are just the extreme points of the bounded feasible sets. Obviously, in most situations, those extreme points are either infeasible or not optimal to the original problem. This lack of coordinability in linear programming has also been reported and discussed in [167, 168]. In [167], the author proposed an average strategy to construct the primal optimal solution motivated by the fact that the optimal solution to the original problem is a nontrivial convex combination of the extreme points of the subproblems.

By combining the algorithm proposed in Theorem 3 in [167] and dual decomposition, the Problem 3.5 is solved as follows:

At the beginning of this algorithm, λ^1 is initialized as $\mathbf{1}$ and X_j^0 are set to be $\mathbf{0}$, $j = 1, \dots, N$. Give constants $a, c > 0$ and $b \geq 0$ and threshold $\epsilon_2 > 0$. Let $p = 1$:

- Step 1: The value of dual variable λ is set as λ^p and each λ_j^p is broadcast in the communication network.
- Step 2: Based on given dual variable λ , subproblems 3.6 are constructed and solved by its corresponding optimizer. The primal variables are constructed as $[(U_j^p)^T, (s_{1j}^p)^T, (s_{2j}^p)^T]^T = \frac{\sum_{l=1}^p X_j^l}{p}$.
- Step 3: Break out and set $[(U_j)^T, (s_{1j})^T, (s_{2j})^T]^T = [(U_j^p)^T, (s_{1j}^p)^T, (s_{2j}^p)^T]^T$ when $\|(U_j^p)^T - (U_j^{p-1})^T\| < \epsilon_2$; else go to Step 4.
- Step 4: Each U_j^p is broadcast in the communication network.
- Step 5: Each λ_j^{p+1} is updated by

$$\lambda_j^{p+1} = \mathcal{P}[\lambda_j^p + \alpha_p (H_j[U^T, s_{1j}^T, s_{2j}^T]^T - [(\tilde{Y}_u^j)^T, (\tilde{Y}_l^j)^T]^T)]_+,$$

where $\alpha_p = \frac{a}{b+cp}$, a , b , c are positive constants and $\mathcal{P}[*]_+$ is the projection of $*$ to the positive orthant.

- Step 6: Set $p = p + 1$ and go to Step 1.

Remark 3.1. *The selection of the stepsize α_p is critical to guarantee the convergence of the algorithm. In [167], the authors prove that if the stepsize is chosen as in Step 5, then $\frac{\sum_{l=1}^p X_j^l}{p}$ will converge to the optimal solution of the original problem as $p \rightarrow \infty$.*

Remark 3.2. *It can be observed that the on-line computational complexity is mainly dependent on Problem 3.3, whose dimension is independent with the network size. Therefore, we can conclude that online computational complexity is constant and the proposed algorithm scales well.*

Remark 3.3. *From the formulation of Problems 3.5 and 3.6, one can observe that to update λ_j^p and to construct M_j , the j th optimizer needs to know U_i^p and λ_i^p , $i = 1, \dots, N$ if we directly decompose Problem 3.3. This fact implies for any optimizer, it has to know the value of primal and dual variables from all other optimizers. However, by decomposing Problem 3.4 and solving corresponding subproblems, the communication cost is reduced. The reason is that some $\tilde{K}_{ij}$ are zero which leads to that the corresponding E_{ij} and columns in H_i are zero. Therefore U_j is not necessary for optimizer i and λ_i is not necessary for optimizer j . Especially, when K_{ij} and K_{ji} are both set to be zero, no information exchange is needed between optimizers i and j .*

3.4 Simulation results

To estimate the distribution of prediction error, we use the data downloaded from hvac.ee.kth.se and the dependence is constructed by using Matlab[®] copula package.

In this case study we consider a network of 10 rooms. The connection topology of rooms is $\mathcal{G} = \{\{1, \dots, 10\}, \{(1, 2), (2, 3), \dots, (9, 10)\}\}$. The configuration is shown

in Fig. 3.1. The parameters of rooms are given by $a_{ii} = 0.64$, $b_i = -3$, $a_{ji} = 0.1$, $c_i = 0.26$ when $i = 1$ or 10 , $c_i = 0.16$ when $i = 2, \dots, 9$. The comfortable region of temperature is set to be $[22^\circ\text{C } 23^\circ\text{C}]$ during the working time while $[20^\circ\text{C } 25^\circ\text{C}]$ during the off time.

By solving the centralized MPC problem directly, the control sequences, which represent the duty circles of AC devices, and temperature evolution are shown in Fig. 3.2 and Fig. 3.3. The cost is 238.6. The average computational time is 39 seconds. Fig. 3.4 and Fig. 3.5 show the temperature evolution under the distributed algorithm with fully connected and simplified topologies respectively. In the simplified topology case, we set those K_{ij} whose norms are less than $\epsilon_1 = 0.35$ to 0. In the fully connected topology case, there are 45 edges among all optimizers while in the simplified topology, there are only 17 edges. The costs are 240.1 and 240.2 respectively. One can see that under closed-loop implementation, the performance loss is negligible. The average computational time for the distributed algorithm is 9 seconds. Fig. 3.6 and Fig. 3.7 show the temperature controlled by PID controllers and the corresponding control sequences. It is observed that the upper bound and lower bound are both violated. The cost is 263.6 which is larger than all of the MPC results above. It can be also observed that by using MPC controllers, the temperature variation is much smaller and the control action is smoother. These properties are good for indoor thermal comfort and prolonging the life of devices.

The energy saving of the proposed MPC algorithm is not surprising because from the simulation, we can see that under MPC, the temperature trajectory is much closer to the upper bound of the temperature constraint. Since we assume that the HVAC system is working on a cooling mode, being close to the upper bound of the temperature constraint means saving energy. However, it cannot track the upper bound of the temperature constraints perfectly because of the uncertain disturbance.

Remark 3.4. *This simulation result is based on the simplified linear discrete-time model. System behavior may be different if we apply the proposed algorithm to a more realistic model or a real plant but the energy saving and constraint satisfaction shall not be compromised too much.*

Figure 3.1: Building configuration

3.5 Conclusions

In this chapter, we have considered the temperature control problem for a group of rooms. By using RMPC method, random disturbances such as weather, human activity and probabilistic constraints have been handled. By using scenario-based approach, this stochastic optimization problem has been translated to a deterministic one. Then a Lagrangian relaxation based algorithm has been proposed to solve this optimization problem in a distributed manner. To reduce the communication, a matrix norm based approximate criterion has been used. Finally, we have compared the control performance of centralized MPC, distributed MPC and PID controller. It has been shown that distributed MPC saves 8% energy cost compared with PID controller while the computational time is less than 25% of that of the centralized MPC algorithm.

Figure 3.2: Control profile under centralized MPC

Figure 3.3: Indoor temperature controlled by centralized MPC

Figure 3.4: Indoor temperature controlled by distributed MPC with fully connected topology

Figure 3.5: Indoor temperature controlled by distributed MPC with simplified topology

Figure 3.6: Indoor temperature controlled by PID

Figure 3.7: Control profile generated by decentralized PID

Chapter 4

Distributed Non-Cooperative Robust MPC Based on Reduced-Order Models

In the previous chapter, we formulated the optimal temperature regulation as a stochastic optimization problem by treating the external disturbances as random variables. Though scenario-based approaches we used in the previous chapter greatly simplify the on-line computation, it is difficult to analyze the performance of the closed-loop system. In this chapter, a non-cooperative distributed MPC algorithm based on reduced order model is proposed to stabilize large-scale systems. The large-scale system consists of a group of interconnected subsystems. Each subsystem can be partitioned into two parts: measurable fast part, whose states can be directly measured by sensors, and unmeasurable slow part. In the on-line computation phase, only the measurable dynamics of the corresponding subsystem and neighbour-to-neighbour communication are necessary for the local controller design. Satisfaction of the state constraints and the practical stability are guaranteed while the complexity of the optimization problem is reduced.

This chapter is organized as follows. In Section 4.1, the problem is formulated. In Section 4.2 and 4.3, the design and implementation procedures of the MPC controller

are given. In Section 4.4, a numerical example is given to illustrate our algorithm. Finally, some conclusions are drawn in Section 4.5.

4.1 Problem Formulation

Consider a linear discrete-time system

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{D}\mathbf{w}(k), \quad (4.1)$$

where $\mathbf{x}(k) \in \mathbb{R}^n$ is the state of the system, $\mathbf{u}(k) \in \mathbb{R}^m$ is the control input and $\mathbf{w}(k) \in \mathbb{R}^p$ is the external disturbance.

The system can be partitioned into N linear, discrete-time, interconnected non-overlapping subsystems. For each subsystem $\mathcal{S}_i$, the system dynamics is given by

$$x_i(k+1) = A_{i,i}x_i(k) + B_i u_i(k) + \sum_{j \in \mathcal{N}_i^1} A_{i,j}x_j(k) + D_i w_i(k), \quad (4.2)$$

where $x_i(k) \in \mathbb{R}^{n_i}$, $u_i(k) \in \mathbb{R}^{m_i}$ and $w_i(k) \in \mathbb{R}^{w_i}$ are the state, the control input and the bounded external disturbance for subsystem i and $\mathcal{N}_i^1 = \{j | A_{i,j} \neq \mathbf{0}, j = 1, \dots, N, j \neq i\}$. According to this decomposition, we have $\mathbf{x}(k) = (x_1^T(k), \dots, x_N^T(k))^T$, $\mathbf{u}(k) = (u_1^T(k), \dots, u_N^T(k))^T$, $\mathbf{w}(k) = (w_1^T(k), \dots, w_N^T(k))^T$,

$$\mathbf{A} = \begin{pmatrix} A_{1,1} & \dots & A_{1,N} \\ \vdots & \ddots & \vdots \\ A_{N,1} & \dots & A_{N,N} \end{pmatrix}, \mathbf{B} = \text{diag}(B_1, \dots, B_N) \text{ and } \mathbf{D} = \text{diag}(D_1, \dots, D_N).$$

Each $x_i(k)$ can be further partitioned into two parts: fast dynamics $x_i^1(k) \in \mathbb{R}^{n_{1,i}}$ and slow dynamics $x_i^2(k) \in \mathbb{R}^{n_{2,i}}$, where $n_{1,i} \ll n_{2,i}$. The matrices $A_{i,j}$, B_i and D_i have the corresponding structures:

$$A_{i,j} = \begin{pmatrix} A_{i,j}^{11} & A_{i,j}^{12} \\ A_{i,j}^{21} & A_{i,j}^{22} \end{pmatrix}, B_i = \begin{pmatrix} B_i^1 \\ B_i^2 \end{pmatrix}, D_i = \begin{pmatrix} D_i^1 \\ D_i^2 \end{pmatrix}.$$

The initial value of $x_i^2(k)$ is bounded by a known polyhedron Δ_i . The external disturbance $w_i(k)$ is bounded by $\mathbb{W}_i$. The matrix $A_{i,i}^{22}$ is Schur. The objective is to design a distributed control law to stabilize the system while enforcing the local constraints $x_i^1(k) \in \mathbb{X}_i$, $u_i(k) \in \mathbb{U}_i$ and the joint constraints $(x_1^T(k), \dots, x_N^T(k), u_1^T(k), \dots, u_N^T(k))^T \in \mathbb{J}_l$, $l = 1, \dots, n_c$. The sets $\mathbb{W}_i$, $\mathbb{X}_i$, $\mathbb{U}_i$ and $\mathbb{J}_l$ are polyhedrons containing the origin in their interiors. If the projection of the polyhedron $\mathbb{J}_l$ on (x_i^1, u_i) is not the whole hyperplane, (x_i^1, u_i) is called an argument of $\mathbb{J}_l$. If (x_i^1, u_i) and (x_j^1, u_j) are arguments of $\mathbb{J}_l$, $\mathbb{J}_l$ is defined as a joint constraint on subsystem i and j . We call subsystem j a neighbor of subsystem i if $A_{i,j} \neq \mathbf{0}$ and/or they have at least one common joint constraint h_m . We do not consider constraint on $x_i^2(k)$.

4.2 Controller Design

For simplicity, in this section, we assume that D_i^2 , $A_{i,j}^{12}$ and $A_{i,j}^{22}$ are $\mathbf{0}$, $\forall i \neq j$. This assumption means that slow dynamics of subsystems is decoupled. However, the fast dynamics of subsystems is still coupled through $A_{i,j}^{11}$. This assumption comes from the observation that furniture, electronic devices and occupancy in each room/zone cannot affect temperature of neighbour room/zone directly.

4.2.1 Bounded Unmeasurable State

First we rewrite the model of $x_i^2(k)$ into the following form:

$$\begin{aligned} x_i^2(k+1) &= A_{i,i}^{22}x_i^2(k) + d_i^2(k), \\ d_i^2(k) &= A_{i,i}^{21}x_i^1(k) + B_1^2u_i(k) + \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{21}x_j^1(k). \end{aligned}$$

Thus we obtain

$$x_i^2(k) = (A_{i,i}^{22})^k x_i^2(0) + \sum_{l=0}^{k-1} (A_{i,i}^{22})^{k-1-l} d_i^2(l). \quad (4.3)$$

Considering the bounds on $x_i(k)$, $x_j(k)$ and $u_i(k)$, we know that $d_i^2(k) \in A_{i,i}^{21}\mathbb{X}_i \oplus B_1^2\mathbb{U}_i \oplus \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{21}\mathbb{X}_j \triangleq \mathbb{D}_i^2$. Therefore, by (4.3), we have $x_i^2(k) \in (A_{i,i}^{22})^k \Delta_i \oplus \sum_{l=0}^{k-1} (A_{i,i}^{22})^l \mathbb{D}_i^2 \triangleq \mathbb{S}_i^2(k)$. An over-approximation $\mathbb{S}_i^2(\infty)$ for all $\mathbb{S}_i^2(k)$, $k \in \mathbb{N}$ can be easily given by $\Delta_i \oplus (I - A_{i,i}^{22})^{-1} \mathbb{D}_i^2$.

Rewrite the model of $x_i^1(k)$ in the following form:

$$\begin{aligned} x_i^1(k+1) &= A_{i,i}^{11}x_i^1(k) + B_i^1u_i(k) + \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{11}x_j^1(k) + d_i^1(k), \\ d_i^1(k) &= A_{i,i}^{12}x_i^2(k) + D_i^1w_i(k). \end{aligned} \quad (4.4)$$

From the above discussion, we know that $d_i^1(k) \in \mathbb{S}_i^2(\infty) \oplus D_i^1\mathbb{W}_i \triangleq \mathbb{D}_i^1$. This fact implies that $d_i^1(k)$ can be treated as an additive bounded disturbance to $x_i^1(k)$. Therefore distributed robust MPC can be designed to stabilize this system.

4.2.2 Distributed Robust MPC

Consider the nominal subsystem model

$$\hat{x}_i(k+1) = A_{i,i}^{11}\hat{x}_i(k) + B_i^1\hat{u}_i(k) + \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{11}\tilde{x}_j(k), \quad (4.5)$$

where $\hat{x}_i(k)$ is the nominal state trajectory, $\hat{u}_i(k)$ is the nominal control input and $\tilde{x}_i(k)$ is the reference state trajectory which will all be specified later.

Before providing the distributed robust MPC algorithm, the following assumption on decentralized stabilizability is necessary.

Assumption 4.1. *There exist matrices K_i , $i = 1, \dots, N$ such that (i) $A_{K_i} = A_{i,i}^{11} + B_i^1K_i$ is Schur stable, (ii) $\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K}$ is Schur, where $\hat{\mathbf{A}} = \begin{pmatrix} A_{1,1}^{11} & \dots & A_{1,N}^{11} \\ \vdots & \ddots & \vdots \\ A_{N,1}^{11} & \dots & A_{N,N}^{11} \end{pmatrix}$, $\hat{\mathbf{B}} = \text{diag}(B_1^1, \dots, B_N^1)$ and $\mathbf{K} = \text{diag}(K_1, \dots, K_N)$.*

The control law for the i -th subsystem (4.2) is given by

$$u_i(k) = \hat{u}_i(k) + K_i(x_i^1(k) - \hat{x}_i(k)). \quad (4.6)$$

By defining $e_i(k) = x_i^1(k) - \hat{x}_i(k)$, we have

$$\begin{aligned} e_i(k+1) = & A_{K_i} e_i(k) + \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{11} (x_j^1(k) - \tilde{x}_j(k)) \\ & + d_i^1(k) \end{aligned} \quad (4.7)$$

Due to the stability of the matrix A_{K_i} and the boundedness of $d_i(k)$, if it can be further guaranteed that $x_j^1(k) - \tilde{x}_j(k)$ are bounded, $\forall j = 1, \dots, N$, there exists a robust positively invariant set $\mathbb{E}_i$ for (4.7), such that $\forall e_i(k) \in \mathbb{E}_i$, $e_i(k+1) \in \mathbb{E}_i$ which implies that if $\hat{x}_i(k) \rightarrow 0$ as $k \rightarrow \infty$, $x_i^1(k) \rightarrow \mathbb{E}_i$ as $k \rightarrow \infty$.

At each time instant t , each subsystem $\mathcal{S}_i$ transmits its future state and control reference trajectory over the prediction horizon N_c to its neighbors $\mathcal{N}_i^1$, and the MPC controller of each subsystem $\mathcal{S}_i$ solves the following i -th optimization problem:

$$\min_{\hat{x}_i(t), \hat{u}_i[t:t+N_c-1]} V_i^N$$

subject to (4.5),

$$x_i^1(t) - \hat{x}_i(t) \in \mathbb{E}_i, \quad (4.8)$$

and for $k = t, \dots, t + N_c - 1$ and $l = 1, \dots, n_c$,

$$\hat{x}_i(k) - \tilde{x}_i(k) \in \mathbb{Z}_i, \quad (4.9)$$

$$\hat{u}_i(k) - \tilde{u}_i(k) \in \mathbb{S}_i, \quad (4.10)$$

$$\hat{x}_i(k) \in \hat{\mathbb{X}}_i \subseteq \mathbb{X}_i \ominus \mathbb{E}_i,$$

$$\hat{u}_i(k) \in \hat{\mathbb{U}}_i \subseteq \mathbb{U}_i \ominus K_i \mathbb{E}_i,$$

$$(\hat{x}_i^1(k), \hat{u}_i(k), \tilde{x}_{-i}^1(k), \tilde{u}_{-i}(k)) \in \hat{\mathbb{J}}_l,$$

$$\hat{x}_i(t + N_c) \in \hat{\mathbb{X}}_i^F,$$

where $V_i^N = \sum_{k=t}^{t+N_c-1} l_i(\hat{x}_i(k), \hat{u}_i(k)) + V_i^F(\hat{x}_i(t+N_c))$, $\mathbb{Z}_i$ is a convex set which contains the origin as its interior point, $\hat{\mathbb{X}}_i^F$ is a nominal terminal set, $l_i(\hat{x}_i(k), \hat{u}_i(k)) = \hat{x}_i(k)^T P_i \hat{x}_i(k) + \hat{u}_i(k)^T Q_i \hat{u}_i(k)$ is the stage cost, P_i and Q_i are positive definite matrices, $V_i^F(\hat{x}_i(t+N_c)) = \hat{x}_i(t+N_c)^T R_i \hat{x}_i(t+N_c)$ is the terminal cost, $\hat{\mathbb{J}}_l$ is constructed such that if $(\hat{x}_i^1(k), \hat{u}_i(k), \tilde{x}_{-i}^1(k), \tilde{u}_{-i}(k)) \in \hat{\mathbb{J}}_l$, then $(x_1^1(k), \dots, x_N^1(k), u_1(k), \dots, u_N(k)) \in \mathbb{J}_l$ with

$$\begin{aligned}\tilde{x}_{-i}^1(k) &= (\tilde{x}_1^1(k), \dots, \tilde{x}_{i-1}^1(k), \tilde{x}_{i+1}^1(k), \dots, \tilde{x}_N^1(k)) \\ \tilde{u}_{-i}^1(k) &= (\tilde{u}_1^1(k), \dots, \tilde{u}_{i-1}^1(k), \tilde{u}_{i+1}^1(k), \dots, \tilde{u}_N^1(k))\end{aligned}$$

Remark 4.1. (4.8) implies that $e_i(t) \in \mathbb{E}_i$. Based on the invariance of $\mathbb{E}_i$ and by induction, it follows that $e_i(k) \in \mathbb{E}_i$, $k = t, \dots, t+N_c-1$. Combining this fact with constraint (4.9), it can be obtained that $x_i^1(k) - \tilde{x}_i(k) \in \mathbb{E}_i \oplus \mathbb{Z}_i$, $k = t, \dots, t+N_c-1$ which guarantees the boundedness of $\sum_{j \in \mathcal{N}_i^1} A_{i,j}^{11}(x_j^1(k) - \tilde{x}_j(k)) + d_i^1(k)$ in (4.7). Considering (4.7), a sufficient condition to ensure the invariance of $\mathbb{E}_i$, $i = 1, \dots, N$ is that

$$A_{K_i} \mathbb{E}_i \oplus \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{11}(\mathbb{E}_j \oplus \mathbb{Z}_j) \oplus \mathbb{D}_i^1 \subseteq \mathbb{E}_i. \quad (4.11)$$

Remark 4.2. In the i -th optimization problem, only local state and control input are considered as decision variables and neighbors' information are used to formulate constraints so we can see that online computational complexity will not grow even when network scale becomes larger.

Remark 4.3. For linear discrete-time system, there exists other non-optimization based methods to design control law subject to state and control constraints, for example, see [169]. Compared with those approaches, our proposed algorithm can have additional flexibility on transient performance optimization at the expense of higher on-line computational complexity.

The following assumptions for the terminal constraint $\hat{\mathbb{X}}_i^F$ and terminal cost function $V_i^F(\cdot)$ are necessary to guarantee the recursive feasibility of the optimization problem and stability of the system.

Assumption 4.2. Denote $\hat{\mathbb{X}} = \prod_{i=1}^N \hat{\mathbb{X}}_i$, $\hat{\mathbb{U}} = \prod_{i=1}^N \hat{\mathbb{U}}_i$, $\hat{\mathbb{X}}^F = \prod_{i=1}^N \hat{\mathbb{X}}_i^F$ and $\hat{\mathbf{x}} = (\hat{x}_1, \dots, \hat{x}_N)$.

i) $\hat{\mathbb{X}}^F \subseteq \hat{\mathbb{X}}$ is an invariant set for $\hat{\mathbf{x}}(k+1) = (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K})\hat{\mathbf{x}}(k)$;

ii) $\mathbf{K}\hat{\mathbf{x}}(k) \in \hat{\mathbb{U}}$ for all $\hat{\mathbf{x}}(k) \in \hat{\mathbb{X}}^F$;

iii) for all $\hat{\mathbf{x}}(k) \in \hat{\mathbb{X}}^F$ and for a given constant $\alpha > 0$,

$$\sum_{i=1}^N [V_i^F(\hat{x}_i(k+1)) - V_i^F(\hat{x}_i(k))] \leq -(1+\alpha) \sum_{i=1}^N l_i(\hat{x}_i(k), \hat{u}_i(k)).$$

At each time instant t , a pair $(\hat{x}_i(t), \hat{u}_i[t : t+N_c-1])$ can be obtained by $\mathcal{S}_i$ by solving the i -th problem. Based on (4.5) and (4.6), the future state trajectory $\hat{x}_i[t+1 : t+N_c]$ can also be computed. Then set $\tilde{x}_i(t+N_c) = \hat{x}_i(t+N_c)$ and $\tilde{u}_i(t+N_c) = \hat{u}_i(t+N_c)$ as the last elements of the reference state and control trajectory and transmit them to the neighbours of the i -th subsystem for updating and reformulating the local optimization problems at time instant $t+1$.

Based on the above setup, we have the following convergence result of the distributed MPC algorithm.

Theorem 4.1. If Assumption 4.1 and 4.2 hold, then $(x_1^{1T}(k), \dots, x_N^{1T}(k))^T$ exponentially converges to the invariant set of system $\mathbf{x}(k+1) = (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K})\mathbf{x}(k) + \mathbf{d}(k)$, where $\mathbf{d}(k) = (d_1^{1T}(k), \dots, d_N^{1T}(k))^T$.

The proof goes along the line of Theorem 1 in [122] so it is omitted here.

4.3 Implementation

In this section, some algorithms for the computation of the weighted matrices, the feedback gains, the invariant sets and the initial feasible reference trajectory will be briefly introduced [170].

4.3.1 Feedback Gains and Weighted Matrices

Denote $\mathbf{R} = \text{diag}(R_1, \dots, R_N)$. Define matrices $\mathbf{S} = \mathbf{R}^{-1}$ and $\mathbf{Y} = \mathbf{K}\mathbf{S}$. Considering $\mathbf{S}$ and $\mathbf{Y}$ as feasible solutions to the following linear matrix inequalities (LMI), the feedback gain K_i and terminal weighted matrices R_i which satisfy Assumption 4.1 with $\mathbf{R} - (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K})^T \mathbf{R} (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K}) \succ 0$ can be obtained by the definition of $\mathbf{S}$ and $\mathbf{Y}$.

$$\begin{pmatrix} \mathbf{S} & (\hat{\mathbf{A}}\mathbf{S} + \hat{\mathbf{B}}\mathbf{Y})^T \\ \hat{\mathbf{A}}\mathbf{S} + \hat{\mathbf{B}}\mathbf{Y} & \mathbf{S} \end{pmatrix} \succ 0, \quad (4.12)$$

$$\begin{pmatrix} \mathbf{S}_{i,i} & (\hat{\mathbf{A}}_{i,i}\mathbf{S}_{i,i} + \hat{\mathbf{B}}_{i,i}\mathbf{Y}_{i,i})^T \\ \hat{\mathbf{A}}_{i,i}\mathbf{S}_{i,i} + \hat{\mathbf{B}}_{i,i}\mathbf{Y}_{i,i} & \mathbf{S}_{i,i} \end{pmatrix} \succ 0, \quad (4.13)$$

$$\mathbf{S}_{i,j} = \mathbf{0}, \forall i, j = 1, \dots, N (i \neq j), \quad (4.14)$$

$$\mathbf{Y}_{i,j} = \mathbf{0}, \forall i, j = 1, \dots, N (i \neq j), \quad (4.15)$$

where $\mathbf{S}_{i,j}$ and $\mathbf{Y}_{i,j}$, $\forall i, j = 1, \dots, N$ are the (i, j) th block of $\mathbf{S}$ and $\mathbf{Y}$.

Once $\mathbf{K}$ and $\mathbf{R}$ are computed, $\mathbf{P} = \text{diag}(P_1, \dots, P_N)$ and $\mathbf{Q} = \text{diag}(Q_1, \dots, Q_N)$ can be constructed to satisfy Assumption 4.2 iii) by considering the following inequality:

$$\mathbf{R} - (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K})^T \mathbf{R} (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K}) - (\mathbf{Q} + \mathbf{K}^T \mathbf{P} \mathbf{K})(1 + \alpha) \succ 0$$

Due to the positiveness of $\mathbf{R} - (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K})^T \mathbf{R} (\hat{\mathbf{A}} + \hat{\mathbf{B}}\mathbf{K})$, the above inequality can be satisfied by choosing sufficiently small α , $\mathbf{P}$ and $\mathbf{Q}$.

4.3.2 Computation of Sets

Sets $\mathbb{D}_i^1$, $i = 1, \dots, N$ can be easily computed following the discussion in Section 4.2. Then it needs to find $\mathbb{Z}_i \neq \emptyset$, $\mathbb{E}_i \subset \mathbb{X}_i$ and $K_i \mathbb{E}_i \subset \mathbb{U}_i$ which satisfy (4.11) for all $i = 1, \dots, N$. By using the $\text{box}(\cdot)$ operator, sets $\mathbb{E}_i$ and $\mathbb{Z}_i$, $i = 1, \dots, N$ can be constructed by the following algorithm.

Algorithm 1

- Step 1: initialize $\mathbb{Z}_i, i = 1, \dots, N$ as arbitrary hyper-rectangles
- Step 2: initialize $\mathbb{E}_i = \sum_{j \in \mathcal{N}_i^1} \text{box}(A_{i,j}^{11} \mathbb{Z}_j) \oplus \text{box}(\mathbb{D}_i^1), i = 1, \dots, N.$
- Step 3: compute $\mathbb{E}_i^+ = \text{box}(A_{K_i} \mathbb{E}_i) \oplus \sum_{j \in \mathcal{N}_i^1} \text{box}(A_{i,j}^{11} (\mathbb{E}_j \oplus \mathbb{Z}_j)) \oplus \text{box}(\mathbb{D}_i^1), i = 1, \dots, N.$
- Step 4: for all $i = 1, \dots, N$, if $\mathbb{E}_i^+ \subset \mathbb{E}_i$, which means that $\mathbb{E}_i$ satisfies (4.11), then go to step 5. Otherwise set $\mathbb{E}_i = \mathbb{E}_i^+$ and repeat step 3.
- Step 5: if $\mathbb{E}_i \subset \mathbb{X}_i$ and $K_i \mathbb{E}_i \subset \mathbb{U}_i$ then stop. Otherwise set $\mathbb{Z}_i = \beta \mathbb{Z}_i$ with $\beta \in (0, 1)$ and go to step 2.

Finally, $\hat{\mathbb{X}}_i^F$ can be simply chosen as $\beta \mathbb{E}_i$ with $\beta \in (0, 1)$.

4.3.3 Initial Feasible Reference Trajectories

Denote the current measurement of system state as $\tilde{\mathbf{x}}(0) = (\tilde{x}_1(0), \dots, \tilde{x}_N(0))$. The algorithm to find the initial feasible reference trajectories of each subsystem $\mathcal{S}_i$ is based on the one-step prediction.

Algorithm 2

- Step 1: for all $i = 1, \dots, N$, initialize $\tilde{x}_i(0)$ and $k = 0$.
- Step 2: for all $i = 1, \dots, N$, each subsystem $\mathcal{S}_i$ transmits its state $\hat{x}_i(k)$ to its neighbours $\mathcal{S}_j, j \in \mathcal{N}_i^1$ and solves the following optimization problem.

$$\min_{\hat{u}_i(k)} \|\tilde{x}_i(k+1)\|^2$$

subject to

$$\tilde{x}_i(k+1) = A_{i,i}^{11} \tilde{x}_i(k) + B_i^1 \hat{u}_i(k) + \sum_{j \in \mathcal{N}_i^1} A_{i,j}^{11} \tilde{x}_j(k)$$

$$\tilde{x}_i(k+1) \in \hat{\mathbb{X}}_i$$

$$(\hat{x}_i^1(k), \hat{u}_i(k), \tilde{x}_{-i}^1(k), \tilde{u}_{-i}(k)) \in \hat{\mathbb{J}}_l$$

$$\hat{u}_i(k) \in \hat{\mathbb{U}}_i$$

- Step 3: for $i = 1, \dots, N$, if $\tilde{x}_i(k+1) \in \hat{\mathbb{X}}_i^F$, set prediction horizon as $k+1$ and stop. Otherwise, set $k := k+1$ and go to step 2.

4.4 Numerical Example

In this section, the proposed control algorithm is tested on a building temperature regulation problem proposed in [170] with slight modification.

Consider four connected rooms A , B , C and D as in Fig. 4.1. Room A and B are combined together as one apartment while room C and D as the other apartment. The air temperature T_A , T_B , T_C and T_D are considered as the measurable states and the control objective is to regulate them to the set point. Each room is equipped with an air-conditioner q_A , q_B , q_C and q_D . The heat transfer coefficient between rooms $A - C$ and $B - D$ is $k_1^t = 5.8\text{W/m}^2\text{K}$, the one between rooms $A - B$ and $C - D$ is $k_2^t = 5.8\text{W/m}^2\text{K}$, and the one between each room and the external environment is $k_e^t = 3\text{W/m}^2\text{K}$. The nominal external temperature $\bar{T}_E$ is 35°C and solar radiation is not considered for simplicity. The volume of each room is $V = 48\text{m}^3$, and the wall surfaces between the rooms are all equal to $s_r = 12\text{m}^2$, while those between the rooms and the environment are equal to $s_e = 24\text{m}^2$. Air density and heat capacity are $\rho = 1.225\text{kg/m}^3$ and $c = 1005\text{J/kgK}$, respectively. Letting $\phi = \rho cV$, the dynamic model is the following:

$$\begin{aligned} \phi \frac{dT_A}{dt} &= s_r k_2^t (T_B - T_A) + s_r k_1^t (T_C - T_A) + s_e k_e^t (T_E - T_A) + \sum_{i=1}^3 s_A^i k_A^i (T_A^i - T_A) - q_A \\ \phi \frac{dT_B}{dt} &= s_r k_2^t (T_A - T_B) + s_r k_1^t (T_D - T_B) + s_e k_e^t (T_E - T_B) + \sum_{i=1}^3 s_B^i k_B^i (T_B^i - T_B) - q_B \\ \phi \frac{dT_C}{dt} &= s_r k_2^t (T_A - T_C) + s_r k_1^t (T_D - T_C) + s_e k_e^t (T_E - T_C) + \sum_{i=1}^3 s_C^i k_C^i (T_C^i - T_C) - q_C \end{aligned}$$

$$\begin{aligned}
\phi \frac{dT_D}{dt} &= s_r k_2^t (T_B - T_D) + s_r k_1^t (T_C - T_D) + s_e k_e^t (T_E - T_D) + \sum_{i=1}^3 s_D^i k_D^i (T_D^i - T_D) - q_D \\
\phi \frac{dT_A^i}{dt} &= 60 s_A^i k_A^i (T_A - T_A^i), \quad i = 1, 2, 3 \\
\phi \frac{dT_B^i}{dt} &= 60 s_B^i k_B^i (T_B - T_B^i), \quad i = 1, 2, 3 \\
\phi \frac{dT_C^i}{dt} &= 60 s_C^i k_C^i (T_C - T_C^i), \quad i = 1, 2, 3 \\
\phi \frac{dT_D^i}{dt} &= 60 s_D^i k_D^i (T_D - T_D^i), \quad i = 1, 2, 3
\end{aligned}$$

where T_A^i , T_B^i , T_C^i and T_D^i , $i = 1, 2, 3$ are used to represent the thermal dynamics of furniture and walls which are assumed unmeasurable and slow, s_A^i , s_B^i , s_C^i and s_D^i are the equivalent surfaces chosen randomly, and k_A^i , k_B^i , k_C^i and k_D^i are the equivalent heat transfer coefficients chosen randomly.

Remark 4.4. *In building HVAC systems, what can be directly adjusted are the opening ratios of dampers and the speed of supply fans, therefore after obtaining the power of cooling we need to transfer it to those controllable variables. If we mainly consider energy consumed by the cooling coil q_i . Then q_i is proportional to $\dot{m}_i(T_i - T_s)$ where $\dot{m}_i$ is air flow rate, T_i is indoor temperature while T_s is temperature of supply air. After obtaining q_i by solving MPC problem, we can calculate the corresponding air flow rate $\dot{m}_i$. Then the open ratio and speed of supply fan can be determined from characteristic curve.*

The considered equilibrium point is: $q_A = q_B = q_C = q_D = \bar{q} = s_e k_e^t (\bar{T} - \bar{T}_E) = 1081.4W$, with $T_A = T_B = T_C = T_D = \bar{T} = 20^\circ C$. Let $\delta T_A = T_A - \bar{T}$, $\delta T_B = T_B - \bar{T}$, $\delta T_C = T_C - \bar{T}$, $\delta T_D = T_D - \bar{T}$, $\delta q_A = (q_A - \bar{q})/c\rho V$, $\delta q_B = (q_B - \bar{q})/c\rho V$, $\delta q_C = (q_C - \bar{q})/c\rho V$ and $\delta q_D = (q_D - \bar{q})/c\rho V$.

The corresponding discrete-time model of the form (4.1) is obtained by mE-ZOH discretization [171] with sampling time $h = 10s$. The partition of inputs and measurable states is:

$$x^1 = [\delta T_A \quad \delta T_B]^T, \quad u^1 = [\delta q_A \quad \delta q_B]^T$$

$$x^2 = [\delta T_C \ \delta T_D]^T, \ u^2 = [\delta q_C \ \delta q_D]^T$$

The constraints on the inputs and the states of the discretized system are:

$$\begin{aligned} x_{\min}^1 &= [-10 \ -10]^T, \ x_{\max}^1 = [10 \ 10]^T \\ x_{\min}^2 &= [-10 \ -10]^T, \ x_{\max}^2 = [10 \ 10]^T \\ u_{\min}^1 &= [-2500 \ -2500], \ u_{\max}^1 = [2500 \ 2500] \\ u_{\min}^2 &= [-2500 \ -2500], \ u_{\max}^2 = [2500 \ 2500] \\ \|u^1\|_1 + \|u^2\|_1 &\leq 8000 \end{aligned}$$

Matrices $K_i = \begin{pmatrix} -0.0193 & -0.0002 \\ -0.0002 & -0.0193 \end{pmatrix}$ and $R_i = \begin{pmatrix} 2.9249 \cdot 10^7 & 118.2915 \\ 118.2915 & 2.9249 \cdot 10^7 \end{pmatrix}$ are constructed by finding feasible solutions to LMI (4.12)-(4.15). The selected weighting matrices are $P_i = I_2$ and $Q_i = 100I_2$. Algorithm 1 is used to find the invariant sets and the initial feasible reference trajectory is constructed by using Algorithm 2 with $N_c = 6$.

The initial condition of this numerical example is $\delta T_A = 3^\circ C$, $\delta T_B = 2^\circ C$, $\delta T_C = 4^\circ C$ and $\delta T_D = 5^\circ C$. The initial temperatures of the unmeasurable states are randomly chosen between $[22^\circ C, 25^\circ C]$. The real external temperature is assumed to randomly vary between $[25^\circ C, 45^\circ C]$. At time instant $t = 700s$, there is a sudden increase of temperature T_A and T_B representing the opening of doors and windows. Over time interval $[700s, 1400s]$ additional heat sources with $30^\circ C$ are added in both rooms. The control input is calculated by using the discretized model and applied to the continuous-time model. The simulation results are shown in Fig. 4.2. To compare the performance of the proposed distributed algorithm, a decentralized MPC and a centralized MPC with complete state information are also used to regulate the temperature. The simulation results are shown in Fig. 4.3-4.4. To show the difference among these algorithms more clearly, the temperature deviation between distributed algorithm and the decentralized algorithm are shown in Fig. 4.5-4.6. The temperature deviation between distributed algorithm and the centralized

algorithm are shown in Fig. 4.7-4.8.

In the Fig. 4.2-4.4, the black dashed lines represent the corresponding states and control inputs for fast dynamics of room A and C while the gray solid lines represent those of room C and D . In Fig. 4.5 the black dashed line represents $\delta T_{A,dis} - \delta T_{A,dec}$ and the gray solid line represents $\delta T_{B,dis} - \delta T_{B,dec}$, and in Fig. 4.6 they represent $\delta T_{C,dis} - \delta T_{C,dec}$ and $\delta T_{D,dis} - \delta T_{D,dec}$ respectively. In Fig. 4.7 the black dashed line represents $\delta T_{A,dis} - \delta T_{A,c}$ and the gray solid line represents $\delta T_{B,dis} - \delta T_{B,c}$, and in Fig. 4.8 they represent $\delta T_{C,dis} - \delta T_{C,c}$ and $\delta T_{D,dis} - \delta T_{D,c}$ respectively. The subscripts dis , dec , c represent distributed algorithm, decentralized algorithm and centralized algorithm respectively. For clarity reason, the slow dynamics are not shown in these figures.

In the decentralized algorithm, there is no information exchange between controllers. Therefore the joint constraint $\|u^1\|_1 + \|u^2\|_1 \leq 8000$ is tightened as $\|u^1\|_1 \leq 4000$ and $\|u^2\|_1 \leq 4000$. The dynamics coupling terms are treated as unknown bounded disturbance and the method in [172] are used to design local controller. For the centralized algorithm, we also follow the way in [172] to design the centralized controller.

From Figs. 4.2-4.4 it can be observed that with the additional heat source, all the three algorithms have some off-set. From Fig 4.5-4.6 one can see that distributed algorithm has faster response than the decentralized one. This is due to the fact that in the distributed algorithm, controllers can communicate with each other. Therefore, controllers for room A and B know how much power will be used by the other two controllers and the overall power, which is restricted to be less or equal to $8000W$, can be used more efficiently. This fact can also be observed in Fig. 4.2-4.3. In Fig. 4.2, controllers for room A and B attain the maximal power $2500W$ while in Fig. 4.3, their power has to satisfy the constraint $q_A + q_B \leq 4000W$. The off-set of both algorithms are similar. Fig 4.7-4.8 show that centralized algorithm has faster response than the distributed one and its off-set is also smaller than that of the distributed one. This is not surprising because the centralized algorithm utilizes the complete state information.

Figure 4.1: Schematic representation of a building with two apartments

Finally, we compare the energy, which is defined as $\int_0^{2110} \sum_{A,B,C,D} \|q_i\| dt$, and the average computational time consumed by the three algorithms in Table 4.1.

Table 4.1: Cost and average computational time of the three algorithms

	time(s)	cost(kJ)
Centralized (full model)	0.47	732
Distributed (reduced order)	0.23	758
Decentralized (reduced order)	0.35	893

4.5 Conclusion

In this chapter, we have proposed a distributed robust MPC algorithm. It has been assumed that each subsystem could be partitioned into two parts: main fast dynamics and slow dynamics. The future trajectory is predicted based on the main dynamics while the slow dynamics is not involved in the online computation, by which the complexity of the optimization problem has been significantly reduced. The effectiveness of our algorithm has been illustrated by numerical examples.

Figure 4.2: State and input trajectory of distributed algorithm. Black dashed lines represent room A and C while gray solid lines represent room B and D.

Figure 4.3: State and input trajectory of decentralized algorithm. Black dashed lines represent room A and C while gray solid lines represent room B and D.

Figure 4.4: State and input trajectory of centralized full order algorithm. Black dashed lines represent room A and C while gray solid lines represent room B and D.

Figure 4.5: Temperature deviation between distributed (black dashed line) and decentralized (gray solid line) algorithm (room A and B). In this example, the air-conditioning system is working on cooling mode. Therefore the distributed algorithm has faster response than the decentralized one since the former one has lower temperature.

Figure 4.6: Temperature deviation between distributed (black dashed line) and decentralized (gray solid line) algorithm (room C and D)

Figure 4.7: Temperature deviation between distributed (black dashed line) and centralized (gray solid line) algorithm (room A and B)

Figure 4.8: Temperature deviation between distributed (black dashed line) and centralized (gray solid line) algorithm (room C and D)

Chapter 5

Distributed Nonlinear Model Predictive Control Based on Contraction Theory

In the previous chapter, we have designed a robust distributed MPC algorithm for HVAC systems in buildings where a linear discrete-time model is used. However, nonlinearity of the HVAC systems cannot be captured and the system behavior between sampling time instants is not considered. In this chapter, a novel distributed MPC algorithm for continuous-time nonlinear systems is proposed. Contraction theory is used to estimate the prediction error in the algorithm, leading to new feasibility and stability conditions. Compared to existing analysis based on Lipschitz continuity, the proposed approach leads to a distributed MPC algorithm under less conservative conditions, allowing stronger couplings between subsystems and a larger sampling interval.

This chapter is organized as follows. In Section 5.1, the system model and control objective are stated together with some necessary assumptions and preliminaries. In Section 5.2, a distributed MPC algorithm is proposed. In Section 5.3, the recursive feasibility and stability of the proposed algorithm are analyzed. In Section 5.4, a numerical example is given to illustrate the effectiveness and advantages of the proposed algorithm. In Section 5.5, conclusions are drawn.

5.1 Problem Formulation

In this section, the system model will be given and the control problem under investigation will be formulated.

Without loss of generality, we assume that the states as well as control inputs of all the subsystems have the same dimension. Subsystem i has the following dynamics:

$$\dot{x}_i(t) = f_i(x_i(t), x_{-i}(t), u_i(t)), \quad t \geq t_0, \quad (5.1)$$

where $x_i \in \mathbb{R}^n$ is the state, $u_i \in \mathbb{U}_i \subseteq \mathbb{R}^m$ is the control input and $x_{-i} = \{x_j, j \in \mathcal{N}_i^u\}$ denotes the concatenated vector of the states of the upstream neighbours of subsystem i . Denote $x = (x_1^T, \dots, x_N^T)^T$, $u = (u_1^T, \dots, u_N^T)^T$ and the whole system can be written as

$$\dot{x}(t) = f(x(t), u(t)), \quad t \geq t_0, \quad (5.2)$$

where $f(x, u) = (f_1^T(x_1, x_{-1}, u_1), \dots, f_N^T(x_N, x_{-N}, u_N))^T$. Our objective is to design a distributed control algorithm to stabilize system (5.2) while enforcing the input constraints $u_i \in \mathbb{U}_i$.

To facilitate the controller design, we need the following assumptions [121].

Assumption 5.1. (a) The function $f : \mathbb{R}^{nN} \times \mathbb{R}^{mN} \rightarrow \mathbb{R}^{nN}$ is twice continuously differentiable and satisfies $f(\mathbf{0}, \mathbf{0}) = \mathbf{0}$; (b) System (5.2) has a unique, absolutely continuous solution for any initial condition $x(t_0)$ and any piecewise right-continuous control $u : [t_0, \infty) \rightarrow \mathbb{U} \triangleq \mathbb{U}_1 \times \dots \times \mathbb{U}_N$; (c) Sets $\mathbb{U}_i$ are compact subsets of $\mathbb{R}^m$ with the origin as their interior.

Consider the linearization of (5.1) around the origin as

$$\dot{x}_i(t) = A_{ii}x_i(t) + \sum_{j \in \mathcal{N}_i^u} A_{ij}x_j(t) + B_i u_i(t), \quad (5.3)$$

where $A_{ij} = \frac{\partial f_i}{\partial x_j}|_{(\mathbf{0}, \mathbf{0})}$ and $B_i = \frac{\partial f_i}{\partial u_i}|_{(\mathbf{0}, \mathbf{0})}$ and the decentralized model

$$\dot{x}_i^d(t) = f_i(x_i^d(t), \mathbf{0}, u_i(t)), \quad t \geq t_0.$$

Then we need the following decentralized stabilizability assumption:

Assumption 5.2. *There exists a decentralized linear feedback controller*

$$K = \text{diag}(K_1, \dots, K_N)$$

such that (5.3) is stabilized with $u_i(t) = K_i x_i(t)$ and $A_{ii} + B_i K_i$ is Hurwitz, $i = 1, \dots, N$.

Assumption 5.2 is used to construct a terminal controller which is critical for proving recursive feasibility and stability. It is standard and can be found in many existing works, see [121], [122], [124] and so on. Based on this assumption, we have the following lemma.

Lemma 5.1. *([76], [121]) Under Assumption 5.2, there exist positive constants ϵ , r and positive definite matrices $M_i \in \mathbb{R}^{n \times n}$, $i = 1, \dots, N$, such that*

$$\Omega_\epsilon \triangleq \{x \in \mathbb{R}^{nN} \mid \|x\|_M \leq \epsilon\}$$

is a positively invariant set of $\dot{x}(t) = f(x(t), Kx(t))$, where $M = \text{diag}(M_1, \dots, M_N)$, and $\|x(t)\|_M$ and $\|x_i^d(t)\|_{M_i}$ exponentially decay to 0 with decay rate not less than r and $Kx \in \mathbb{U}$ for all $x \in \Omega_\epsilon$.

The proof of the above lemma and the calculation of ϵ and r can be found in [121].

We also need the Lipschitz continuity of f_i :

Assumption 5.3. *f_i satisfies that*

$$\|f_i(x_i, x_{-i}, u_i) - f_i(x_i, x'_{-i}, u_i)\| \leq \sum_{j \in \mathcal{N}_i^u} L_{ij} \|x_j - x'_j\|,$$

where $x'_{-i} = \{x'_j, j \in \mathcal{N}_i^u\}$ and L_{ij} is a positive constant.

Remark 5.1. *In most of the existing nonlinear MPC formulations, the Lipschitz constant of f_i with respect to x_i is required which is the main source of conservativeness. In this work, however, we only use the Lipschitz constants of the couplings.*

5.2 Controller Design and Implementation

5.2.1 Offline design

In the off-line phase we assume that a feasible control input has been found at the beginning of the proposed algorithm. This control input should satisfy the conditions in Assumption 5.4 below such that the corresponding state trajectory has a contraction property:

Assumption 5.4. *A feasible open-loop control input $u^0(t) = (u_1^{0T}(t), \dots, u_N^{0T}(t))^T \in \mathbb{U}$, $t \in [t_0, t_0 + T]$ has been found, where $T > 0$ is the prediction horizon, such that the corresponding state trajectory of system (5.2) $x^0(t) = (x_1^{0T}(t), \dots, x_N^{0T}(t))^T$ satisfies*

$$\begin{aligned} x^0(t_0) &= x_0, \\ \frac{\partial f_i}{\partial x_i} M_i + M_i \frac{\partial f_i}{\partial x_i} &\leq -\beta_i M_i, \quad \forall x_i \in \Theta_i, \\ x_i^0(t_0 + T) &\in \Omega_i(\alpha\epsilon) \triangleq \{x_i \mid \|x_i\|_{M_i} \leq \frac{\alpha\epsilon}{\sqrt{N}}\}, \\ i &= 1, \dots, N, \end{aligned} \tag{5.4}$$

where

$$\Theta_i \triangleq \{x_i \mid \inf_{z_i \in \Gamma_i} \|x_i - z_i\|_{M_i} \leq \bar{l}_i\}, \tag{5.5}$$

$\beta_i > 0$, $\bar{l}_i > 0$, $0 < \alpha < 1$ and $\Gamma_i \triangleq \{x_i^0(t), t \in [t_0, t_0 + T]\} \cup \Omega_i(\alpha\epsilon)$.

Inequality (5.4) replaces the commonly used Lipschitz continuity condition. It guarantees that two trajectories starting from different initial conditions with the same control input converge to each other exponentially. This condition is also used in process control [10], networked system [173], Kalman filter [174] and so on.

Suppose that for each subsystem, the same prediction horizon T and sampling period δ are used. Denote $c_i \triangleq 2 \sum_{j \in \mathcal{N}_i^u} \frac{\lambda_{\max}(M_i^{\frac{1}{2}}) L_{ij} l_j}{\lambda_{\min}(M_j^{\frac{1}{2}})}$, $l_i < \bar{l}_i$, $d_i = 2 \sum_{j \in \mathcal{N}_i^u} \frac{L_{ij} \alpha \epsilon \lambda_{\max}(M_i^{\frac{1}{2}})}{\sqrt{N} \beta_i \lambda_{\min}(M_j^{\frac{1}{2}})}$. Then we also need to choose T , δ , l_i and γ_i such that the following inequalities hold:

$$\frac{c_i}{\beta_i} + d_i - l_i \leq 0, \tag{5.6}$$

$$\gamma \triangleq \sum_{i=1}^N \gamma_i < 1, \quad (5.7)$$

$$\xi_i T + \frac{2}{\beta_i} (l_i - d_i - \frac{c_i}{\beta_i}) (1 - e^{-\frac{1}{2}\beta_i \delta}) < (\gamma_i \epsilon - d_i - \frac{c_i}{\beta_i}) \delta. \quad (5.8)$$

In this off-line design phase, we need to find u^0 , x^0 , M_i , β_i , $\bar{l}_i$, T , δ , γ_i which satisfy Assumption 5.4, (5.6), (5.7) and (5.8). In particular, x^0 will serve as reference trajectory for on-line optimization, M_i and β_i are determined by the contraction property of the corresponding subsystem and $\bar{l}_i$ is a parameter which provides a margin of freedom. In general, u^0 can be found by solving a constrained problem with constraints in Assumption 5.4. However, due to the dynamical coupling and the nonlinearity, how to solve it in a distributed way is still under study. In this chapter we assume that these parameters are already known. Indeed, to have an initial feasible solution is necessary for most optimization problems and a lot of centralized implementations of MPC also assume that an initial feasible solution is available [58], [74]. This assumption can also be found in [138], [137] and [121] for distributed cases.

5.2.2 Initialization

Given initial state x_0 , at time t_0 , and initial feasible control input $u^0(t)$, $t \in [t_0, t_0 + T]$. For $t > t_0 + T$, $x_i^0(t)$ and $u_i^0(t)$, $i = 1, \dots, N$ are defined by the following decentralized subsystem:

$$\begin{aligned} \dot{x}_i^0(t) &= f_i(x_i^0(t), \mathbf{0}, K_i x_i^0(t)), \\ u_i^0(t) &= K_i x_i^0(t), \quad t \in (t_0 + T, \infty), \end{aligned} \quad (5.9)$$

where K_i satisfies Assumption 5.2.

Denote $t_k = t_0 + k\delta$, $k = 1, 2, \dots$. Then the control algorithm is stated as follows:

At initial time t_0 , given initial state x_0 , if $x_0 \in \Omega_\epsilon$, apply $u = u^0 = Kx^0$ where $K = \text{diag}(K_1, \dots, K_N)$; else send $x_i^0(s)$, $s \in [t_0, t_0 + T]$ to subsystem i and its downstream neighbours.

For any subsystem i over time interval $[t_0, t_1]$,

- 1) apply $u_i^0(s)$, $s \in [t_0, t_1]$;
- 2) compute $x_i^0(s)$, $s \in [t_0 + T, t_1 + T]$ according to (5.9);
- 3) transmit $x_i^0(s)$, $s \in [t_0 + T, t_1 + T]$ to its downstream neighbours;
- 4) receive $x_j^0(s)$, $s \in [t_0 + T, t_1 + T]$ from its upstream neighbours.

5.2.3 Online implementation

For any subsystem i at time instant t_k , $k \geq 1$, if $x(t_k) \in \Omega_\epsilon$, apply $u_i = K_i x_i$; else solve the following problem.

Subproblem i

$$\min_{\hat{u}_i(\cdot; t_k) \in \mathbb{U}_i} J_i(x_i(t_k), \hat{u}_i(\cdot; t_k))$$

subject to

$$\begin{aligned} \hat{x}_i(t_k; t_k) &= x_i(t_k), \\ \dot{\hat{x}}_i(s; t_k) &= f_i(\hat{x}_i(s; t_k), x_{-i}^0(s), \hat{u}_i(s; t_k)), \end{aligned} \quad (5.10)$$

$$\|\hat{x}_i(s; t_k) - x_i^0(s)\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)}), \quad (5.11)$$

$$\|\hat{x}_i(s; t_k) - \tilde{x}_i(s; t_k)\|_{M_i} \leq \xi_i, \quad (5.12)$$

$$s \in [t_k, t_k + T],$$

where $\tilde{x}_i(s; t_k)$ is defined as $\tilde{x}_i(t_k; t_k) = x_i(t_k)$, $\dot{\tilde{x}}_i(s; t_k) = f_i(\tilde{x}_i(s; t_k), x_{-i}^0(s), \tilde{u}_i(s; t_k))$, $\tilde{u}_i(s; t_k) \triangleq u_i^*(s; t_{k-1})$ when $s \in [t_k, t_{k-1} + T]$, $\tilde{u}_i(s; t_k) \triangleq K_i x_i^0(s)$ when $s \in [t_{k-1} + T, t_k + T]$ and $u_i^*(s; t_{k-1})$ is the optimal solution of **Subproblem i** obtained at t_{k-1} and $u_i^*(s; t_0) = u_i^0(s)$, $\xi_i > 0$.

Denote $x_i^*(s; t_k)$, $s \in [t_k, t_k + T]$ as the state trajectory of (5.10) with initial condition $x_i^*(t_k; t_k) = x_i(t_k)$ and control input $u_i^*(s; t_k)$.

For any subsystem i over time interval $[t_k, t_{k+1}]$, $k \geq 1$,

- 1) apply $u_i^*(s; t_k)$, $s \in [t_k, t_{k+1}]$;
- 2) compute $x_i^0(s)$, $s \in [t_k + T, t_{k+1} + T]$ according to (5.9);
- 3) transmit $x_i^0(s)$, $s \in [t_k + T, t_{k+1} + T]$ to downstream neighbours;
- 4) receive $x_j^0(s)$, $s \in [t_k + T, t_{k+1} + T]$ from upstream neighbours.

In this Chapter, state constraints are not involved for conciseness. However, the proposed algorithm can be adapted to deal with state constraints by using constraint tighten technique as [175]. In **Subproblem i** , continuous-time system models, constraints and control inputs are considered. Therefore the resulted optimization problems are only conceptual since they are defined over continuous-time functional spaces. In the numerical example in Section 5.4, approximate discrete-time problems are solved in practice with integration time much smaller than δ . See [78] for detailed discussion.

In this proposed control algorithm, a dual mode strategy [74] is applied since if $x \in \Omega_\epsilon$, according to Lemma 5.1, the linear feedback controller $u = Kx$ can stabilize the system to the origin. However, this requires that every subsystem knows the full state information $x(t_k)$. In the simulation given in Section 5.4, the MPC algorithm is used even after $x(t)$ enters Ω_ϵ but the system is still stabilized to the origin.

In **Subproblem i** , two constraints (5.11) and (5.12) are used to guarantee recursive feasibility and stability. In particular, constraint (5.11) is used to make sure that the real trajectory is not far away from x^0 so that the nominal model (5.10) is a reasonable approximation. Constraint (5.12) is used to guarantee that the solution of the current time instant is not far away from that obtained at the previous time instant, which is critical for stability. Therefore the objective function J_i can be chosen as any performance index and the commonly used assumptions on the objective function in [58] are not required.

In [121], each subsystem i needs to transmit the whole predicted state trajectory $x_i^*(s; t_k)$, $s \in [t_k, t_k + T]$ to its downstream neighbours. While in our algorithm,

only the incremental part $x_i^0(t)$, $t \in [t_k + T, t_{k+1} + T]$ needs to be transmitted to the downstream neighbors. This results in that the amount of transmitted data is much smaller.

5.3 Analysis

In this section, recursive feasibility and stability of the system under the proposed control algorithm are established.

5.3.1 Feasibility

Since the receding horizon algorithm is optimization based, one may wonder if there exists some future time instant t_k , say, when **Subproblem** i becomes infeasible and then the controller of subsystem i has no control input. Therefore, recursive feasibility, which means that given an initial feasible solution at time t_0 , for all t_k , $k = 1, \dots$, every **Subproblem** i is feasible, is an important property to be established.

In **Subproblem** i , x_{-i}^0 is used to predict the future behaviour of subsystem i . There is a discrepancy between $x_i^*(s; t_k)$, $s \in [t_k, t_k + T]$ and the real trajectory $x_i(s)$, $s \in [t_k, t_k + T]$. We introduce the following lemmas to estimate the discrepancy. The first lemma is similar to Theorem 5 in [176].

Lemma 5.2. *Consider two systems $\dot{x}_0 = f(x_0, \mathbf{0}, t)$ and $\dot{x}_1 = f(x_1, w, t)$, where w is a bounded disturbance. Suppose that $\frac{\partial f^T}{\partial x} M + M \frac{\partial f}{\partial x} \leq -\beta M$ holds for some positive number β and positive definite matrix M in some region C where the two systems evolve. Then $V(t) \leq \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})} + (V(0) - \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})})e^{-\frac{1}{2}\beta t}$ where $V(t) = \|x_0(t) - x_1(t)\|_M$, L is a positive constant that satisfies $\|f(x, y, z) - f(x, y', z)\| \leq L\|y - y'\|$, $d = \sup_w \|w\|_M$.*

Proof. Similar to the proof of Theorem 5 in [176], pick $\gamma(r) = x_0(0) + r(x_1(0) - x_0(0))$. Consider an auxiliary trajectory $\dot{x}_r = f(x_r, rw, t)$ which starts from $x_r(0) = \gamma(r)$, $r \in [0, 1]$. Define $p(t, r) = \frac{\partial x_r}{\partial r}(t, r)$. Then it follows that

$$\frac{\partial p}{\partial t}(t, r) = \frac{\partial}{\partial t} \left(\frac{\partial x_r}{\partial r} \right) = \frac{\partial}{\partial r} \left(\frac{\partial x_r}{\partial t} \right) = \frac{\partial}{\partial r} f(x_r, rw, t) = \frac{\partial f}{\partial x_r} p + \frac{\partial f}{\partial rw} w.$$

Consider $\Phi(t, r) = p(t, r)^T M p(t, r)$. Then by the contraction property we have

$$\frac{\partial}{\partial t} \Phi(t, r) \leq -\beta \Phi(t, r) + 2\sqrt{\Phi(t, r)} L \frac{d\lambda_{\max}(M^{\frac{1}{2}})}{\lambda_{\min}(M^{\frac{1}{2}})}.$$

Note that $\sqrt{\Phi(0, r)} = V(0)$ and that it follows from comparison principle that

$$\sqrt{\Phi(t, r)} \leq \mu + (V(0) - \mu)e^{-\frac{1}{2}\beta t}, \quad (5.13)$$

where $\mu = \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})}$, $\forall x \in C$, $t \geq 0$, and $\forall r \in [0, 1]$.

According to the fundamental theorem of calculus one can write that $x_1(t) - x_0(t) = \int_0^1 p(t, r) dr$ and by inequality (5.13), we conclude that

$$V(t) \leq \int_0^1 \|p(t, r)\|_M dr = \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})} + (V(0) - \frac{2Ld\lambda_{\max}(M^{\frac{1}{2}})}{\beta\lambda_{\min}(M^{\frac{1}{2}})})e^{-\frac{1}{2}\beta t}.$$

■

The following lemma provides an estimate of the discrepancy.

Lemma 5.3. *If over time interval $[t_k, t_k + T]$, the real state trajectory $x_i(s)$ and the predicted state trajectory $x_i^*(s)$ are both in Θ_i defined in (5.5), $i = 1, \dots, N$, then the inequality $\|x_i(s) - x_i^*(s; t_k)\|_{M_i} \leq \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)})$, $s \in [t_k; t_k + T]$ holds for $i = 1, \dots, N$.*

Proof. Given the initial condition $x_i(t_k)$, the real state trajectory is determined by

$$\dot{x}_i(s) = f_i(x_i(s), x_{-i}(s), u_i^*(s; t_k)), \quad s \in [t_k; t_k + T].$$

The predicted state trajectory is given by

$$\dot{x}_i^*(s; t_k) = f_i(x_i^*(s; t_k), x_{-i}^0(s), u_i^*(s; t_k)), \quad s \in [t_k; t_k + T].$$

Since $x_i(s)$ is in Θ_i , $\|x_i(s) - x_i^0(s)\|_{M_i} \leq l_i$. Define $V_i(s) = \|x_i(s) - x_i^*(s; t_k)\|_{M_i}$. By considering $w = x_{-i}(s) - x_{-i}^0(s)$ and noting that $V_i(t_k) = 0$, the result directly follows from Lemma 5.2. $\blacksquare$

Based on the above lemmas, we now establish the recursive feasibility of the proposed control algorithm.

Theorem 5.1. *Suppose that Assumptions 5.1, 5.2, 5.3 and 5.4 hold, initial feasible control $u^0(t)$, $t \in [t_0, t_0 + T]$ is available at time instant t_0 and **Subproblem** i is feasible for $i = 1, \dots, N$ at time instant t_k . Then **Subproblem** i , $i = 1, \dots, N$, are feasible at time instant t_{k+1} with the feasible solution $\tilde{u}_i(s; t_{k+1})$ defined as*

$$\begin{cases} u_i^*(s; t_k), & s \in [t_{k+1}, t_k + T] \\ K_i x_i^0(s), & s \in [t_k + T, t_{k+1} + T] \end{cases}$$

provided that (5.6) holds.

Proof. Firstly we show that $\tilde{u}_i(s; t_{k+1}) \in \mathbb{U}_i$, $s \in [t_{k+1}, t_{k+1} + T]$. When $s \in [t_{k+1}, t_k + T]$, $\tilde{u}_i(s; t_{k+1}) = u_i^*(s; t_k) \in \mathbb{U}_i$ since $u_i^*(s; t_k)$ is the optimal solution of **Subproblem** i at time instant t_k . When $s \in (t_k + T, t_{k+1} + T]$, by Lemma 5.1 and Assumption 5.4 we know that $K_i x_i^0(s) \in \mathbb{U}_i$. Therefore $\tilde{u}_i(s; t_{k+1}) \in \mathbb{U}_i$, $s \in [t_{k+1}, t_{k+1} + T]$.

In the following of this proof it will be shown that the state trajectory given by

$$\dot{\tilde{x}}_i(s; t_{k+1}) = f_i(\tilde{x}_i(s; t_{k+1}), x_{-i}^0(s), \tilde{u}_i(s; t_{k+1}))$$

with initial condition $\tilde{x}_i(t_{k+1}, t_{k+1}) = x_i(t_{k+1})$ satisfies the constraints in **Subproblem** i . The optimal state trajectory computed at time instant t_k over time interval

$[t_{k+1}, t_k + T)$ is given by $\dot{x}_i^*(s; t_k) = f_i(x_i^*(s; t_k), x_{-i}^0(s), u_i^*(s; t_k))$ with initial condition $x_i^*(t_{k+1}; t_k)$.

Consider $V_i(s) = \|x_i^*(s; t_k) - \tilde{x}_i(s; t_{k+1})\|_{M_i}$. According to Lemma 5.3, we know that

$$V_i(t_{k+1}) = \|x_i(t_{k+1}) - x_i^*(t_{k+1}; t_k)\|_{M_i} \leq \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta}).$$

By constraint (5.11), it can be obtained that

$$\|x^*(t_{k+1}; t_k) - x_i^0(t_{k+1})\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta}).$$

Therefore we obtain that

$$\|\tilde{x}_i(t_{k+1}; t_{k+1}) - x_i^0(t_{k+1})\|_{M_i} \leq \|x_i(t_{k+1}) - x_i^*(t_{k+1}; t_k)\|_{M_i} + \|x^*(t_{k+1}; t_k) - x_i^0(t_{k+1})\|_{M_i} \leq l_i,$$

which implies that at time instant t_{k+1} the feasible state trajectory is in the contraction region Θ_i and the contraction property holds. Then by Lemma 5.2 one can obtain that

$$\|\tilde{x}_i(s; t_{k+1}) - x_i^*(s; t_k)\|_{M_i} \leq \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta})e^{-\frac{1}{2}\beta_i(s-t_{k+1})}.$$

According to constraint (5.11), we have

$$\|x_i^*(s; t_k) - x_i^0(s)\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)}),$$

which leads to

$$\|\tilde{x}_i(s; t_{k+1}) - x_i^0(s)\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_k)}) + \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i\delta})e^{-\frac{1}{2}\beta_i(s-t_{k+1})} \triangleq C_i.$$

By some simple calculation, we have

$$\begin{aligned} C_i &= l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})}) + \frac{c_i}{\beta_i}e^{-\frac{1}{2}\beta_i(s-t_k)} - \frac{c_i}{\beta_i}e^{-\frac{1}{2}\beta_i\delta - \frac{1}{2}\beta_i(s-t_{k+1})} \\ &= l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})}), \end{aligned}$$

which implies that

$$\|\tilde{x}_i(s; t_{k+1}) - x_i^0(s)\|_{M_i} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})})$$

when $s \in [t_{k+1}, t_k + T]$.

When $s \in [t_k + T, t_{k+1} + T]$, consider the state trajectory given by

$$\dot{\tilde{x}}_i(s; t_{k+1}) = f_i(\tilde{x}_i(s; t_{k+1}), x_{-i}^0(s), K_i x_i^0(s)),$$

where $x_i^0(s)$ is defined by (5.9).

Define $E_i(s) = \|\tilde{x}_i(s; t_{k+1}) - x_i^0(s)\|_{M_i}$ with initial condition $E_i(t_k + T) = l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(T-\delta)})$ and $s \in [t_k + T, t_{k+1} + T]$. It follows from Lemma 5.2 that $E_i(s) \leq d_i + (E_i(t_k + T) - d_i)e^{-\frac{1}{2}\beta_i(s-t_k-T)}$. By letting $d_i + (E_i(t_k + T) - d_i)e^{-\frac{1}{2}\beta_i(s-t_k-T)} \leq l_i - \frac{c_i}{\beta_i}(1 - e^{-\frac{1}{2}\beta_i(s-t_{k+1})})$ and rearranging this inequality we obtain that

$$[(E_i(t_k + T) - d_i)e^{\frac{1}{2}\beta_i T} - \frac{c_i}{\beta_i}e^{\frac{1}{2}\beta_i \delta}]e^{-\frac{1}{2}\beta_i(s-t_k)} \leq l_i - \frac{c_i}{\beta_i} - d_i. \quad (5.14)$$

Note that the equality is achieved when $s = t_k + T$. Therefore (5.14) holds $\forall s \in (t_k + T, t_{k+1} + T]$ if $(E_i(t_k + T) - d_i)e^{\frac{1}{2}\beta_i T} - \frac{c_i}{\beta_i}e^{\frac{1}{2}\beta_i \delta} \geq 0$ which is true if $l_i \geq \frac{c_i}{\beta_i} + d_i$.

Finally, constraint (5.12) trivially holds. Therefore $\tilde{x}_i(s; t_{k+1})$, $s \in [t_{k+1}, t_{k+1} + T]$ is a feasible trajectory. ■

5.3.2 Stability

In this subsection, stability of system (5.2) under the proposed control algorithm is analyzed

Theorem 5.2. *Suppose that Assumptions 5.1, 5.2, 5.3 and 5.4 hold, initial feasible control $u^0(t)$, $t \in [t_0, t_0 + T]$ is available at time instant t_0 and inequalities (5.6), (5.7) and (5.8) are satisfied. Then system (5.2) is asymptotically stable under the proposed control algorithm.*

Proof. Since by assumption, after x enters the terminal region Ω_ϵ , the terminal controller stabilizes the system, we only need to show that x will enter the terminal region in finite time.

Consider non-negative function $V_k = \int_{t_k}^{t_k+T} \|x^*(s; t_k) - x^0(s)\|_M ds$. Then according to constraint (5.12) we know that

$$V_k \leq \tilde{V}_k + \int_{t_k}^{t_k+T} \|x^*(s; t_k) - \tilde{x}(s; t_k)\|_M ds \leq \tilde{V}_k + \sum_{i=1}^N \xi_i T,$$

where $\tilde{V}_k \triangleq \int_{t_k}^{t_k+T} \|\tilde{x}(s; t_k) - x^0(s)\|_M ds$. Therefore, we have $V_{k+1} - V_k \leq \tilde{V}_{k+1} + \sum_{i=1}^N \xi_i T - V_k$.

By using triangle inequality and noticing that

$$\int_{t_{k+1}}^{t_k+T} \|x^*(s; t_k) - x^0(s)\|_M ds - V_k = - \int_{t_k}^{t_{k+1}} \|x^*(s; t_k) - x^0(s)\|_M ds,$$

it can be obtained that

$$\begin{aligned} \tilde{V}_{k+1} - V_k &\leq - \int_{t_k}^{t_{k+1}} \|x^*(s; t_k) - x^0(s)\|_M ds + \int_{t_{k+1}}^{t_k+T} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\|_M ds \\ &\quad + \int_{t_k+T}^{t_{k+1}+T} \|\tilde{x}(s; t_{k+1}) - x^0(s)\|_M ds. \end{aligned} \quad (5.15)$$

From

$$\int_{t_{k+1}}^{t_k+T} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\|_M ds \leq \sum_{i=1}^N \int_{t_{k+1}}^{t_k+T} \|\tilde{x}_i(s; t_{k+1}) - x_i^*(s; t_k)\|_{M_i} ds$$

and by the proof of Theorem 5.1 and some simple calculation, one can write

$$\begin{aligned} \int_{t_{k+1}}^{t_k+T} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\|_M ds &\leq \sum_{i=1}^N \int_{t_{k+1}}^{t_k+T} \frac{c_i}{\beta_i} (1 - e^{-\frac{1}{2}\beta_i\delta}) e^{-\frac{1}{2}\beta_i(s-t_{k+1})} ds \\ &= \sum_{i=1}^N \frac{2c_i}{\beta_i^2} (1 - e^{-\frac{1}{2}\beta_i\delta}) (1 - e^{-\frac{1}{2}\beta_i(T-\delta)}) \end{aligned}$$

and similarly

$$\int_{t_k+T}^{t_{k+1}+T} \|\tilde{x}(s; t_{k+1}) - x^0(s)\|_M ds \leq \sum_{i=1}^N d_i \delta + \frac{2}{\beta_i} (l_i - \frac{c_i}{\beta_i} (1 - e^{-\frac{1}{2}\beta_i(T-\delta)}) - d_i) \times (1 - e^{-\frac{1}{2}\beta_i \delta}).$$

Suppose that over time interval $[t_k, t_{k+1}]$, $\|x(s) - x^0(s)\|_M \geq \gamma\epsilon$. Then by Lemma 5.3, over this time interval, we have

$$\begin{aligned} \|x^*(s; t_k) - x^0(s)\|_M &\geq \|x(s) - x^0(s)\|_M - \|x(s) - x^*(s; t_k)\|_M \\ &\geq \gamma\epsilon - \sum_{i=1}^N \frac{c_i}{\beta_i} (1 - e^{-\frac{1}{2}\beta_i(s-t_k)}). \end{aligned}$$

Substituting the above inequalities into (5.15) leads to

$$\tilde{V}_{k+1} - V_k \leq -\gamma\epsilon\delta + \sum_{i=1}^N (\frac{c_i}{\beta_i} + d_i)\delta + \sum_{i=1}^N \frac{2}{\beta_i} (l_i - d_i - \frac{c_i}{\beta_i}) (1 - e^{-\frac{1}{2}\beta_i \delta})$$

and then it follows from inequality (5.8) that there exists a positive constant ρ_i such that

$$\xi_i T + \frac{2}{\beta_i} (l_i - d_i - \frac{c_i}{\beta_i}) (1 - e^{-\frac{1}{2}\beta_i \delta}) \leq \gamma_i \epsilon \delta - (d_i + \frac{c_i}{\beta_i}) \delta - \rho_i,$$

which implies

$$V_{k+1} - V_k \leq \sum_{i=1}^N \xi_i T + \tilde{V}_{k+1} - V_k \leq -\sum_{i=1}^N \rho_i.$$

Similar to Theorem 2 in [121], we can now claim that $x(s)$ converges to $\Omega_{x^0(s)}(\gamma\epsilon) \triangleq \{x \mid \|x - x^0(s)\|_M \leq \gamma\epsilon\}$ in finite time. On the other hand, by (5.9) and Lemma 5.1, $x^0(s)$ asymptotically converges to the origin. Therefore $x^0(s)$ converges to $\Omega_{(1-r)\epsilon}$ in finite time and stays in it thereafter, which implies that $x(s)$ converges to Ω_ϵ in finite time. This completes the proof. $\blacksquare$

Figure 5.1: Initial feasible state trajectory and control profile

5.4 Numerical Example

In this section, the proposed algorithm is applied to a building HVAC system with four neighbouring rooms, see [103], [170]. The building structure is the same as that in Chapter 4 but the model is nonlinear continuous-time. The system dynamics is given by

$$\begin{aligned}\dot{T}_1 &= \frac{60}{C}(\dot{m}_s^1 c_p (T_s - T_1) + \frac{T_2 - T_1}{R} + \frac{T_3 - T_1}{R} + \frac{T_e - T_1}{R_e}) \\ \dot{T}_2 &= \frac{60}{C}(\dot{m}_s^2 c_p (T_s - T_2) + \frac{T_1 - T_2}{R} + \frac{T_4 - T_2}{R} + \frac{T_e - T_2}{R_e}) \\ \dot{T}_3 &= \frac{60}{C}(\dot{m}_s^3 c_p (T_s - T_3) + \frac{T_1 - T_3}{R} + \frac{T_4 - T_3}{R} + \frac{T_e - T_3}{R_e})\end{aligned}$$

Figure 5.2: Comparison of the state trajectories and control profiles under centralized and distributed algorithms

Figure 5.3: State trajectories and control profiles under decentralized PI controller

$$\dot{T}_4 = \frac{60}{C}(\dot{m}_s^4 c_p (T_s - T_4) + \frac{T_2 - T_4}{R} + \frac{T_3 - T_4}{R} + \frac{T_e - T_4}{R_e})$$

where T_j is the temperature of room j , $C = 9163$ kJ/K is the lumped mass of each room, $c_p = 1012$ J/kg·K is the heat capacity of air, $R = 1$ K/kW is the heat resistance between neighbouring rooms, $R_e = 30$ K/kW is the heat resistance between rooms and the outside environment, $T_e = 30^\circ\text{C}$ is the outside temperature, $T_s = 16^\circ\text{C}$ is the temperature of the supply air, $\dot{m}_s^j$ is the air flow rate of room j . Our aim is to regulate the temperature of each room to the desired set point $T_d = 22^\circ\text{C}$ by adjusting air flow rate $\dot{m}_s^j \in [0.005 \ 3]\text{kg/s}$. The partitions of the states and inputs are $x_1 = [T_1 - T_d \ T_2 - T_d]^T$, $x_2 = [T_3 - T_d \ T_4 - T_d]^T$, $u_1 = [\dot{m}_s^1 - \bar{\dot{m}}_s \ \dot{m}_s^2 - \bar{\dot{m}}_s]^T$ and $u_2 = [\dot{m}_s^3 - \bar{\dot{m}}_s \ \dot{m}_s^4 - \bar{\dot{m}}_s]^T$ where $\bar{\dot{m}}_s = \frac{T_e - T_d}{(T_d - T_s)R_e c_p}$. Then the whole system can be rewritten as

$$\dot{x}_1 = f_1(x_1, x_2, u_1)$$

$$\dot{x}_2 = f_2(x_2, x_1, u_2).$$

The matrix M_i is chosen as $\begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}$ for $i = 1, 2$. Since the system is stable around the equilibrium, the terminal controller is chosen as $K = \mathbf{0}$. The parameters ϵ and r in Lemma 5.1 are respectively calculated as 0.5 and 0.4 according to Lemma 1 in [121]. β_i is calculated as 0.0267. The design parameters are chosen as: $\alpha = 0.3$, $l_i = 3$, $\delta = 5$, $T = 30$, $\xi_i = 0.5$, $\gamma_i = 0.49$, $i = 1, 2$ to satisfy conditions (5.6), (5.7) and (5.8). The objective functions are chosen as $J_i(x_i(t_k), \hat{u}_i(\cdot, t_k)) = \int_{t_k}^{t_k+T} (\|\hat{x}_i(s; t_k)\|^2 + 0.01\|\hat{u}_i(s; t_k) + \bar{m}_s\|^3)ds$, $i = 1, 2$, where the quadratic term represents the tracking error while the cubic term represents the power consumed by the supply fan [150]. The initial condition is given by $x_1(0) = (4, 3)$, $x_2(0) = (2, 1)$. Note that under these given parameters, a Lipschitz constant of f_i with respect to x_i is 0.0332 and no feasible parameters can be found to satisfy those sufficient conditions in [121] and [126]. However, by using the proposed method, we are able to obtain an optimal solution. Therefore compared with the proposed algorithms in [121] and [126] our algorithm is less conservative.

The initial feasible state trajectory and control input are shown in Fig. 5.1. The state responses and control inputs under the proposed distributed and centralized algorithms are shown in Fig. 5.2, where the solid line denotes the state response and control input under the distributed control while the dashed line represents those under the centralized one. It can be observed that the temperature response under the distributed algorithm is close to that under centralized control. Temperature trajectory and control profile under a decentralized PI controller given by $u_i(t) = 0.75x_i(t) + 0.01 \int_0^t x_i(s)ds$ is shown in Fig. 5.3. The performance index $\int_0^{50} (\|x_i(s)\|^2 + 0.01\|u_i(s) + \bar{m}_s\|^3)ds$, $i = 1, 2$ of centralized, distributed and PI controller are 1.1782×10^4 , 1.1785×10^4 and 1.8298×10^4 respectively.

Finally, the average computational time with respect to different discretization steps and algorithms is compared in Table 5.1. The algorithms used in this section are coded in Matlab[®] and run on a PC with Intel Core Duo i5-2400 CPU 3.10GHz. All examples are simulated in a continuous-time setting while the optimization problems are described by using ICLOCS [177] and solved by IPOPT [9].

Table 5.1: Average computational time of distributed and centralized algorithm

Steps	120	240	360	480	600
Distributed	1.37s	1.80s	2.14s	2.67s	3.02s
Centralized	5.63s	8.09s	11.62s	14.62s	18.29s

5.5 Conclusion

A novel distributed nonlinear continuous-time MPC algorithm has been proposed. In this algorithm, the traditional Lipschitz continuity condition has been replaced by a contraction condition which implies that there is a contractive tube around the nominal state trajectory. By leveraging the contraction condition, sufficient conditions have been derived to guarantee feasibility and asymptotic stability. A numerical example has been provided to show that the proposed algorithm is less conservative than those proposed in the literature.

Chapter 6

A Hierarchical Distributed MPC for HVAC Systems

In the previous chapter, we have developed a distributed MPC algorithm for nonlinear continuous-time systems. In this chapter we will apply this algorithm to HVAC systems in buildings. We will propose a hierarchical distributed MPC scheme based on the theory developed in the previous chapter. The building consists in multiple connected rooms and zones. The control objective is to keep the temperature of each room and zone at a given comfortable level with minimal energy consumption. This control scheme is divided into two levels. The upper level controller collects temperature and predictive information of all rooms and zones to generate reference trajectories while a lower level controller only uses local information to track the reference and optimize energy efficiency and thermal comfort. By using a contraction property of building's dynamics, recursive feasibility of the proposed algorithm is guaranteed.

This chapter is organized as follows. Section 6.1 outlines the proposed hierarchical distributed MPC control algorithm as well as the recursive feasibility analysis. Numerical examples and some discussions are given in Section 6.2. Finally, conclusions are drawn in Section 6.3.

6.1 Controller design

First we recall the simplified HVAC system model proposed in Chapter 3

$$C_i \dot{T}_i = u_i c_p (T_s - T_i) + \sum_{j \in \mathcal{N}_i} \frac{T_j - T_i}{R_{ij}} + \frac{T_{oa} - T_i}{R_{oa}^i} + P_d^i, \quad (6.1)$$

where T_i is the temperature of room i , C_i is the lumped mass of room i , u_i is the ventilation mass flow of the i -th room, c_p is the heat capacity of air, T_s is the temperature of supply air and T_{oa} is environment temperature, $R_{ij} = R_{ji}$ is the heat resistance between room i and j , R_{oa}^i is the heat resistance between room i and outdoor environment and P_d^i collects unmeasured disturbances in room i such as occupancy, usage of electronic devices and so on. Then equation (6.1) can be rewritten into a compact form as follows:

$$\dot{T}_i = f_i(T_i, T_{-i}, u_i, d_i),$$

where d_i collects T_{oa} and P_d^i and $-i$ denotes $\{j | j \in \mathcal{N}_i\}$.

Due to the development of modern methods to solve nonlinear optimization problems, in this section, we will use the simplified HVAC system model (6.1) to design the controller and analyze the recursive feasibility.

The prediction of outside temperature T_{oa} , which is denoted as $\tilde{T}_{oa}$, can be obtained from weather forecast station and the prediction of P_d^i , which is denoted as $\tilde{P}_d^i$ can be learned from historical data and schedule. In the following of this chapter, we assume that prediction errors are zero since the main purpose of this chapter is to introduce the hierarchical distributed structure and the compensation of bounded disturbances has been discussed in many other works [172].

The system states and control inputs are also subject to the following constraints:

1) $T_i \in [\underline{T}, \bar{T}] = [20, 24]^\circ C$. 2) $u_i \in [\underline{m}, \bar{m}] = [0.005, 5] kg/s$. The maximal mass flow rate is limited by the power of fans and the size of VAV boxes while the minimal mass flow rate is used to guarantee ventilation requirement.

The objective of this chapter is to design a distributed control law to maintain temperature T_i within a comfortable region while minimizing energy consumption. Because of the high computational complexity of the centralized MPC, it is not practical to apply it in real applications when the dimension of the system becomes high. Distributed MPC, which decomposes the centralized optimization problem into a group of smaller ones, makes it possible to be used for large scale dynamically coupled systems. However, purely distributed MPC algorithms are usually difficult to have convergence and feasibility guarantees [178] or those conditions are very conservative for application [121] [126] especially when the system is nonlinear. In this work we combine the distributed and centralized structure together to design a hierarchical control scheme to balance the computational complexity and the conservativeness.

The control structure is divided into two layers. The upper layer solves a global optimal control problem with long prediction horizon by collecting information from all subsystems to generate reference temperature trajectories for lower layer controllers. The lower layer controllers solve local optimal control problems with short prediction horizon by only using its local information. Both layers work in a receding horizon manner but the sampling period of the upper layer is much larger than that of the lower layer. Therefore, though the upper layer controller solves a global optimization problem, the computational burden is acceptable.

In this section we will make use of the contraction-based MPC developed in Chapter 5 to design the hierarchical distributed algorithm.

Contraction property is given in the following definition.

Definition 6.1. *Consider a subsystem $\mathcal{S}_i : \dot{x}_i = f_i(x_i, x_{-i}, u_i, d_i)$ where x_i is the state, $-i$ denotes $\{j | j \in \mathcal{N}_i\}$, u_i is the control input and d_i is the disturbance. If there exist positive matrix M_i and positive number β_i such that*

$$\frac{\partial f_i^T}{\partial x_i} M_i + M_i \frac{\partial f_i}{\partial x_i} \leq -\beta_i M_i$$

then subsystem $\mathcal{S}_i$ is contractive.

Figure 6.1: Control scheme

Then we have the following proposition.

Proposition 6.1. *Subsystem (3.2) is contractive.*

Proof. By choosing $M_i = 1$ and noticing that $u_i \geq \underline{\dot{m}}$, it can be calculated that

$$\frac{\partial f_i^T}{\partial x_i} M_i + M_i \frac{\partial f_i}{\partial x_i} = -2(u_i c_p + \sum_{j \in \mathcal{N}_i} \frac{1}{R_{ij}} + \frac{1}{R_{oa}^i})/C_i \leq -2(\underline{\dot{m}} c_p + \sum_{j \in \mathcal{N}_i} \frac{1}{R_{ij}} + \frac{1}{R_{oa}^i})/C_i$$

which completes the proof. ■

Lipschitz continuity is also required in the design procedure.

Assumption 6.1. f_i satisfies that

$$\|f_i(x_i, x_{-i}, u_i) - f_i(x_i, x'_{-i}, u_i)\| \leq \sum_{j \in \mathcal{N}_i} L_{ij} \|x_j - x'_j\|$$

Clearly, subsystem (3.2) satisfies the above assumption with $L_{ij} = \frac{1}{R_{ij}}$.

To facilitate the description of the proposed algorithm, the notation $\cdot^u$ will be used for upper layer MPC while $\cdot^l$ will be used for lower layer MPC. We shall abbreviate upper layer MPC controller to UMPC while the lower layer one of room i will be called LMPC i for short. t_k^u denotes the sampling time instant of the UMPC while t_k^l denotes that of the lower layer MPCs; define $c(k, p) = k(M + 1) + p$ where M is a positive integer corresponding to the number of sampling instants of LMPC between two sampling instants of UMPC and p is a positive integer; $t_k^u = t_{c(k,0)}^l$; $\delta^u \triangleq t_{k+1}^u - t_k^u$ and $\delta^l \triangleq t_{k+1}^l - t_k^l$ represent the sampling periods of the UMPC and LMPC respectively; $\delta^u = (M + 1)\delta^l$; T^u denotes the prediction horizon of the UMPC and T^l denotes that of LMPC which satisfies that $T^u \geq T^l + \delta^u$. The control scheme is illustrated in Fig. 6.1.

The hierarchical distributed MPC algorithm is summarized below:

- 1) At sampling time instant t_k^u , UMPC receives temperature measurement $T(t_k^u) = [T_1(t_k^u), \dots, T_N(t_k^u)]^T$.
- 2) UMPC computes the optimal control input

$$u^0(s; t_k^u) = [u_1^0(s; t_k^u), \dots, u_N^0(s; t_k^u)]^T, \quad s \in [t_k^u, t_k^u + T^u]$$

and the corresponding temperature trajectory

$$T^0(s; t_k^u) = [T_1^0(s; t_k^u), \dots, T_N^0(s; t_k^u)]^T, \quad s \in [t_k^u, t_k^u + T^u]$$

$T_i^0(s; t_k^u)$ is transmitted to LMPC i and its neighbours.

- 3) At sampling time instant $t_{c(k,p)}^l$, LMPC i receives local temperature measurement $T_i(t_{c(k,p)}^l)$ and computes the optimal control input $u_i^*(s; t_{c(k,p)}^l)$, $s \in [t_{c(k,p)}^l, t_{c(k,p)}^l + T^l]$.
- 4) Apply $u_i^*(s; t_{c(k,p)}^l)$, $s \in [t_{c(k,p)}^l, t_{c(k,p)}^l + \delta^l]$.
- 5) If $0 \leq p < M$, $p = p + 1$ and go to 3); else $k = k + 1$, $p = 0$ and go to 1).

$u^0(s; t_k^u)$, $s \in [t_k^u, t_k^u + T^u]$ is the optimal solution of the following optimization problem:

$$\min_{\hat{u}(\cdot)} J^u(T^u(t_k^u), \hat{u}(\cdot))$$

subject to

$$\begin{aligned}
C_i \dot{\hat{T}}_i^u(s) &= \hat{u}_i(s) c_p (T_s - \hat{T}_i^u(s)) + \frac{\sum_{j \in \mathcal{N}_i} (\hat{T}_j^u(s) - \hat{T}_i^u(s))}{R_{ij}} \\
&\quad + (\tilde{T}_{oa}(s) - \hat{T}_i^u(s)) / R_{oa}^i + \tilde{P}_d^i(s), \\
\hat{u}_i(s) &\in [\underline{\dot{m}}, \bar{\dot{m}}], \\
\hat{T}_i^u(s) &\in [\underline{T}, \bar{T}], \\
\hat{T}_i^u(t_k^u) &= T(t_k^u), \\
s &\in [t_k^u, t_k^u + T^u], \quad i = 1, \dots, N
\end{aligned}$$

where J^u represents performance index which will be optimized by UMPC.

Denote $r(s; t_k) = \frac{c_i}{\beta_i} (1 - e^{-\frac{1}{2}\beta_i(s-t_k)})$. After obtaining temperature trajectory

$$T^0(s; t_k^u) = [T_1^0(s; t_k^u), \dots, T_N^0(s; t_k^u)]^T, \quad s \in [t_k^u, t_k^u + T^u],$$

$u_i^*(s; t_{c(k,p)}^l)$, $s \in [t_{c(k,p)}^l, t_{c(k,p)}^l + T^l]$ is computed by solving the following optimization problem:

$$\min_{\hat{u}_i(\cdot)} J_i^l(T_i^l(t_{c(k,p)}^l), \hat{u}_i(\cdot))$$

subject to

$$\begin{aligned}
C_i \dot{\hat{T}}_i^l(s) &= \hat{u}_i(s) c_p (T_s - \hat{T}_i^l(s)) + \sum_{j \in \mathcal{N}_i} (T_j^0(s; t_k) - \hat{T}_i^l(s)) / R_{ij} \\
&\quad + \frac{(\tilde{T}_{oa}(s) - \hat{T}_i^l(s))}{R_{oa}^i} + \tilde{P}_d^i(s), \tag{6.2}
\end{aligned}$$

$$|\hat{T}_i^l(s) - T_i^0(s; t_k)| \leq l_i - r(s; t_{c(k,p)}^l), \tag{6.3}$$

$$\begin{aligned}
\hat{u}_i(s) &\in [\underline{\dot{m}}, \bar{\dot{m}}], \\
\hat{T}_i^l(t_{c(k,p)}^l) &= T_i(t_{c(k,p)}^l), \\
s &\in [t_{c(k,p)}^l, t_{c(k,p)}^l + T^l]
\end{aligned}$$

where J_i^l is the performance index which will be optimized by LMPC, $\beta_i = \frac{2}{C_i}(\underline{\dot{m}}c_p +$

$\sum_{j \in \mathcal{N}_i} \frac{1}{R_{ij}} + \frac{1}{R_{oa}^i}$), $c_i = 2 \sum_{j \in \mathcal{N}_i} \frac{l_j}{C_i R_{ij}}$ and l_i is a parameter to be chosen by designer. Denote the optimal control input of the above problem as $u_i^*(s; t_{c(k,p)}^l)$ and the corresponding optimal temperature trajectory as $T_i^*(s; t_{c(k,p)}^l)$.

The purpose of constraint (6.3) is to guarantee that the real temperature trajectory will not be far away from the reference. Therefore (6.2) is a reasonable approximate model of subsystem i .

Now we give the following proposition to ensure the recursive feasibility of the proposed hierarchical distributed MPC algorithm. Note that Lemma 5.2 and 5.3 still hold since each subsystem (3.2) satisfies Definition 6.1 and Assumption 6.1.

Proposition 6.2. *Consider a group of interconnected rooms with thermal dynamics described by (3.2). If the optimization problems of the UMPC formulated at t_k^u is feasible and $\frac{c_i}{\beta_i} \leq l_i$, then problems for LMPC i formulated at $t_{k(M+1)+p}$, $0 \leq p < M$ are feasible.*

Proof. Since UMPC is feasible, for any t_k^u , $T^0(s; t_k^u)$ and $u^0(s; t_k^u)$ are available. Clearly, when $p = 0$, subproblems formulated at $t_{c(k,0)}^l$ are feasible since $u_i^0(s; t_k)$, $s \in [t_k, t_k + T^l]$ is a feasible solution for each LMPC i . Now suppose that subproblem of LMPC i formulated at $p \geq 0$ has an optimal solution $u_i^*(s; t_{c(k,p)}^l)$, $i = 1, \dots, N$. We will show that subproblem formulated at $p + 1$ has a feasible solution given by

$$\tilde{u}_i(s; t_{c(k,p+1)}^l) = \begin{cases} u_i^*(s; t_{c(k,p)}^l), & s \in [t_{c(k,p+1)}^l, t_{c(k,p)}^l + T^l] \\ u_i^0(s; t_k), & s \in [t_{c(k,p)}^l + T^l, t_{c(k,p+1)}^l + T^l] \end{cases}$$

According to the definition of u_i^* and u_i^0 , it is obvious that $\tilde{u}_i \in [\underline{m}, \bar{m}]$. Then we will show that $\tilde{T}_i(s; t_{c(k,p+1)}^l)$ given by

$$\begin{aligned} C_i \dot{\tilde{T}}_i(s; t_{c(k,p+1)}^l) &= \tilde{u}_i(s; t_{c(k,p+1)}^l) c_p (T_s - \tilde{T}_i(s; t_{c(k,p+1)}^l)) \\ &\quad + \sum_{j \in \mathcal{N}_i} \frac{T_j^0(s; t_k) - \tilde{T}_i(s; t_{c(k,p+1)}^l)}{R_{ij}} \\ &\quad + \frac{\tilde{T}_{oa}(s) - \tilde{T}_i(s; t_{c(k,p+1)}^l)}{R_{oa}^i} + \tilde{P}_d^i(s) \end{aligned}$$

with initial condition $\tilde{T}_i(t_{c(k,p+1)}^l; t_{c(k,p+1)}^l) = T_i(t_{c(k,p+1)}^l)$ satisfies constraints (6.3).

Based on (6.3) and following similar arguments in Theorem 5.1, we have $|T_i(s) - T_i^0(s; t_k)| \leq l_i$. Then by considering $w(s) = T_{-i}(s) - T_{-i}^0(s; t_k)$ and some simple calculations we get

$$|\tilde{T}_i(t_{c(k,p+1)}^l; t_{c(k,p+1)}^l) - T_i^*(t_{c(k,p+1)}^l; t_{c(k,p)}^l)| \leq \frac{c_i}{\beta_i} (1 - e^{-\frac{1}{2}\beta_i \delta^l})$$

which leads to

$$|\tilde{T}_i(s; t_{c(k,p+1)}^l) - T_i^*(s; t_{c(k,p)}^l)| \leq \frac{c_i}{\beta_i} (1 - e^{-\frac{1}{2}\beta_i \delta^l}) e^{-\frac{1}{2}\beta_i (s - t_{c(k,p+1)}^l)},$$

where $s \in [t_{c(k,p+1)}^l, t_{c(k,p)}^l + T^l]$. On the other hand, by constraint (6.3), $|T_i^*(s; t_{c(k,p)}^l) - T_i^0(s; t_k)| \leq l_i - r(s; t_{c(k,p)}^l)$. Combining the above inequalities results in $|\tilde{T}_i(s; t_{c(k,p+1)}^l) - T_i^0(s; t_k)| \leq l_i - r(s; t_{c(k,p+1)}^l)$, $s \in [t_{c(k,p+1)}^l, t_{c(k,p)}^l + T^l]$. When $s \in [t_{c(k,p)}^l + T^l, t_{c(k,p+1)}^l + T^l]$, we check $|\tilde{T}_i(s; t_{c(k,p+1)}^l) - T_i^0(s; t_k)|$ directly. Note that $|\tilde{T}_i(t_{c(k,p)}^l + T^l; t_{c(k,p+1)}^l) - T_i^0(t_{c(k,p)}^l + T^l; t_k)| \leq l_i - r(t_{c(k,p)}^l + T^l; t_{c(k,p)}^l)$. Then based on (6.3) and following similar arguments in Theorem 5.1, $|\tilde{T}_i(s; t_{c(k,p+1)}^l) - T_i^0(s; t_k)| \leq (l_i - r(t_{c(k,p)}^l + T^l; t_{c(k,p)}^l)) e^{-\frac{1}{2}\beta_i (s - t_{c(k,p)}^l - T^l)} \leq l_i - r(s; t_{c(k,p+1)}^l)$. This proves that constraint (6.3) is satisfied. ■

6.2 Simulation and Discussion

In this section, a six-room model is used to simulate the temperature response under the proposed hierarchical distributed MPC algorithm. The six rooms are connected and the undirected graph is $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ where $\mathcal{V} = \{1, 2, 3, 4, 5, 6\}$ and $\mathcal{E} = \{(1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 1)\}$. System parameters are given: $C_i = 9.163 \times 10^3 \text{ kg/K}$, $T_s = 16^\circ \text{C}$, $R_{ij} = 4 \text{ K/kW}$, $R_{\alpha}^i = 57 \text{ K/kW}$. The design parameter l_i is chosen as 2. The prediction horizon of UMPC is 24 hours while that of LMPC*i*

is 2 hours. Sampling periods of UMPC and LMPC i are 1 hour and 15 minutes respectively. The objective function of UMPC is defined as

$$J^u(T(t_k^u), \hat{u}(\cdot; t_k^u)) = \int_{t_k^u}^{t_k^u + T^u} (0.01 \|\hat{T}^u(s; t_k^u) - T_d\|^2 + 100 \|\hat{u}(s; t_k^u)\|^2) ds$$

and the objective function of LMPC i is defined as

$$\begin{aligned} J_i^l(T_i(t_{c(k,p)}^l), \hat{u}_i(\cdot; t_{c(k,p)}^l)) = & \int_{t_k^u}^{t_k^u + T^l} (c_1 \|\hat{T}_i^l(s; t_{c(k,p)}^l) - T_d\|^2 + \|\hat{u}(s; t_k^u)\|^2 \\ & + c_2 \|\hat{T}_i^l(s; t_{c(k,p)}^l) - T_i^0(s; t_k^u)\|^2) ds \end{aligned}$$

where T_d is the desired temperature which is set as $21^\circ C$. Note that different types of objective functions are used in two layers. In particular, the upper layer mainly considers energy efficiency since it has a long prediction horizon while the lower layer can balance energy efficiency and thermal comfort by tuning c_1 and c_2 . Note that c_2 term is introduced to have some margin to ensure feasibility.

The simulation starts from 0:00 am to 23:59 pm. The environment temperature information in two days is downloaded from a local weather station and the temperature profile is plotted in Fig. 6.2.

In Fig. 6.3 thermal load profile P_d in two days is depicted. Here we assume that from 6:00 to 19:00 rooms are occupied so the thermal load is relatively higher while for the rest time of the day rooms are empty thus the thermal load is set to the minimum caused by only electronic devices. The thermal load for each room i is set as $P_d^i = (1 + (i - 1)/10)P_d$. The initial room temperature is set as $[21 \ 21 \ 21 \ 21 \ 21 \ 21]^\circ C$. We also set $c_1 = c_2 = 0.1$.

In the simulation, we compare the temperature responses under the proposed hierarchical distributed MPC and a decentralized PI control scheme with $u_i = (T_i - T_d) + 10^{-4} \int (T_i - T_d)$. Note that the temperature constraint $T_i \in [20, 24]^\circ C$ is satisfied in all examples. The proposed algorithm is coded in Matlab[®] and run on a PC with Intel Core Duo i5-2400 CPU 3.10GHz. The optimization problems are

Figure 6.2: Environment temperature

Figure 6.3: Thermal load

described by using ICLOCS [177] and solved by IPOPT [9]. The average time to solve a subproblem is 6.8 seconds.

In Fig. 6.4, temperature response and control input under the proposed algorithm are depicted. The temperature response and manipulated input under a decentralized PI control are shown in Fig. 6.5. The proposed algorithm starts to cool down room temperature before 4:00 while the decentralized PI keeps almost inactive. Between 5:00 and 6:00, the proposed algorithm exhibits a precooling effect which saves energy because of lower environment temperature and thermal load which leads to cheaper cooling. After 18:00, due to the predictive information, the proposed algorithm can avoid overcooling such that the room temperature will not be too low because of the lower environment temperature and thermal load. On the other hand, the decentralized PI controller continues to cool down room temperature and leads to some negative overshoot. It can also be observed that the proposed algorithm balances the tracking performance and energy saving. It does not track the desired temperature as well as the decentralized PI controller. Therefore the proposed algorithm saves more energy with some loss of tracking performance. However, it should be noticed that the temperature is still in the comfort zone so that the thermal comfort is not affected.

Considering the relatively small coefficients in the temperature related terms, we mainly compare the energy cost defined as $J_{energy} = \int_0^{T^l} \|u(s)\|^2 ds$ for the two control schemes. In the two examples, the energy costs J_{energy} for the proposed control and the decentralized PI control are 3.24×10^4 and 5.98×10^4 respectively. Clearly, the proposed controllers consume less energy because of the optimal nature of MPC and the PI controllers consume more energy though it has better tracking performance.

6.3 Conclusion

In this chapter, a hierarchical distributed MPC strategy has been proposed for building temperature regulation. A two layer control structure has been designed.

Figure 6.4: Temperature response and control input under the proposed algorithm

Figure 6.5: Temperature response and control input under decentralized PI algorithm

The upper level is a centralized coordinator, which collects global information and solves an optimal control problem to generate reference signals to be tracked by lower layer controllers. The reference signal is used to decouple the interconnected system. The lower level controllers only use local information to formulate and solve local optimization problems to track the reference trajectories while optimizing energy efficiency. Different objective functions could be used in the two levels. Numerical examples have been given to illustrate the performance of the proposed control scheme.

Chapter 7

Quantized Self-Triggered MPC

In previous chapters we assumed the communication channel is perfect. However, due to the limited bandwidth of the communication channel, conventional framework of MPC for continuous-time systems cannot be used here since the transmission of exact state measurement and continuous-time control input signal is not practical. In this chapter, a self-triggered MPC is proposed for a continuous-time nonlinear networked control system. By considering the quantization error as bound disturbance, a self-triggered scheme based on prediction error is proposed. The next sampling time is determined based on current information of the system and continuous measurement of the state is not necessary. The stability and feasibility are given for this framework.

In Section 7.1, the system model and the quantizer are introduced and the network-based control problem is formulated. In Section 7.2, the constrained optimal control problem is given and the self-triggered condition is designed. In Section 7.3, feasibility of the optimization problem and stability of the system under the proposed control algorithm are analyzed. In Section 7.4, numerical examples are given to illustrate the proposed approach. Finally, conclusions are drawn in Section 7.5.

Figure 7.1: Scheme of the networked control system

7.1 Problem formulation

7.1.1 System Definition

Consider a networked control system in Fig. 7.1, where the plant with sensor and the actuator of the system are connected with a model predictive controller through communication network. The dynamics of the nonlinear continuous-time system is given by

$$\dot{x}(t) = f(x(t), u(t)) + w(t), \quad (7.1)$$

where $x \in \mathbb{R}^n$ is the state of the system, $u \in \mathbb{U} \subset \mathbb{R}^m$ is the control input and $w \in \mathcal{W} \triangleq \{v \in \mathbb{R}^n \mid \|v\| \leq d\}$ is the bounded external disturbance. Here $\mathbb{U}$ and $\mathcal{W}$ are compact sets containing the origin as their interior point. The nonlinear function f satisfies that $f(0, 0) = 0$ and there exists positive constants L_x and L_u such that $\|f(x, u) - f(x', u')\| \leq L_x \|x - x'\| + L_u \|u - u'\|, \forall x \in \mathbb{R}^n, u \in \mathbb{U}$.

Since $f(0, 0) = 0$, system (7.1) can be linearized as follows:

$$\dot{x}(t) = Ax(t) + Bu(t) + w(t), \quad (7.2)$$

where $A = \frac{\partial f}{\partial x}|_{x=0}$, $B = \frac{\partial f}{\partial u}|_{u=0}$.

We assume the system satisfies the following assumption:

Assumption 7.1. *The system (7.2) with $w(t) = 0$ is stabilizable.*

Based on this assumption, we know that there exists a matrix $K \in \mathbb{R}^{m \times n}$ such that $A + BK$ is stable. Then following the same procedure in [76], [121], [179] we have the following lemma:

Lemma 7.1. *For system (7.1), suppose that Assumption 7.1 holds. Given two matrices $R > 0$ and $Q > 0$, there exists a state-feedback gain K , a constant $\epsilon > 0$ and a matrix $P > 0$ such that: 1) the set $\Omega(\epsilon) \triangleq \{x | V(x) \leq \epsilon^2\}$ is an invariant set for the system $\dot{x}(t) = f(x(t), Kx(t))$; 2) $\dot{V}(x(t))|_{\dot{x}(t)=f(x(t), Kx(t))} \leq -\|x(t)\|_{Q^*}^2$ and $Kx(t) \in \mathbb{U}$, $\forall x(t) \in \Omega(\epsilon)$. Here $V(x(t)) = \|x\|_P^2$ and $Q^* \triangleq Q + K^T R K$.*

7.1.2 Quantizer

In most existing MPC strategies for continuous-time systems, a continuous-time control signal will be transmitted to the actuator and applied until the next sampling time instant. However, in a networked control framework, an unlimited bandwidth is required to transmit such a continuous-time signal which is impossible in real applications. Even though in some other literatures [78], [175] sample-and-hold mechanism is proposed to overcome this issue, it was assumed that the data transmitted to the actuator is accurate which also requires unlimited bandwidth of the communication channel. Inspired by these deficiencies, we introduce a quantizer to quantize the continuous-time control signal as well as the state information before they are transmitted to the actuator and the controller respectively. The quantizer is a function $q : \mathbb{R} \rightarrow \mathbb{V} := \{0, \pm\delta, \pm 2\delta, \dots, \pm p\delta\}$ defined as follows:

$$q(u) = \begin{cases} p\delta, & \text{if } (p - \frac{1}{2})\delta \leq u, \\ (p-1)\delta, & \text{if } (p - \frac{3}{2})\delta \leq u < (p - \frac{1}{2})\delta, \\ \vdots, & \vdots, \\ 0, & \text{if } (-\frac{1}{2}\delta) \leq u < \frac{1}{2}\delta, \\ \vdots, & \vdots, \\ -p\delta, & \text{if } u < -(p - \frac{1}{2})\delta \end{cases}$$

where δ is the quantization step and p is the quantization level. We denote δ_u and δ_x as the quantization step for control signal and state and q_u and q_x as the

corresponding quantizer, respectively. Then the quantized control signal $u_q(t)$ and the quantized state $x_q(t)$ are given as follows:

$$\begin{aligned} u_q(t) &= (q_u(u_1(t)), \dots, q_u(u_m(t)))^T, \\ x_q(t) &= (q_x(x_1(t)), \dots, q_x(x_n(t)))^T. \end{aligned}$$

Note that saturation problem can be avoided if the control constraint $\mathbb{U}$ and state constraint are properly chosen.

The objective of this chapter is to design a controller to stabilize the system (7.1) to a neighborhood of the origin by using quantized state information and control input. In the following of this chapter, we will adopt an MPC framework to achieve this goal.

7.2 Self-triggered MPC design

7.2.1 Optimal control problem

At every sampling time instant t_k , the controller receives the quantized measurement of the system state $x_q(t_k)$ and solves the following optimal control problem (OCP):

$$\min_{\hat{u}(s; t_k) \in \mathbb{U}} J_k(\hat{x}(t_k; t_k), \hat{u}(s; t_k))$$

subject to

$$\begin{aligned} \hat{x}(t_k; t_k) &= x_q(t_k) \\ \dot{\hat{x}}(s; t_k) &= f(\hat{x}(s; t_k), \hat{u}(s; t_k)) \\ \hat{x}(t_k + T; t_k) &\in \Omega(\alpha\epsilon) \end{aligned} \tag{7.3}$$

where $J_k(\hat{x}(t_k; t_k), \hat{u}(s; t_k)) = \int_{t_k}^{t_k+T} (\|\hat{x}(\tau; t_k)\|_Q^2 + \|\hat{u}(\tau; t_k)\|_R^2) d\tau + \|\hat{x}(t_k + T; t_k)\|_P^2$, $x_q(t_k) = q(x(t_k))$, P , Q and R are positive definite matrices and $\alpha \in (0, 1)$ and $\Omega(\cdot)$ is defined in Lemma 7.1.

In this OCP design we do not consider state constraints but it can be adapted easily by using constraint tighten technique as [175].

7.2.2 Self-triggered condition

Denote the optimal solution of the OCP as $u^*(s; t_k)$, $s \in [t_k, t_k + T]$ and the corresponding state trajectory $x^*(s; t_k)$, $s \in [t_k, t_k + T]$.

After obtaining the quantized control signal, we define a virtual state trajectory $x_q(s; t_k)$ by the following equation

$$\dot{x}_q(s; t_k) = f(x_q(s; t_k), u_q^*(s; t_k), 0), \quad x_q(t_k; t_k) = x_q(t_k)$$

Then the next sampling time instant t_{k+1} is given as follows:

$$\begin{aligned} \tilde{t}_{k+1} &= \inf\{\hat{t} | \hat{t} > t_k, \|x^*(\hat{t}; t_k) - x_q(\hat{t}; t_k)\| = \sigma_1\}, \\ t_{k+1} &= \min\{\tilde{t}_{k+1}, \frac{1}{L_x} \ln\left(\frac{L_x \sigma_2 + d}{L_x \sqrt{n} \delta_x + d}\right) + t_k\}, \end{aligned} \quad (7.4)$$

where σ_1 and σ_2 are two positive constants which satisfy

$$T \geq \frac{1}{L_x} \ln\left(\frac{L_x \sigma_2 + d}{L_x \sqrt{n} \delta_x + d}\right) \geq \frac{1}{L_x} \left(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u}\right). \quad (7.5)$$

To avoid Zeno behavior, we need to guarantee that the inter-execution time will not be arbitrarily small. To this end, the following theorem gives a lower bound on the inter-execution time.

Theorem 7.1. *For system (7.1), if the sampling time instant t_k is determined by (7.4), then the inter-execution time $\delta \triangleq t_{k+1} - t_k$ satisfies that $\frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u}) \leq \delta \leq \frac{1}{L_x} \ln(\frac{L_x \sigma_2 + d}{L_x \sqrt{n} \delta_x + d})$.*

Proof. By the definition of x^* and x_q we have the following:

$$x^*(s; t_k) = x_q(t_k) + \int_{t_k}^s f(x^*(\tau; t_k), u^*(\tau; t_k), 0) d\tau$$

$$x_q(s; t_k) = x_q(t_k) + \int_{t_k}^s f(x_q(\tau; t_k), u_q^*(\tau; t_k), 0) d\tau,$$

which leads to

$$\|x^*(s; t_k) - x_q(s; t_k)\| \leq L_x \int_{t_k}^s (\|x^*(\tau; t_k) - x_q(\tau; t_k)\| + L_u \sqrt{m} \delta_u) d\tau.$$

Then by Gronwall-Bellman inequality it results in

$$\|x^*(s; t_k) - x_q(s; t_k)\| \leq \frac{L_u \sqrt{m} \delta_u}{L_x} (e^{L_x(s-t_k)} - 1).$$

Letting $\frac{L_u \sqrt{m} \delta_u}{L_x} (e^{L_x(s-t_k)} - 1) = \sigma_1$ leads to $\tilde{t}_{k+1} - t_k \geq \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u})$. Since $\frac{L_x \sigma_2 + d}{L_x \sqrt{n} \delta_x + d} \geq 1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u}$, one can conclude that $t_{k+1} - t_k \geq \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u})$. From (7.4) it is obvious that $t_{k+1} - t_k \leq \frac{1}{L_x} \ln(\frac{L_x \sigma_2 + d}{L_x \sqrt{n} \delta_x + d})$ which completes the proof. ■

7.3 Analysis

7.3.1 Feasibility

Since MPC is based on an optimization problem, the control signal is dependent on the feasibility of this problem. Therefore we establish the following results to avoid this situation.

Lemma 7.2. *For system (7.1), if the sampling time instant is given by (7.4), then*

$$\|x(t_{k+1}) - x^*(t_{k+1}; t_k)\| \leq \sigma_1 + \sigma_2.$$

Proof. From Leibniz formula we have

$$\begin{aligned} x(s) &= x(t_k) + \int_{t_k}^s f(x(\tau), u_q(\tau; t_k), w(\tau)) d\tau \\ x_q(s; t_k) &= x_q(t_k) + \int_{t_k}^s f(x_q(\tau; t_k), u_q(\tau; t_k), 0) d\tau \end{aligned}$$

By Gronwall-Bellman inequality it results in

$$\|x(s) - x_q(s; t_k)\| \leq \frac{1}{L_x} ((L_x \sqrt{n} \delta_x + d) e^{L_x(s-t_k)} - d).$$

Since $t_{k+1} - t_k \leq \frac{1}{L_x} \ln(\frac{L_x \sigma_2 + d}{L_x \sqrt{n} \delta_x + d})$, $\|x(t_{k+1}) - x_q(t_{k+1}; t_k)\| \leq \sigma_2$. By combining it with the trigger condition (7.4), one can obtain that

$$\begin{aligned} \|x(t_{k+1}) - x^*(t_{k+1}; t_k)\| &\leq \|x(t_{k+1}) - x_q(t_{k+1}; t_k)\| + \|x_q(t_{k+1}; t_k) - x^*(t_{k+1}; t_k)\| \\ &\leq \sigma_1 + \sigma_2. \end{aligned}$$

■

Theorem 7.2. *For system (7.1), suppose Assumption 7.1 holds and at time t_0 OCP is feasible. Then OCP is feasible for all time t_k , $k \geq 1$, provided that the following inequalities hold:*

$$\bar{\lambda}(\sqrt{P}) \frac{(\sigma_1 + \sigma_2 + \sqrt{n} \delta_x) e^{L_x T} L_u \sqrt{m} \delta_u}{L_u \sqrt{m} \delta_u + \sigma_1 L_x} \leq (1 - \alpha) \epsilon, \quad (7.6)$$

$$\frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{u} \delta_u}) + \frac{2\bar{\lambda}(P) \ln \alpha}{\underline{\lambda}(Q^*)} \geq 0. \quad (7.7)$$

Proof. Induction will be used to show this result. Given the optimal control obtained at time t_k is $u^*(s; t_k)$ and the corresponding state trajectory $x^*(s; t_k)$, $s \in [t_k, t_k + T]$. We will show that

$$\tilde{u}(s; t_{k+1}) = \begin{cases} u^*(s; t_k), & s \in [t_{k+1}, t_k + T] \\ K \tilde{x}(s; t_{k+1}), & s \in (t_k + T, t_{k+1} + T] \end{cases}$$

is a feasible control input for time instant t_{k+1} . Here $\tilde{x}(s; t_{k+1})$ is defined as

$$\dot{\tilde{x}}(s; t_{k+1}) = f(\tilde{x}(s; t_{k+1}), \tilde{u}(s; t_{k+1})),$$

with initial condition $\tilde{x}(t_{k+1}; t_{k+1}) = x_q(t_{k+1})$.

First, since $u^*(s; t_k)$ is the optimal solution at time t_k , $u^*(s; t_k) \in \mathbb{U}$, $s \in [t_{k+1}, t_k + T]$

is obvious.

Then we need to show that $K\tilde{x}(s; t_{k+1}) \in \mathbb{U}$, $s \in (t_k + T, t_{k+1} + T]$. By Lemma 7.2 and $\|x(t_{k+1}) - x_q(t_{k+1})\| \leq \sqrt{n}\delta_x$ we can get $\|x_q(t_{k+1}) - x^*(t_{k+1}; t_k)\| \leq \sigma_1 + \sigma_2 + \sqrt{n}\delta_x$. Hence, we have

$$\begin{aligned} \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\| &\leq \|x_q(t_{k+1}) - x^*(t_{k+1}; t_k)\| \\ &\quad + L_x \int_{t_{k+1}}^s \|\tilde{x}(\tau; t_{k+1}) - x^*(\tau; t_k)\| d\tau, \quad s \in [t_{k+1}, t_k + T] \end{aligned}$$

which leads to $\|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\| \leq (\sigma_1 + \sigma_2 + \sqrt{n}\delta_x)e^{L_x(s-t_{k+1})}$. Furthermore, since $t_{k+1} - t_k \geq \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m}\delta_u})$, we have

$$t_k + T - t_{k+1} \leq T - \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m}\delta_u})$$

and

$$\|\tilde{x}(t_k + T; t_{k+1}) - x^*(t_k + T; t_k)\| \leq \frac{(\sigma_1 + \sigma_2 + \sqrt{n}\delta_x)e^{L_x T} L_u \sqrt{m}\delta_u}{L_u \sqrt{m}\delta_u + \sigma_1 L_x}.$$

From (7.3) we know that $\|x^*(t_k + T; t_k)\|_P^2 \leq \alpha^2 \epsilon^2$. Then by using inequality (7.6) one can obtain that

$$\|\tilde{x}(t_k + T; t_{k+1})\|_P \leq \|x^*(t_k + T; t_k)\|_P + \bar{\lambda}(\sqrt{P}) \|\tilde{x}(t_k + T; t_{k+1}) - x^*(t_k + T; t_k)\| \leq \epsilon.$$

This implies $\tilde{x}(t_k + T; t_{k+1})$ enters $\Omega(\epsilon)$ and according to Lemma 7.1, $u = K\tilde{x}$ can stabilize the system while keeping $\tilde{x}$ inside $\Omega(\epsilon)$ and $K\tilde{x} \in \mathbb{U}$.

The last thing we need to show is that $\tilde{x}(t_{k+1} + T; t_{k+1}) \in \Omega(\alpha\epsilon)$. By using Lemma 7.1 again we have

$$\|\tilde{x}(s; t_{k+1})\|_P^2 \leq \|\tilde{x}(t_k + T; t_{k+1})\|_P^2 e^{-\frac{\lambda(Q^*)}{\lambda(P)}(s-t_k-T)} \leq \epsilon^2 e^{-\frac{\lambda(Q^*)}{\lambda(P)}(s-t_k-T)}.$$

Since $t_{k+1} - t_k \geq \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m}\delta_u}) \geq -\frac{2\bar{\lambda}(P) \ln \alpha}{\lambda(Q^*)}$, $\|\tilde{x}(t_{k+1} + T; t_{k+1})\|_P^2 \leq \alpha^2 \epsilon^2$ which means $\tilde{x}(t_{k+1} + T; t_{k+1}) \in \Omega(\alpha\epsilon)$. ■

7.3.2 Stability

To analyze the stability, first we need to show that if $x \notin \Omega(\epsilon)$, x will enter $\Omega(\epsilon)$ in finite time.

Theorem 7.3. *For system (7.1), suppose that Assumption 7.1 holds. If $x \notin \Omega(\epsilon)$, then x will converge into $\Omega(\epsilon)$ in finite time, provided that the following inequality holds:*

$$\begin{aligned} & \frac{(L_Q + L_{V_f} L_x) L_u \sqrt{m} \delta_u (\sigma_1 + \sigma_2 + \sqrt{n} \delta_x) e^{L_x T}}{L_x (L_u \sqrt{m} \delta_u + \sigma_1 L_x)} \\ & < \underline{\lambda}(Q) \left(\frac{\epsilon}{\bar{\lambda}(P^{0.5})} - \sigma_1 - \sigma_2 \right)^2 \frac{1}{L_x} \ln \left(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u} \right) \\ & \quad + \frac{L_Q (\sigma_1 + \sigma_2 + \sqrt{n} \delta_x)}{L_x}, \end{aligned} \quad (7.8)$$

where L_Q and L_{V_f} are the Lipschitz constants of $\|\cdot\|_Q^2$ and $\|\cdot\|_P^2$ respectively.

Proof. Denote the optimal value of $J_k(\hat{x}(t_k; t_k), \hat{u}(s; t_k))$ as J_k^* and $J_{k+1}(\hat{x}(t_{k+1}; t_{k+1}), \tilde{u}(s; t_{k+1}))$ as $\tilde{J}_{k+1}$. Then by the optimality of J_{k+1}^* , we know that $J_{k+1}^* \leq \tilde{J}_{k+1}$. Therefore $J_{k+1}^* - J_k^* \leq \tilde{J}_{k+1} - J_k^*$. Note that

$$\begin{aligned} \tilde{J}_{k+1} &= \int_{t_{k+1}}^{t_k+T} (\|\tilde{x}(s; t_{k+1})\|_Q^2 + \|u^*(s; t_k)\|_R^2) ds \\ & \quad + \int_{t_k+T}^{t_{k+1}+T} (\|\tilde{x}(s; t_{k+1})\|_Q^2 + \|K\tilde{x}(s; t_{k+1})\|_R^2) ds + \|\tilde{x}(t_{k+1} + T; t_{k+1})\|_P^2 \end{aligned}$$

and

$$\begin{aligned} J_k^* &= \int_{t_{k+1}}^{t_k+T} (\|x^*(s; t_k)\|_Q^2 + \|u^*(s; t_k)\|_R^2) ds \\ & \quad + \int_{t_k}^{t_{k+1}} (\|x^*(s; t_k)\|_Q^2 + \|u^*(s; t_k)\|_R^2) ds + \|x^*(t_k + T; t_k)\|_P^2. \end{aligned}$$

It can be obtained that

$$\tilde{J}_{k+1} - J_k^* = \Delta_1 + \Delta_2 + \Delta_3$$

where

$$\begin{aligned}\Delta_1 &= - \int_{t_k}^{t_{k+1}} (\|x^*(s; t_k)\|_Q^2 + \|u^*(s; t_k)\|_R^2) ds, \\ \Delta_2 &= \int_{t_{k+1}}^{t_k+T} (\|\tilde{x}(s; t_{k+1})\|_Q^2 + \|u^*(s; t_k)\|_R^2) ds - \int_{t_{k+1}}^{t_k+T} (\|x^*(s; t_k)\|_Q^2 + \|u^*(s; t_k)\|_R^2) ds, \\ \Delta_3 &= \int_{t_k+T}^{t_{k+1}+T} (\|\tilde{x}(s; t_{k+1})\|_Q^2 + \|K\tilde{x}(s; t_{k+1})\|_R^2) ds + \|\tilde{x}(t_{k+1} + T; t_{k+1})\|_P^2 \\ &\quad - \|x^*(t_k + T; t_k)\|_P^2.\end{aligned}$$

Since $x(s) \notin \Omega(\epsilon)$, $s \in [t_k, t_{k+1}]$, by Lemma 7.2 we have $\|x^*(s; t_k)\| \geq \frac{\epsilon}{\lambda(P^{0.5})} - \sigma_1 - \sigma_2$.

From Theorem 7.1 we know that $t_{k+1} - t_k \geq \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u})$. Then we get

$$\Delta_1 \leq -\lambda(Q) \left(\frac{\epsilon}{\lambda(P^{0.5})} - \sigma_1 - \sigma_2 \right)^2 \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u}).$$

For Δ_2 , it is easy to see that

$$\begin{aligned}\Delta_2 &\leq \int_{t_{k+1}}^{t_k+T} L_Q \|\tilde{x}(s; t_{k+1}) - x^*(s; t_k)\| ds \leq \int_{t_{k+1}}^{t_k+T} L_Q (\sigma_1 + \sigma_2 + \sqrt{n} \delta_x) e^{L_x(s-t_{k+1})} ds \\ &\leq \frac{L_Q (\sigma_1 + \sigma_2 + \sqrt{n} \delta_x)}{L_x} (e^{L_x(T-\delta)} - 1).\end{aligned}$$

By Theorem 7.1 again, we know that

$$T - \delta \leq T - \frac{1}{L_x} \ln(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u})$$

and

$$\Delta_2 \leq \frac{L_Q (\sigma_1 + \sigma_2 + \sqrt{n} \delta_x)}{L_x} \left(\frac{e^{L_x T}}{1 + \frac{\sigma_1 L_x}{L_u \sqrt{m} \delta_u}} - 1 \right).$$

Finally we estimate Δ_3 . Since

$$\begin{aligned}\Delta_3 &= \int_{t_k+T}^{t_{k+1}+T} (\|\tilde{x}(s; t_{k+1})\|_Q^2 + \|K\tilde{x}(s; t_{k+1})\|_R^2) ds + \|\tilde{x}(t_{k+1} + T; t_{k+1})\|_P^2 \\ &\quad - \|\tilde{x}(t_k + T; t_{k+1})\|_P^2 + \|\tilde{x}(t_k + T; t_{k+1})\|_P^2 - \|x^*(t_k + T; t_k)\|_P^2,\end{aligned}$$

by Lemma 7.1 we have

$$\int_{t_k+T}^{t_{k+1}+T} (\|\tilde{x}(s; t_{k+1})\|_Q^2 + \|K\tilde{x}(s; t_{k+1})\|_R^2) ds + \|\tilde{x}(t_{k+1} + T; t_{k+1})\|_P^2 - \|\tilde{x}(t_k + T; t_{k+1})\|_P^2 \leq 0.$$

$$\text{Therefore } \Delta_3 \leq L_{V_f} \|\tilde{x}(t_k + T; t_{k+1}) - x^*(t_k + T; t_k)\| \leq \frac{L_{V_f}(\sigma_1 + \sigma_2 + \sqrt{n}\delta_x)e^{L_x T} L_u \sqrt{m}\delta_u}{L_u \sqrt{m}\delta_u + \sigma_1 L_x}.$$

By combining the above inequalities we can get that

$$\begin{aligned} \tilde{J}_{k+1} - J_k^* &\leq -\frac{\lambda(Q)}{L_x} \left(\frac{\epsilon}{\lambda(P^{0.5})} - \sigma_1 - \sigma_2 \right)^2 \ln \left(1 + \frac{\sigma_1 L_x}{L_u \sqrt{m}\delta_u} \right) \\ &\quad + \frac{L_u \sqrt{m}\delta_u e^{L_x T} L_Q (\sigma_1 + \sigma_2 + \sqrt{n}\delta_x)}{L_x (L_u \sqrt{m}\delta_u + \sigma_1 L_x)} \\ &\quad - \frac{L_Q (\sigma_1 + \sigma_2 + \sqrt{n}\delta_x)}{L_x} \\ &\quad + \frac{L_{V_f} (\sigma_1 + \sigma_2 + \sqrt{n}\delta_x) e^{L_x T} L_u \sqrt{m}\delta_u}{L_u \sqrt{m}\delta_u + \sigma_1 L_x} \end{aligned}$$

From (7.8) we know that there exists a positive constant η such that $\tilde{J}_{k+1} - J_k^* \leq -\eta$. By contradiction, we can show that x will enter $\Omega(\epsilon)$ in finite time. Otherwise, $J_{k+1}^* \leq J_0^* - k\eta \rightarrow -\infty$ as $k \rightarrow \infty$ which contradicts the fact that $J_k^* \geq 0$. ■

Then we have the following ultimately bounded result.

Theorem 7.4. *For system (7.1) under the proposed control algorithm, if Assumption 7.1 holds and design parameters $\sigma_1, \sigma_2, \delta_x, \delta_u, Q, R, P$ and T are chosen to satisfy conditions (7.5), (7.6), (7.7) and (7.8), then the state trajectory will converge to the set $\Omega(\bar{\epsilon})$, where $\bar{\epsilon} = \sqrt{\frac{2\lambda(P)\|\sqrt{P}\|}{\lambda(Q^*)}} \epsilon d$.*

Proof. Theorem 7.3 shows that if $x \notin \Omega(\epsilon)$, then x will enter $\Omega(\epsilon)$ in finite time. Therefore to prove this theorem we only consider the case when $x \in \Omega(\epsilon)$. According to the dual-mode strategy, after x enters $\Omega(\epsilon)$, $u = Kx$ is applied. Taking the Lyapunov functional as $V(x(t)) = \|x(t)\|_P^2$ we have $\dot{V}(x(t)) \leq -\|x\|_{Q^*}^2 + 2x^T(t)Pw(t) \leq \frac{\lambda(Q^*)}{\lambda(P)} (-\|x(t)\|_P^2 + \bar{\epsilon}^2)$. Then the following arguments directly follow from those in [179] and they are omitted here. ■

Remark 7.1. *Since in this work we consider a general class of nonlinear systems with only a few mild assumptions, the sufficient conditions (7.6), (7.7) and (7.8) are rather conservative. Future work should be to reduce the conservativeness for certain kind of systems to obtain larger bounds on the quantization steps and disturbances. One possible way is to introduce a feedback controller $h(x)$ and let the control input $u = h(x) + v$ where v is a new decision variable. If $h(x)$ has some good property such as reducing the Lipschitz constant of the closed-loop system, we may have more flexibility on system design.*

7.4 Numerical Examples

In this section, two examples are given to illustrate the proposed self-triggered scheme. The open-loop constrained optimal control problem is described by using ICLOCS [177] and solved by IPOPT [9].

Example 7.1. *Consider a first-order linear system*

$$\dot{x}(t) = x(t) + 4u(t) + w(t),$$

where $x(t) \in \mathbb{R}$ is the system state which can be measured, $u(t) \in \mathbb{R}$ is the control input and $w(t) \in \mathbb{R}$ is the external disturbance. The control constraint is given by

$$-1 \leq u(t) \leq 1, \quad \forall t \geq 0.$$

The weighted matrices Q , R and P are chosen as $Q = 1$, $R = 1$ and $P = 0.3202$. The terminal controller is designed as $u = -1.2808x$ and the corresponding terminal constraint is chosen as $P\|x\|^2 \leq 0.1952$. The prediction horizon is chosen as $T = 0.3$. The Lipschitz constants of the system are $L_x = 1$ and $L_u = 4$. The intensity of the disturbance is chosen as $\|w(t)\| \leq 10^{-4}$. The quantization steps of the control input and the state measurement are designed as $\delta_u = 0.06$ and $\delta_x = 10^{-4}$ while the triggering level is designed as $\sigma_1 = 3\delta_u$ and $\sigma_2 = 1.3\delta_x$. In Fig. 7.2 and Fig. 7.3, we

Figure 7.2: Comparison of the state trajectories under different control schemes

compare the state trajectories and the control inputs under the proposed algorithm and the time-triggered MPC. By using the proposed algorithm, the average update interval is 0.14 while the update interval of the time-triggered MPC is set as 0.08. However, we can see from Fig. 7.2 that the performances of the two algorithms are very close to each other.

Example 7.2. In this example, we consider an HVAC system which consists of two connected rooms. In each room a cool air inlet is equipped. The cool air flow rate can be adjusted to control the indoor temperature. The system dynamics is given by [103]:

$$\begin{aligned}\dot{T}_1(t) &= \frac{60}{C}(\dot{m}_s^1(t)c_p(T_s - T_1(t)) + \frac{T_2(t) - T_1(t)}{R} + \frac{T_e - T_1(t)}{R_e}), \\ \dot{T}_2(t) &= \frac{60}{C}(\dot{m}_s^2(t)c_p(T_s - T_2(t)) + \frac{T_1(t) - T_2(t)}{R} + \frac{T_e - T_2(t)}{R_e}),\end{aligned}$$

where T_1 and T_2 are the temperature of room 1 and room 2, $C = 9163$ kJ/K is the lumped mass of each room, $c_p = 1012$ J/kg · K is the heat capacity of air, $R = 1$ K/kW is the heat resistance between neighbouring rooms, $R_e = 30$ K/kW is the heat resistance between rooms and the outside environment, $T_e = 30^\circ\text{C}$ is the

Figure 7.3: Comparison of the control inputs under different control schemes

outside temperature, $T_s = 16^\circ\text{C}$ is the temperature of the supplied cool air and $\dot{m}_s^1$ and $\dot{m}_s^2$ are the air flow rate of the corresponding rooms. The desired temperature is $T_d = 22^\circ\text{C}$ and the control objective is to drive the room temperature to the desired one. Then the system can be rewritten as follows:

$$\dot{x}(t) = f(x(t), u(t)),$$

where $x = (T_1 - T_d \ T_2 - T_d)^T$, $u = (\dot{m}_s^1 - \bar{\dot{m}}_s \ \dot{m}_s^2 - \bar{\dot{m}}_s)^T$, $\bar{\dot{m}}_s = \frac{T_e - T_d}{(T_d - T_s)Re c_p}$. The control constraint is

$$0 \leq \dot{m}_s^i(t) \leq 3, \ i = 1, 2, \ \forall t \geq 0.$$

The weighted matrices are chosen as $Q = 0.1I_2$, $R = 0.01I_2$ and $P = 0.5I_2$. The terminal controller is chosen as $K = 0$ and the terminal constraint is chosen as $\|x\|^2 \leq 0.25$. The prediction horizon is set as $T = 30$ minutes. The Lipschitz constants of the system are $L_x = 0.0332$ and $L_u = 0.0265$. According to sufficient conditions (7.6), (7.7) and (7.8), the quantization steps of the control input and the state measurement should be $\delta_u = 0.002$ and $\delta_x = 10^{-4}$ while the triggering level is designed as $\sigma_1 = 3\delta_u$ and $\sigma_2 = 1.3\delta_x$. However, in the simulation example, we set $\delta_u = 0.5$ and the algorithm still works well. In Fig. 7.4 and Fig. 7.5, we compare the

Figure 7.4: Comparison of the state trajectories under different control schemes

state trajectories and the control inputs under the proposed algorithm and the time-triggered MPC. By using the proposed algorithm, the average update interval is 18.7 minutes while the update interval of the time-triggered MPC is set as 10 minutes. However, we can see from Fig. 7.4 that the performances of the two algorithms are very close to each other.

7.5 Conclusion

In this chapter, a self-triggered MPC framework for networked nonlinear continuous-time system has been proposed. The sensor and the actuator of the plant are connected with the model predictive controller through communication networks. Exact state and control input could not be transmitted due to limited bandwidth. By considering the quantization errors as bounded disturbances, the robustness of the proposed control scheme has been analyzed. Sufficient conditions on stability of the resulted close-loop system and feasibility of the constrained optimal control problem have been given. Zeno behavior has been avoided by showing a positive lower bound on the inter-execution time. Two numerical examples have been given to show the effectiveness of the proposed approach.

Figure 7.5: Comparison of the control inputs under different control schemes

Chapter 8

Conclusions and Future Work

8.1 Conclusion

Buildings account for approximately 40% global energy, half of which is consumed by HVAC systems in buildings [180]. This level of consumption makes the improvement of energy efficiency of HVAC systems important and attractive. In this thesis, we have developed several model-based control strategies to reduce the energy consumption of HVAC systems while maintaining thermal comfort properly. In particular, distributed model predictive control is adopted as the main method to reduce the computational burden and robustness and stability of the proposed algorithms are ensured. The contributions of the thesis can be summarized in the following aspects.

- We have proposed a distributed SMPC strategy for building temperature regulation. We have used a scenario-based method to approximate the stochastic optimization problem, a dual decomposition algorithm to decouple the large scale problem and a sub-gradient method to update the dual variables. Temperature could be regulated in a comfortable region with predefined probability while the energy cost is reduced.

- We have proposed a distributed robust MPC based on reduced order models. Under some mild conditions, modelling error has been considered as bounded disturbances. A neighbor-to-neighbor communication protocol has been used to exchange predicted trajectories among subsystems. A decentralized linear feedback controller has been designed to ensure that real trajectories would evolve around predicted ones. Feasibility and stability have been guaranteed by tube-based MPC strategies.
- We have proposed a novel distributed MPC for dynamically coupled nonlinear continuous-time systems. We have assumed contraction property for local systems and applied it for estimating the discrepancy between predicted and real trajectories. Feasibility of the local optimization problems and stability of the closed-loop system have been proved.
- We have designed a hierarchical distributed MPC scheme for building HVAC systems. A two-layer controller has been proposed to handle the complex structure of a building. A global optimizer has been designed for the upper layer to collect full information of the building and generate references for lower layer controllers. Local optimizers have been designed to track the references and optimize local performance indexes by using local information. Recursive feasibility of lower layer controller has been shown provided feasibility of upper layer controller.
- We have proposed a nonlinear continuous-time MPC framework for networked control systems. The limited bandwidth of the digital communication channel has been considered. The quantization effect has been modelled as a bounded disturbance and a self-triggered control scheme has been proposed to reduce communication and computational burden. Sufficient conditions on feasibility and stability have been provided.
- We have applied our algorithms to several HVAC control systems. Numerical examples have been provided to illustrate the effectiveness of our results.

8.2 Future Work

We list some possible directions for future studies.

- In Chapter 3, we have used a dual-decomposition based algorithm to solve the optimal control problem in a distributed way. Though the convergence of the sub-gradient method is guaranteed, its convergence rate is rather slow. In recent years, an accelerated sub-gradient method [181] has been widely used in distributed MPC. It should be one of our next steps to apply this method to achieve a faster convergence. Furthermore, the simulation results are obtained by using a linear discrete-time model. To validate our results in more realistic model and real HVAC system is also necessary.
- We have proposed a non-cooperative distributed MPC for nonlinear continuous-time system in Chapter 5. Though compared with existing works, the conservativeness is reduced, it is still rather conservative when compared with linear cases [122]. Recently, tube-based MPC has been extended to nonlinear systems [88], [89]. The extension of the approaches to distributed case would be interesting.
- In Chapter 7, we have studied quantization effect of digital communication channels on MPC for networked nonlinear continuous-time systems. However, the analysis is merely based on inherent robustness of quasi-infinite horizon model predictive control and the result is rather conservative. To reduce the conservativeness will be an interesting direction. Feedback MPC used in [86] could be a promising method. A local feedback controller is used to stabilize the system while a feed-forward signal computed by MPC can be used to improve the performance. An extension to distributed self-triggered MPC is also interesting.

Author's Publications

Journal papers:

1. Y. Long, S. Liu, and L. Xie, "Distributed consensus of discrete-time multi-agent systems with multiplicative noises," *International Journal of Robust and Nonlinear Control*, vol. 25, no. 16, pp. 3113–3131, 2015.
2. Y. Long, S. Liu, L. Xie, and K. H. Johansson, "Distributed non-cooperative robust MPC based on reduced order models," *Control Theory and Technology*, vol. 14, no. 1, pp. 11–21, 2016.
3. Y. Long, S. Liu, L. Xie, and K. H. Johansson, "Distributed nonlinear model predictive control based on contraction theory," *submitted*, 2016.

Conference papers:

1. S. Liu, Y. Long, L. Xie, and A. M. Bayen, "Cooperative control of air flow for HVAC systems," in *Proceedings of the 9th IEEE International Conference on Automation Science and Engineering (CASE)*, 2013, pp.422–427.
2. Y. Long, S. Liu, and L. Xie, "Consensus of discrete multi-agent systems with multiplicative noises," in *Proceedings of the 11th International Conference on Control & Automation (ICCA)*, 2014, pp.255–260.

3. Y. Long, S. Liu, L. Xie, and K. H. Johansson, “A scenario-based distributed stochastic MPC for building temperature regulation,” in *Proceedings of the 10th IEEE International Conference on Automation Science and Engineering (CASE)*, 2014, pp.1091–1096.
4. S. Liu, L. Xie and Y. Long, “Mean square average consensus over directed communication topology with measurement noises,” in *Proceedings of the 33rd Chinese Control Conference (CCC)*, 2014, pp.1609–1614.
5. X. Lu, Y. Long, H. Zou, C. Yu, and L. Xie, “Robust extreme learning machine for regression problems with its application to wifi based indoor positioning system,” in *Proceedings of the 24th IEEE International Workshop on Machine Learning for Signal Processing (MLSP)*, 2014.
6. Y. Long, Z. Qiu, S. Liu, and L. Xie, “Constrained consensus with multiplicative noises,” in *Proceedings of the 13th International Conference on Control & Automation (ICCA)*, 2016, pp.1–6.
7. Y. Long, S. Liu, L. Xie, and K. H. Johansson, “A hierarchical distributed MPC for HVAC systems,” *accepted by the American Control Conference (ACC)*, 2016.

Bibliography

- [1] S. Lemmet, “Building and climate change: summary for decision maker,” United Nations Environment Programme, Tech. Rep., 2009.
- [2] A. Aswani, N. Master, J. Taneja, D. Culler, and C. Tomlin, “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.
- [3] Y. Ma, F. Borrelli, B. Hencsey, A. Packard, and S. Bortoff, “Model predictive control of thermal energy storage in building cooling systems,” in *Conference on Decision and Control and Chinese Control Conference, 2009*, 2009, pp. 392–397.
- [4] Y. Ma, F. Borrelli, B. Hencsey, B. Coffey, S. Benghea, and P. Haves, “Model predictive control for the operation of building cooling systems,” *IEEE Transactions on Control Systems Technology*, vol. 20, no. 3, pp. 796–803, 2012.
- [5] G. Grimm, M. J. Messina, S. E. Tuna, and A. R. Teel, “Examples of zero robustness in constrained model predictive control,” in *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, vol. 4, pp. 3724–3729 vol.4.
- [6] —, “Examples when nonlinear model predictive control is nonrobust,” *Automatica*, vol. 40, no. 10, pp. 1729–1738, 2004.

- [7] T. Kim, S. Kato, and S. Murakami, "Indoor cooling/heating load analysis based on coupled simulation of convection, radiation and hvac control," *Building and Environment*, vol. 36, no. 7, pp. 901–908, 2001.
- [8] F. Moukalled, S. Verma, and M. Darwish, "The use of cfd for predicting and optimizing the performance of air conditioning equipment," *International Journal of Heat and Mass Transfer*, vol. 54, no. 1C3, pp. 549–563, 2011.
- [9] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [10] W. Lohmiller and J.-J. E. Slotine, "Nonlinear process control using contraction theory," *AIChE Journal*, vol. 46, no. 3, pp. 588–596, 2000.
- [11] J. McQuade, "A system approach to high performance buildings," United Technologies Corporation, Tech. Rep., 2009.
- [12] J. Laustsen, "Energy efficiency requirements in building codes, energy efficiency policies for new buildings," International Energy Agency(IEA), Tech. Rep., 2008.
- [13] A. Aswani, N. Master, J. Taneja, A. Krioukov, D. Culler, and C. Tomlin, "Energy-efficient building HVAC control using hybrid system LBMPC," *arXiv preprint arXiv:1204.4717*, 2012.
- [14] K. J. Chua, S. K. Chou, W. M. Yang, and J. Yan, "Achieving better energy-efficient air conditioning c A review of technologies and strategies," *Applied Energy*, vol. 104, pp. 87–104, 2013.
- [15] K. Roth, J. Dieckmann, R. Zogg, and J. Brodrick, "Chilled beam cooling," *ASHRAE JOURNAL*, vol. 49, no. 9, p. 84, 2007.
- [16] D. Alexander and M. ORourke, "Design considerations for active chilled beams," *ASHRAE Journal*, vol. 50, no. 9, pp. 50–58, 2008.

- [17] G. Calafiore and M. C. Campi, “Uncertain convex programs: randomized solutions and confidence levels,” *Mathematical Programming*, vol. 102, no. 1, pp. 25–46, 2005.
- [18] G. C. Calafiore and M. C. Campi, “The scenario approach to robust control design,” *IEEE Transactions on Automatic Control*, vol. 51, no. 5, pp. 742–753, 2006.
- [19] G. Calafiore, “Random convex programs,” *SIAM Journal on Optimization*, vol. 20, no. 6, pp. 3427–3464, 2010.
- [20] R. J. Rabehl, J. W. Mitchell, and W. A. Beckman, “Parameter estimation and the use of catalog data in modeling heat exchangers and coils,” *HVAC&R Research*, vol. 5, no. 1, pp. 3–17, 1999.
- [21] Y. Yao, Z. Lian, Z. Hou, and X. Zhou, “Optimal operation of a large cooling system based on an empirical model,” *Applied Thermal Engineering*, vol. 24, no. 16, pp. 2303–2321, 2004.
- [22] S. Wang, “Dynamic simulation of building VAV air-conditioning system and evaluation of EMCS on-line control strategies,” *Building and Environment*, vol. 34, no. 6, pp. 681–705, 1999.
- [23] Y. Wang, W. Cai, Y. Soh, S. Li, L. Lu, and L. Xie, “A simplified modeling of cooling coils for control and optimization of hvac systems,” *Energy Conversion and Management*, vol. 45, no. 18, pp. 2915–2930, 2004.
- [24] G. Maxwell, H. Shapiro, and D. Westra, “Dynamics and control of a chilled water coil,” *ASHRAE transactions*, vol. 95, pp. 1243–1255, 1989.
- [25] D. Underwood, “Modeling and nonlinear control of a hot-water-to-air heat exchanger,” DTIC Document, Tech. Rep., 1990.
- [26] X. Wang, W. Cai, J. Lu, Y. Sun, and X. Ding, “A hybrid dehumidifier model for real-time performance monitoring, control and optimization in liquid des-

- iccant dehumidification system,” *Applied Energy*, vol. 111, no. 0, pp. 449–455, 2013.
- [27] G. Platt, J. Li, R. Li, G. Poulton, G. James, and J. Wall, “Adaptive HVAC zone modeling for sustainable buildings,” *Energy and Buildings*, vol. 42, no. 4, pp. 412–421, 2010.
- [28] S. Goyal, C. Liao, and P. Barooah, “Identification of multi-zone building thermal interaction model from data,” pp. 181–186, 2011.
- [29] S. Goyal and P. Barooah, “A method for model-reduction of non-linear thermal dynamics of multi-zone buildings,” *Energy and Buildings*, vol. 47, no. 0, pp. 332–340, 2012.
- [30] B. Tashtoush, M. Molhim, and M. Al-Rousan, “Dynamic model of an hvac system for control analysis,” *Energy*, vol. 30, no. 10, pp. 1729–1745, 2005.
- [31] B. Li and A. G. Alleyne, “Optimal on-off control of an air conditioning and refrigeration system,” in *Proceedings of the 2010 American Control Conference*, pp. 5892–5897.
- [32] K. J. ?ström, T. Hägglund, and A. Wallenborg, “Automatic tuning of digital controllers with applications to hvac plants,” *Automatica*, vol. 29, no. 5, pp. 1333–1343, 1993.
- [33] W. Qing-Guo, L. Tong-Heng, F. Ho-Wang, B. Qiang, and Z. Yu, “Pid tuning for improved performance,” *IEEE Transactions on Control Systems Technology*, vol. 7, no. 4, pp. 457–465, 1999.
- [34] W. Ya-Gang, S. Zhi-Gang, and C. Wen-Jian, “Pid autotuner and its application in hvac systems,” in *Proceedings of the 2001 American Control Conference. (Cat. No.01CH37148)*, vol. 3, pp. 2192–2196 vol.3.
- [35] Q. Bi, W.-J. Cai, Q.-G. Wang, C.-C. Hang, L. Eng-Lock, Y. Sun, K.-D. Liu, Y. Zhang, and B. Zou, “Advanced controller auto-tuning and its application

- in HVAC systems,” *Control Engineering Practice*, vol. 8, no. 6, pp. 633–644, 2000.
- [36] J. Bai, S. Wang, and X. Zhang, “Development of an adaptive smith predictor-based self-tuning pi controller for an hvac system in a test room,” *Energy and Buildings*, vol. 40, no. 12, pp. 2244–2252, 2008.
- [37] R. Schumann, “Digital parameter-adaptive control of an air-conditioning plant,” *Automatica*, vol. 18, no. 5, pp. 569–575, 1982.
- [38] C. Ozsoy, “Self-tuning control of a heating, ventilating and air-conditioning system,” *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, vol. 207, no. 4, pp. 243–251, 1993.
- [39] J. E. Seem, “A new pattern recognition adaptive controller with application to HVAC systems,” *Automatica*, vol. 34, no. 8, pp. 969–982, 1998.
- [40] G. Singh, M. Zaheer-uddin, and R. V. Patel, “Adaptive control of multivariable thermal processes in HVAC systems,” *Energy Conversion and Management*, vol. 41, no. 15, pp. 1671–1685, 2000.
- [41] T. Y. Chen, “Application of adaptive predictive control to a floor heating system with a large thermal lag,” *Energy and Buildings*, vol. 34, no. 1, pp. 45–51, 2002.
- [42] J. Bai, S. Wang, and X. Zhang, “Development of an adaptive smith predictor-based self-tuning PI controller for an HVAC system in a test room,” *Energy and Buildings*, vol. 40, no. 12, pp. 2244–2252, 2008.
- [43] S. Mukherjee, S. Mishra, and J. T. Wen, “Building temperature control: A passivity-based approach,” in *IEEE 51st Annual Conference on Decision and Control, 2012*, pp. 6902–6907.
- [44] J. T. Wen, S. Mishra, S. Mukherjee, N. Tantisujjatham, and M. Minakais, “Building temperature control with adaptive feedforward,” in *IEEE 52nd Annual Conference on Decision and Control, 2013*, pp. 4827–4832.

- [45] H. Zhang and C. Lilong, "Decentralized nonlinear adaptive control of an H-VAC system," *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, vol. 32, no. 4, pp. 493–498, 2002.
- [46] F. Calvino, M. La Gennusa, G. Rizzo, and G. Scaccianoce, "The control of indoor thermal comfort conditions: introducing a fuzzy adaptive controller," *Energy and Buildings*, vol. 36, no. 2, pp. 97–102, 2004.
- [47] S. Soyguder, M. Karakose, and H. Alli, "Design and simulation of self-tuning PID-type fuzzy adaptive control for an expert HVAC system," *Expert Systems with Applications*, vol. 36, no. 3, Part 1, pp. 4566–4573, 2009.
- [48] S. Soyguder and H. Alli, "Fuzzy adaptive control for the actuators position control and modeling of an expert system," *Expert Systems with Applications*, vol. 37, no. 3, pp. 2072–2080, 2010.
- [49] K. Zhou and J. C. Doyle, *Essentials of robust control*. Prentice hall Upper Saddle River, NJ, 1998, vol. 104.
- [50] M.-c. Chen, *Robust and H_∞ control*. Springer, 2000.
- [51] C. Underwood, "Robust control of HVAC plant I: modelling," *Building services engineering research and technology*, vol. 21, no. 1, pp. 53–61, 2000.
- [52] —, "Robust control of HVAC plant II: controller design," *Building Services Engineering Research and Technology*, vol. 21, no. 1, pp. 63–71, 2000.
- [53] J. C. Doyle, K. Glover, P. P. Khargonekar, and B. A. Francis, "State-space solutions to standard H_2 and H_∞ control problems," *IEEE Transactions on Automatic Control*, vol. 34, no. 8, pp. 831–847, 1989.
- [54] S. Wang and X. Xu, "A robust control strategy for combining dev control with economizer control," *Energy Conversion and Management*, vol. 43, no. 18, pp. 2569–2588, 2002.

- [55] ———, “Optimal and robust control of outdoor ventilation airflow rate for improving energy efficiency and iaq,” *Building and Environment*, vol. 39, no. 7, pp. 763–773, 2004.
- [56] M. Anderson, M. Buehner, P. Young, D. Hittle, C. Anderson, T. Jilin, and D. Hodgson, “Mimo robust control for HVAC systems,” *IEEE Transactions on Control Systems Technology*, vol. 16, no. 3, pp. 475–483, 2008.
- [57] J. H. Lee, M. Morari, and C. E. Garcia, “State-space interpretation of model predictive control,” *Automatica*, vol. 30, no. 4, pp. 707–717, 1994.
- [58] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.
- [59] J. Maciejowski, *Predictive control: with constraints*. Prentice Hall, 2002.
- [60] E. F. Camacho and C. B. Alba, *Model predictive control*, ser. Advanced Textbooks in Control and Signal Processing. Springer, 2004.
- [61] L. Wang, *Model predictive control system design and implementation using MATLAB*. Springer, 2009.
- [62] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill Publishing, LLC, 2013.
- [63] A. I. Propoi, “Use of linear programming methods for synthesizing sample-data automatic systems,” *Automation and Remote Control*, vol. 24, pp. 837–844, 1963.
- [64] D. Kleinman, “An easy way to stabilize a linear constant system,” *IEEE Transactions on Automatic Control*, vol. 15, no. 6, pp. 692–692, 1970.
- [65] W. Kwon and A. Pearson, “On the stabilization of a discrete constant linear system,” *IEEE Transactions on Automatic Control*, vol. 20, no. 6, pp. 800–801, 1975.

- [66] R. Rouhani and R. K. Mehra, "Model algorithmic control (mac); basic theoretical properties," *Automatica*, vol. 18, no. 4, pp. 401–414, 1982.
- [67] Y. Arkun, W. M. Canney, J. Hollett, and M. Morari, "Experimental study of internal model control," *Industrial & Engineering Chemistry Process Design and Development*, vol. 25, no. 1, pp. 102–108, 1986.
- [68] K. L. Levien and M. Morari, "Internal model control of coupled distillation columns," *AIChE Journal*, vol. 33, no. 1, pp. 83–98, 1987.
- [69] J. R. Parrish and C. B. Brosilow, "Inferential control applications," *Automatica*, vol. 21, no. 5, pp. 527–538, 1985.
- [70] J. B. Rawlings and K. R. Muske, "The stability of constrained receding horizon control," *IEEE Transactions on Automatic Control*, vol. 38, no. 10, pp. 1512–1516, 1993.
- [71] D. Q. Mayne and H. Michalska, "Receding horizon control of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 35, no. 7, pp. 814–824, 1990.
- [72] H. Michalska and D. Q. Mayne, "Receding horizon control of nonlinear systems without differentiability of the optimal value function," *Systems & Control Letters*, vol. 16, no. 2, pp. 123–130, 1991.
- [73] P. O. Scokaert, D. Q. Mayne, and J. B. Rawlings, "Suboptimal model predictive control (feasibility implies stability)," *IEEE Transactions on Automatic Control*, vol. 44, no. 3, pp. 648–654, 1999.
- [74] H. Michalska and D. Q. Mayne, "Robust receding horizon control of constrained nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 38, no. 11, pp. 1623–1633, 1993.
- [75] T. H. Yang and E. Polak, "Moving horizon control of nonlinear systems with input saturation, disturbances and plant uncertainty," *International Journal of Control*, vol. 58, no. 4, pp. 875–903, 1993.

- [76] H. Chen and F. Allgöwer, “A quasi-infinite horizon nonlinear model predictive control scheme with guaranteed stability,” *Automatica*, vol. 34, no. 10, pp. 1205–1217, 1998.
- [77] S. L. d. O. Kothare and M. Morari, “Contractive model predictive control for constrained nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 45, no. 6, pp. 1053–1071, 2000.
- [78] L. Magni and R. Scattolini, “Model predictive control of continuous-time nonlinear systems with piecewise constant control,” *IEEE Transactions on Automatic Control*, vol. 49, no. 6, pp. 900–906, 2004.
- [79] S. S. Keerthi and E. G. Gilbert, “Optimal infinite-horizon feedback laws for a general class of constrained discrete-time systems: Stability and moving-horizon approximations,” *Journal of Optimization Theory and Applications*, vol. 57, no. 2, pp. 265–293, 1988.
- [80] L. Magni, G. D. Nicolao, L. Magnani, and R. Scattolini, “A stabilizing model-based predictive control algorithm for nonlinear systems,” *Automatica*, vol. 37, no. 9, pp. 1351–1362, 2001.
- [81] H. Chen, C. W. Scherer, and F. Allgower, “A game theoretic approach to nonlinear robust receding horizon control of constrained systems,” in *American Control Conference, 1997. Proceedings of the 1997*, vol. 5, pp. 3073–3077 vol.5.
- [82] J. H. Lee and Z. Yu, “Worst-case formulations of model predictive control for systems with bounded parameters,” *Automatica*, vol. 33, no. 5, pp. 763–781, 1997.
- [83] P. O. M. Scokaert and D. Q. Mayne, “Min-max feedback model predictive control for constrained linear systems,” *IEEE Transactions on Automatic Control*, vol. 43, no. 8, pp. 1136–1142, 1998.
- [84] F. A. C. C. Fontes and L. Magni, “Special section technical notes and correspondence. min-max model predictive control of nonlinear systems using

- discontinuous feedbacks,” *IEEE Transactions on Automatic Control*, vol. 48, no. 10, pp. 1750–1755, 2003.
- [85] A. Bemporad, F. Borrelli, and M. Morari, “Min-max control of constrained uncertain discrete-time linear systems,” *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1600–1606, 2003.
- [86] L. Chisci, J. A. Rossiter, and G. Zappa, “Systems with persistent disturbances: predictive control with restricted constraints,” *Automatica*, vol. 37, no. 7, pp. 1019–1028, 2001.
- [87] W. Langson, I. Chrysoschoos, S. V. Rakovic, and D. Q. Mayne, “Robust model predictive control using tubes,” *Automatica*, vol. 40, no. 1, pp. 125–133, 2004.
- [88] D. Q. Mayne, E. C. Kerrigan, E. J. van Wyk, and P. Falugi, “Tube-based robust nonlinear model predictive control,” *International Journal of Robust and Nonlinear Control*, vol. 21, no. 11, pp. 1341–1353, 2011.
- [89] S. Yu, C. Maier, H. Chen, and F. Allgöwer, “Tube MPC scheme based on robust control invariant set with application to lipschitz nonlinear systems,” *Systems & Control Letters*, vol. 62, no. 2, pp. 194–200, 2013.
- [90] P. Li, M. Wendt, and G. Wozny, “A probabilistically constrained model predictive controller,” *Automatica*, vol. 38, no. 7, pp. 1171–1176, 2002.
- [91] B. Kouvaritakis, M. Cannon, and P. Couchman, “Mpc as a tool for sustainable development integrated policy assessment,” *IEEE Transactions on Automatic Control*, vol. 51, no. 1, pp. 145–149, 2006.
- [92] P. D. Couchman, M. Cannon, and B. Kouvaritakis, “Stochastic mpc with inequality stability constraints,” *Automatica*, vol. 42, no. 12, pp. 2169–2174, 2006.
- [93] J. A. Primbs and S. Chang Hwan, “Stochastic receding horizon control of constrained linear systems with state and control multiplicative noise,” *IEEE Transactions on Automatic Control*, vol. 54, no. 2, pp. 221–230, 2009.

- [94] M. Cannon, B. Kouvaritakis, and D. Ng, “Probabilistic tubes in linear stochastic model predictive control,” *Systems & Control Letters*, vol. 58, no. 10C11, pp. 747–753, 2009.
- [95] M. Cannon, B. Kouvaritakis, and X. Wu, “Model predictive control for systems with stochastic multiplicative uncertainty and probabilistic constraints,” *Automatica*, vol. 45, no. 1, pp. 167–172, 2009.
- [96] ———, “Probabilistic constrained mpc for multiplicative and additive stochastic uncertainty,” *IEEE Transactions on Automatic Control*, vol. 54, no. 7, pp. 1626–1632, 2009.
- [97] G. C. Calafiore and L. Fagiano, “Robust model predictive control via scenario optimization,” *IEEE Transactions on Automatic Control*, vol. 58, no. 1, pp. 219–224, 2013.
- [98] ———, “Stochastic model predictive control of lpv systems via scenario optimization,” *Automatica*, vol. 49, no. 6, pp. 1861–1866, 2013.
- [99] S. Grammatico, X. Zhang, K. Margellos, P. Goulart, and J. Lygeros, “A scenario approach for non-convex control design,” *IEEE Transactions on Automatic Control*, vol. 61, no. 2, pp. 334–345, 2016.
- [100] T. Tettamanti, I. Varga, B. Kulcsar, and J. Bokor, “Model predictive control in urban traffic network management,” in *2008 16th Mediterranean Conference on Control and Automation*, pp. 1538–1543.
- [101] P. Cortes, M. P. Kazmierkowski, R. M. Kennel, D. E. Quevedo, and J. Rodriguez, “Predictive control in power electronics and drives,” *IEEE Transactions on Industrial Electronics*, vol. 55, no. 12, pp. 4312–4324, 2008.
- [102] R. Hovorka, V. Canonico, L. J. Chassin, U. Haueter, M. Massi-Benedetti, M. O. Federici, T. R. Pieber, H. C. Schaller, L. Schaupp, and T. Vering, “Nonlinear model predictive control of glucose concentration in subjects with type 1 diabetes,” *Physiological measurement*, vol. 25, no. 4, p. 905, 2004.

- [103] Y. Ma, G. Anderson, and F. Borrelli, “A distributed predictive control approach to building temperature regulation,” in *American Control Conference (ACC), 2011*, pp. 2089–2094.
- [104] M. Gomez, J. Rodellar, F. Veà, J. Mantecon, and J. Cardona, “Decentralized predictive control of multi-reach canals,” in *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, vol. 4, pp. 3885–3890 vol.4.
- [105] S. Sawadogo, R. M. Faye, P. O. Malaterre, and F. Mora-Camino, “Decentralized predictive controller for delivery canals,” in *Systems, Man, and Cybernetics, 1998. 1998 IEEE International Conference on*, vol. 4, pp. 3880–3884 vol.4.
- [106] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, “Distributed model predictive control,” *IEEE Control Systems*, vol. 22, no. 1, pp. 44–52, 2002.
- [107] A. Venkat, J. Rawlings, and S. Wright, “Stability and optimality of distributed model predictive control,” in *44th IEEE Conference on Decision and Control, 2005 and 2005 European Control Conference*. IEEE, pp. 6680–6685.
- [108] A. Venkat, I. Hiskens, J. Rawlings, and S. Wright, “Distributed mpc strategies for automatic generation control,” in *Power Plants and Power Systems Control 2006: A Proceedings Volume from the IFAC Symposium on Power Plants and Power Systems Control, Kananaskis, Canada, 2006*. Elsevier Science, p. 383.
- [109] —, “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.
- [110] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia, “Cooperative distributed model predictive control,” *Systems & Control Letters*, vol. 59, no. 8, pp. 460–469, 2010.

- [111] B. T. Stewart, S. J. Wright, and J. B. Rawlings, “Cooperative distributed model predictive control for nonlinear systems,” *Journal of Process Control*, vol. 21, no. 5, pp. 698–704, 2011.
- [112] J. Liu, D. Munoz de la Pena, and P. D. Christofides, “Distributed model predictive control of nonlinear process systems,” *AIChE Journal*, vol. 55, no. 5, pp. 1171–1184, 2009.
- [113] J. Liu, X. Chen, D. Munoz de la Pena, and P. D. Christofides, “Sequential and iterative architectures for distributed model predictive control of nonlinear process systems,” *AIChE Journal*, vol. 56, no. 8, pp. 2137–2149, 2010.
- [114] J. Liu, D. Munoz de la Pena, and P. D. Christofides, “Distributed model predictive control of nonlinear systems subject to asynchronous and delayed measurements,” *Automatica*, vol. 46, no. 1, pp. 52–61, 2010.
- [115] J. Liu, X. Chen, D. M. d. l. Pena, and P. D. Christofides, “Iterative distributed model predictive control of nonlinear systems: Handling asynchronous, delayed measurements,” *IEEE Transactions on Automatic Control*, vol. 57, no. 2, pp. 528–534, 2012.
- [116] P. Giselsson and A. Rantzer, “Distributed model predictive control with suboptimality and stability guarantees,” in *Decision and Control (CDC), 2010 49th IEEE Conference on*, pp. 7272–7277.
- [117] B. Lincoln and A. Rantzer, “Relaxing dynamic programming,” *IEEE Transactions on Automatic Control*, vol. 51, no. 8, pp. 1249–1260, 2006.
- [118] P. Giselsson and A. Rantzer, “On feasibility, stability and performance in distributed model predictive control,” *IEEE Transactions on Automatic Control*, vol. 59, no. 4, pp. 1031–1036, 2014.
- [119] P. Giselsson, “A generalized distributed accelerated gradient method for distributed model predictive control with iteration complexity bounds,” in *American Control Conference (ACC), 2013*, pp. 327–333.

- [120] P. Giselsson, M. D. Doan, T. Keviczky, B. D. Schutter, and A. Rantzer, “Accelerated gradient methods and dual decomposition in distributed model predictive control,” *Automatica*, vol. 49, no. 3, pp. 829–833, 2013.
- [121] W. B. Dunbar, “Distributed receding horizon control of dynamically coupled nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1249–1263, 2007.
- [122] M. Farina and R. Scattolini, “Distributed predictive control: A non-cooperative algorithm with neighbor-to-neighbor communication for linear systems,” *Automatica*, vol. 48, no. 6, pp. 1088–1096, 2012.
- [123] —, “Tube-based robust sampled-data mpc for linear continuous-time systems,” *Automatica*, vol. 48, no. 7, pp. 1473–1476, 2012.
- [124] M. Farina, G. Betti, and R. Scattolini, “Distributed predictive control of continuous-time systems,” *Systems & Control Letters*, vol. 74, no. 0, pp. 32–40, 2014.
- [125] L. Magni and R. Scattolini, “Stabilizing decentralized model predictive control of nonlinear systems,” *Automatica*, vol. 42, no. 7, pp. 1231–1236, 2006.
- [126] X. Liu, Y. Shi, and D. Constantinescu, “Distributed model predictive control of constrained weakly coupled nonlinear systems,” *Systems & Control Letters*, vol. 74, no. 0, pp. 41–49, 2014.
- [127] H. Li and Y. Shi, “Robust distributed model predictive control of constrained continuous-time nonlinear systems: A robustness constraint approach,” *IEEE Transactions on Automatic Control*, vol. 59, no. 6, pp. 1673–1678, 2014.
- [128] W. B. Dunbar and R. M. Murray, “Distributed receding horizon control for multi-vehicle formation stabilization,” *Automatica*, vol. 42, no. 4, pp. 549–558, 2006.

- [129] T. Keviczky, F. Borrelli, and G. J. Balas, “Decentralized receding horizon control for large scale dynamically decoupled systems,” *Automatica*, vol. 42, no. 12, pp. 2105–2115, 2006.
- [130] B. Ding, L. Xie, and W. Cai, “Distributed model predictive control for constrained linear systems,” *International Journal of Robust and Nonlinear Control*, vol. 20, no. 11, pp. 1285–1298, 2010.
- [131] H. Li and Y. Shi, “Distributed model predictive control of constrained nonlinear systems with communication delays,” *Systems & Control Letters*, vol. 62, no. 10, pp. 819–826, 2013.
- [132] C. Wang and C.-J. Ong, “Distributed model predictive control of dynamically decoupled systems with coupled cost,” *Automatica*, vol. 46, no. 12, pp. 2053–2058, 2010.
- [133] A. Gautam, Y. C. Chu, and Y. C. Soh, “Robust H_∞ receding horizon control for a class of coordinated control problems involving dynamically decoupled subsystems,” *IEEE Transactions on Automatic Control*, vol. 59, no. 1, pp. 134–149, 2014.
- [134] A. Gautam, Y.-C. Chu, and Y. C. Soh, “Optimized dynamic policy for receding horizon control of linear time-varying systems with bounded disturbances,” *Automatic Control, IEEE Transactions on*, vol. 57, no. 4, pp. 973–988, 2012.
- [135] W. B. Dunbar and D. S. Caveney, “Distributed receding horizon control of vehicle platoons: Stability and string stability,” *Automatic Control, IEEE Transactions on*, vol. 57, no. 3, pp. 620–633, 2012.
- [136] H. Li, Y. Shi, and W. Yan, “Distributed receding horizon control of constrained nonlinear vehicle formations with guaranteed -gain stability,” *Automatica*, vol. 68, pp. 148–154, 2016.
- [137] A. Richards and J. P. How, “Robust distributed model predictive control,” *International Journal of Control*, vol. 80, no. 9, pp. 1517–1531, 2007.

- [138] P. Trodden and A. Richards, “Cooperative distributed MPC of linear systems with coupled constraints,” *Automatica*, vol. 49, no. 2, pp. 479–487, 2013.
- [139] L. Dai, Y. Xia, Y. Gao, B. Kouvaritakis, and M. Cannon, “Cooperative distributed stochastic mpc for systems with state estimation and coupled probabilistic constraints,” *Automatica*, vol. 61, pp. 89–96, 2015.
- [140] A. Molina, A. Gabaldon, J. A. Fuentes, and F. J. Canovas, “Approach to multivariable predictive control applications in residential HVAC direct load control,” in *IEEE Power Engineering Society Summer Meeting, 2000*, vol. 3, pp. 1811–1816 vol. 3.
- [141] G. P. Henze, D. E. Kalz, S. Liu, and C. Felsmann, “Experimental analysis of model-based predictive optimal control for active and passive building thermal storage inventory,” *HVAC&R Research*, vol. 11, no. 2, pp. 189–213, 2005.
- [142] M. Xu, S. Li, W.-j. Cai, and L. Lu, “Effects of a GPC-PID control strategy with hierarchical structure for a cooling coil unit,” *Energy Conversion and Management*, vol. 47, no. 1, pp. 132–145, 2006.
- [143] S. Yuan and R. Perez, “Multiple-zone ventilation and temperature control of a single-duct vav system using model predictive strategy,” *Energy and Buildings*, vol. 38, no. 10, pp. 1248–1261, 2006.
- [144] M. Xu and S. Li, “Practical generalized predictive control with decentralized identification approach to HVAC systems,” *Energy Conversion and Management*, vol. 48, no. 1, pp. 292–299, 2007.
- [145] M. S. Elliott and B. P. Rasmussen, “Model-based predictive control of a multi-evaporator vapor compression cooling cycle,” in *American Control Conference, 2008*, pp. 1463–1468.
- [146] —, “A model-based predictive supervisory controller for multi-evaporator HVAC systems,” in *American Control Conference, 2009*, pp. 3669–3674.

- [147] F. Blanchin and W. Ukovich, “Linear programming approach to the control of discrete-time periodic systems with uncertain inputs,” *Journal of Optimization Theory and Applications*, vol. 78, no. 3, pp. 523–539, 1993.
- [148] J. Nocedal and S. J. Wright, *Numerical Optimization*. Springer-Verlag, 2006.
- [149] A. Wächter and L. T. Biegler, “On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming,” *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [150] A. Kelman, M. Yudong, and F. Borrelli, “Analysis of local optima in predictive control for energy efficient buildings,” in *2011 50th IEEE Conference on Decision and Control and European Control Conference (CDC-ECC)*, pp. 5125–5130.
- [151] D. M. de la Pena, A. Bemporad, and T. Alamo, “Stochastic programming applied to model predictive control,” in *44th IEEE Conference on Decision and Control and European Control Conference, 2005*, pp. 1361–1366.
- [152] F. Oldewurtel, A. Parisio, C. N. Jones, D. Gyalistras, M. Gwerder, V. Stauch, B. Lehmann, and M. Morari, “Use of model predictive control and weather forecasts for energy efficient building climate control,” *Energy and Buildings*, vol. 45, no. 0, pp. 15–27, 2012.
- [153] Y. Ma and F. Borrelli, “Fast stochastic predictive control for building temperature regulation,” in *American Control Conference, 2012*, pp. 3075–3080.
- [154] Y. Ma, S. Vichik, and F. Borrelli, “Fast stochastic mpc with optimal risk allocation applied to building control systems,” in *Conference on Decision and Control, 2012*, pp. 7559–7564.
- [155] A. Prekopa, *Stochastic Programming*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 1995.

- [156] D. H. van Hessem and O. H. Bosgra, “A conic reformulation of model predictive control including bounded and stochastic disturbances under state and input constraints,” in *Proceedings of the 41st IEEE Conference on Decision and Control, 2002*, vol. 4, pp. 4643–4648 vol.4.
- [157] L. Blackmore, L. Hui, and B. Williams, “A probabilistic approach to optimal robust path planning with obstacles,” in *American Control Conference, 2006*, p. 7 pp.
- [158] A. Parisio, M. Molinari, D. Varagnolo, and K. H. Johansson, “A scenario-based predictive control approach to building HVAC management systems,” in *IEEE International Conference on Automation Science and Engineering, 2013*, 2013, pp. 428–435.
- [159] X. Zhang, G. Schildbach, D. Sturzenegger, and M. Morari, “Scenario-based MPC for energy-efficient building climate control under weather and occupancy uncertainty,” in *European Control Conference, 2013*, pp. 1029–1034.
- [160] X. Zhang, K. Margellos, P. Goulart, and J. Lygeros, “Stochastic model predictive control using a combination of randomized and robust optimization,” in *IEEE 52nd Annual Conference on Decision and Control, 2013*, pp. 7740–7745.
- [161] I. Hazyuk, C. Ghiaus, and D. Penhouet, “Optimal temperature control of intermittently heated buildings using model predictive control: Part I c building modeling,” *Building and Environment*, vol. 51, no. 0, pp. 379–387, 2012.
- [162] M. Gouda, S. Danaher, and C. Underwood, “Building thermal model reduction using nonlinear constrained optimization,” *Building and Environment*, vol. 37, no. 12, pp. 1255–1265, 2002.
- [163] W. Burke and D. Auslander, “Low frequency pulse width modulation design for HVAC compressors,” in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 2009*, pp. 291–297.

- [164] F. Durante and C. Sempì, *Copula theory: an introduction*. Springer, 2010, pp. 3–31.
- [165] R. B. Nelsen, *An introduction to copulas*. Springer, 1999.
- [166] P. K. Trivedi and D. M. Zimmer, *Copula modeling: an introduction for practitioners*. Now Publishers Inc, 2007.
- [167] T. Larsson and Z. Liu, “A Lagrangean relaxation scheme for structured linear programs with application to multicommodity network flows,” *Optimization*, vol. 40, no. 3, pp. 247–284, 1997.
- [168] W. J. Baumol and T. Fabian, “Decomposition, pricing for decentralization and external economies,” *Management Science*, vol. 11, no. 1, pp. 1–32, 1964.
- [169] R. Ghaemi, J. Sun, and I. V. Kolmanovsky, “Robust control of constrained linear systems with bounded disturbances,” *IEEE Transactions on Automatic Control*, vol. 57, no. 10, pp. 2683–2688, 2012.
- [170] G. Betti, M. Farina, and R. Scattolini, “Realization issues, tuning, and testing of a distributed predictive control algorithm,” *Journal of Process Control*, vol. 24, no. 4, pp. 424–434, 2014.
- [171] M. Farina, P. Colaneri, and R. Scattolini, “Block-wise discretization accounting for structural constraints,” *Automatica*, vol. 49, no. 11, pp. 3411–3417, 2013.
- [172] D. Q. Mayne, M. M. Seron, and S. V. Rakovic, “Robust model predictive control of constrained linear systems with bounded disturbances,” *Automatica*, vol. 41, no. 2, pp. 219–224, 2005.
- [173] G. Russo, M. di Bernardo, and E. D. Sontag, “A contraction approach to the hierarchical analysis and design of networked systems,” *IEEE Transactions on Automatic Control*, vol. 58, no. 5, pp. 1328–1331, 2013.

- [174] S. Bonnabel and J. J. Slotine, “A contraction theory-based analysis of the stability of the deterministic extended kalman filter,” *IEEE Transactions on Automatic Control*, vol. 60, no. 2, pp. 565–569, 2015.
- [175] M. Rubagotti, D. M. Raimondo, A. Ferrara, and L. Magni, “Robust model predictive control with integral sliding mode in continuous-time sampled-data nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 56, no. 3, pp. 556–570, 2011.
- [176] G. Russo, M. di Bernardo, and E. D. Sontag, “Global entrainment of transcriptional systems to periodic inputs,” *PLoS Comput Biol*, vol. 6, no. 4, p. e1000739, 2010.
- [177] E. C. K. P. Falugi and E. V. Wyk, *Imperial College London Optimal Control Software User Guide (ICLOCS)*. Department of Electrical Engineering, Imperial College London, London, UK., 2014.
- [178] P. Morosan, 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.
- [179] H. Li and Y. Shi, “Event-triggered robust model predictive control of continuous-time nonlinear systems,” *Automatica*, vol. 50, no. 5, pp. 1507–1513, 2014.
- [180] L. Prez-Lombard, J. Ortiz, and C. Pout, “A review on buildings energy consumption information,” *Energy and Buildings*, vol. 40, no. 3, pp. 394–398, 2008.
- [181] P. Tseng, “On accelerated proximal gradient methods for convex-concave optimization,” *SIAM Journal on Optimization*, submitted.