Title
Distributed Model Predictive Control of Nonlinear and Two-Time-Scale Process Networks

Author
Chen, Xianzhong

Publication Date
2012

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Distributed Model Predictive Control of Nonlinear and Two-Time-Scale Process Networks

A dissertation submitted in partial satisfaction of the requirements for the degree Doctor of Philosophy in Chemical Engineering

by

Xianzhong Chen

2012

ABSTRACT OF THE DISSERTATION

Distributed Model Predictive Control of Nonlinear
and Two-Time-Scale Process Networks

by

Xianzhong Chen

Doctor of Philosophy in Chemical Engineering
University of California, Los Angeles, 2012
Professor Panagiotis D. Christofides, Chair

Large-scale chemical process systems are characterized by highly nonlinear behavior and the coupling of physico-chemical phenomena occurring at disparate time scales. Examples include fluidized catalytic crackers, distillation columns, biochemical reactors as well as chemical process networks in which the individual processes evolve in a fast time-scale and the network dynamics evolve in a slow time-scale. Traditionally, the design of advanced model-based control systems for chemical processes has followed the centralized paradigm in which one control system is used to compute the control actions of all manipulated inputs. While the centralized paradigm to model-based process control has been successful, when the number of the process state variables, manipulated inputs and measurements in a chemical plant becomes large - a common occurrence in modern plants -, the computational time needed for the solution of the centralized control problem may increase significantly and may impede the ability of centralized control systems (particularly when nonlinear con-

strained optimization-based control systems like model predictive control-MPC are used), to carry out real-time calculations within the limits set by process dynamics and operating conditions. One feasible alternative to overcome this problem is to utilize cooperative, distributed control architectures in which the manipulated inputs are computed by solving more than one control (optimization) problems in separate processors in a coordinated fashion.

Motivated by the above considerations, this dissertation presents rigorous, yet practical, methods for the design of distributed model predictive control systems for nonlinear and two-time-scale process networks. Beginning with a review of results on the subject, the first part of this dissertation presents the design of two, sequential and iterative, distributed MPC architectures via Lyapunov-based control techniques for general nonlinear process systems. Key practical issues like the feedback of asynchronous and delayed measurements as well as the utilization of cost functions that explicitly account for economic considerations are explicitly addressed in the formulation and design of the controllers and of their communication strategy. In the second part of the dissertation, we focus on the design of model predictive control systems for nonlinear two-times-scale process networks within the framework of singular perturbations. Both centralized and distributed MPC designs are presented. Throughout the thesis, the applicability, effectiveness and computational efficiency of the control methods are evaluated via simulations using numerous, large-scale chemical process networks.

The dissertation of Xianzhong Chen is approved.

James F. Davis

Gerassimos Orkoulas

Tsu-Chin Tsao

Panagiotis D. Christofides, Committee Chair

University of California, Los Angeles

2012

Contents

List of Figures

List of Tables

ACKNOWLEDGEMENTS

I would like to thank my advisors, Professor Panagiotis D. Christofides and Professor James F. Davis, for their guidance and support throughout the course of the thesis.

I would like to thank Professor Gerassimos Orkoulas and Professor Tsu-Chin Tsao for agreeing to participate in my doctoral committee.

Financial support from the National Science Foundation and from the UCLA Graduate Division is gratefully acknowledged.

I would like to express my deepest gratitude to my parents, Shanhui Chen and Jiamei Li, for their encouragement, patience and support.

VITA

2008 Bachelor of Science, Chemical Engineering
Department of Chemical and Biomolecular Engineering
University of California, Los Angeles

2010 Master of Science, Chemical Engineering
Department of Chemical and Biomolecular Engineering
University of California, Los Angeles

2009–2012 Teaching Assistant/Associate/Fellow
Department of Chemical and Biomolecular Engineering
University of California, Los Angeles

PUBLICATIONS AND PRESENTATIONS

1. X. Chen, M. Heidarinejad, J. Liu, and P. D. Christofides. Composite Fast-Slow MPC Design for Nonlinear Singularly Perturbed Systems. *AIChE Journal*, 58:1801–1810, 2012.

2. X. Chen, M. Heidarinejad, J. Liu, and P. D. Christofides. Distributed Economic MPC: Application to a Nonlinear Chemical Process Network. *Journal of Process Control*, 22:689–699, 2012.

3. X. Chen, M. Heidarinejad, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Model Predictive Control of Nonlinear Singularly Perturbed Systems: Application to a Large-Scale Process Network. *Journal of Process Control*, 21:1296–1305, 2011.

4. J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Sequential and Iterative Architectures for Distributed Model Predictive Control of Nonlinear Process Systems. *AIChE Journal*, 56:2137–2149, 2010.

5. J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Iterative Distributed Model Predictive Control of Nonlinear Systems: Handling Asynchronous, Delayed Measurements. *IEEE Transactions on Automatic Control*, 57:528–534, 2012.

6. W. Qi, J. Liu, X. Chen, and P. D. Christofides. Supervisory Predictive Control of Stand-alone Wind-Solar Energy Generation Systems. *IEEE Transactions on Control Systems Technology*, 19:199–207, 2011.

7. X. Chen, M. Heidarinejad, J. liu, and P. D. Christofides. Composite Fast-slow MPC Design for Nonlinear Singularly Perturbed Systems. *Proceeding of the Amercian Control Conference*, in press, 2012.

8. X. Chen, M. Heidarinejad, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Model Predictive Control of Nonlinear Singularly Perturbed Systems: Application to a Reactor-Separator Process Network. *Proceedings of the 50th IEEE Conference on Decision and Control and European Control Conference*, pages 8125-8132, Orlando, Florida, 2011.

9. X. Chen, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Sequential and Iterative Distributed Model Predictive Control of Nonlinear Process Systems Subject to Asynchronous Measurements. *Proceedings of the 9th International Symposium on Dynamics and Control of Process Systems*, pages 611–616, Leuven, Belgium, 2010.

10. X. Chen, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Distributed Model Predictive Control of Two-Time-Scale Nonlinear Systems. *AIChE Annual Meeting*, paper 393e, Salt Lake City, Utah, 2010.

11. J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Architecture for Distributed Model Predictive Control of Nonlinear Process Systems. *AIChE Annual Meeting*, paper 504c, Nashville, Tennessee, 2009.

12. J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Distributed Model Predictive Control of Nonlinear Systems: Handling Delayed Measurements. *AIChE Annual Meeting*, paper 586e, Salt Lake City, Utah, 2010.

13. J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Sequential and Iterative Architectures for Distributed Model Predictive Control of Nonlinear Process Systems. Part I: Theory. *Proceedings of the American Control Conference*, 3148–3155, Baltimore, Maryland, 2010 (**Best Presentation in Session Award**).

14. J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Sequential and Iterative Architectures for Distributed Model Predictive Control of Nonlinear Process Systems. Part II: Application to a Catalytic Alkylation of Benzene Process. *Proceedings of the American Control Conference*, 3156–3161, Baltimore, Maryland, 2010.

15. W. Qi, X. Chen, J. Liu, and P. D. Christofides. Supervisory Predictive Control of Integrated Wind/Solar Energy Generation Systems. *AIChE Annual Meeting*, paper 454c, Nashville, Tennessee, 2009.

Chapter 1

Introduction

1.1 Background

Chemical processes and plants are characterized by nonlinear behavior and strong coupling of physico-chemical phenomena occurring at disparate time-scales. Examples include fluidized catalytic crackers, distillation columns, biochemical reactors as well as chemical process networks in which the individual processes evolve in a fast time-scale and the network dynamics evolve in a slow time-scale.

The nonlinear behavior of many chemical processes has motivated extensive research on nonlinear process control with the emphasis being primarily on the design of centralized nonlinear control systems (see, for example, [56, 6, 107, 41, 76, 15] for results and references in this area). In addition to nonlinear behavior, industrial process models are characterized by the presence of unknown process parameters and external disturbances which, if not accounted for in the controller design, may cause performance deterioration and even closed-loop instability. This realization has motivated numerous studies on the problem of designing controllers for uncertain nonlinear sys-

tems (e.g., [23, 51, 73, 49, 18]). The problems caused by input constraints have also motivated numerous studies on the dynamics and control of systems subject to input constraints. Important contributions include results on optimization-based methods such as model predictive control (e.g., [36, 88, 98, 76, 1]) and Lyapunov-based control (e.g., [59, 96, 48, 97, 94, 53]). Earlier work of our group developed [28, 29, 18] a unified framework for model-based control of continuous-time nonlinear processes with uncertainty *and* input constraints. This framework adopts a Lyapunov-based approach and leads to the synthesis of robust nonlinear controllers that possess an explicitly characterized region of guaranteed closed-loop stability. Subsequently, we used these results to develop a hybrid predictive control structure that employs switching between Lyapunov-based control and model predictive control (MPC) for stabilization of nonlinear systems with uncertainty and input constraints [31, 78]. The hybrid predictive control structure embeds the implementation of MPC within the stability region of a Lyapunov-based bounded controller which serves as a fall-back controller that can be switched to in the event of infeasibility or instability of MPC. We also note here important other work on a diverse array of hybrid system issues including modeling (e.g., [106, 4]), simulation (e.g., [4, 37]), optimization (e.g., [33, 39, 38, 8, 7]), stability analysis (e.g., [109, 42, 25]), and control (e.g., [5, 43, 55, 32, 30, 81]); all of these results, however, deal with the design of centralized control systems.

While the centralized paradigm to model-based process control has been successful, when the number of the process state variables, manipulated inputs and measurements in a chemical plant becomes large - a common occurrence in modern plants -, the computational time needed for the solution of the centralized control problem may increase significantly and may impede the ability of centralized control systems (particularly when nonlinear constrained optimization-based control systems like model predictive control-MPC are used), to carry out real-time calculations within the limits

set by process dynamics and operating conditions. One feasible alternative to overcome this problem is to utilize cooperative, distributed control architectures in which the manipulated inputs are computed by solving more than one control (optimization) problems in separate processors in a coordinated fashion. With respect to available results in this direction, several distributed MPC methods have been proposed in the literature that deal with the coordination of separate MPC controllers that communicate in order to obtain optimal input trajectories in a distributed manner; see [10, 89, 92] for reviews of results in this area. More specifically, in [27], the problem of distributed control of dynamically coupled nonlinear systems that are subject to decoupled constraints was considered. In [91, 46], the effect of the coupling was modeled as a bounded disturbance compensated using a robust MPC formulation. In [100], it was proven that through multiple communications between distributed controllers and using system-wide control objective functions, stability of the closed-loop system can be guaranteed. In [50], distributed MPC of decoupled systems (a class of systems of relevance in the context of multi-agents systems) was studied. In [72], an MPC algorithm was proposed under the main condition that the system is nonlinear, discrete-time and no information is exchanged between local controllers, and in [85], MPC for nonlinear systems was studied from an input-to-state stability point of view. In [71], a game theory based distributed MPC scheme for constrained linear systems was proposed.

Time-scale multiplicity is a common feature of many chemical processes and plants of industrial interest and usually arises due to the strong coupling of physico-chemical phenomena, like slow and fast reactions, occurring at disparate time-scales. In addition to this, the dynamics of control actuation and measurement sensing systems very often adds a fast-dynamics layer in the closed-loop system. The analysis and controller design problems for multiple-time-scale systems are usually addressed within

the mathematical framework of singular perturbations (e.g., [54]). Within this framework, a variety of explicit controller design methods have been primarily developed for both linear and nonlinear singularly perturbed systems, ranging from optimal control [54] to geometric control (e.g., [16, 57]) and Lyapunov-based control [21].

In the context of MPC of singularly perturbed systems, most of the efforts have been dedicated to linear systems [104] or to MPC of specific classes of two-time-scale processes [9, 99]. In a recent work [12], we studied MPC for nonlinear singularly perturbed systems where MPC is used only in the slow time-scale and the fast dynamics are assumed to be stabilizable by a "fast" explicit controller. Finally, in another recent set of papers [3, 70], MPC of two-time-scale processes described by nonlinear singularly perturbed systems in non-standard form (i.e., systems in which the separation of slow and fast state variables is not explicit in the original coordinates and a coordinate change should be used to obtain a singularly perturbed system in standard form) was addressed; in these works the fast dynamics are also assumed to be stabilizable by a "fast" explicit controller.

1.2 Objectives and Organization of the Dissertation

Motivated by the potential benefit of distributed control systems to regulate complex process networks, the objectives of this dissertation are summarized below:

1. To develop distributed predictive control methods for large-scale nonlinear process networks taking into account asynchronous and time-varying delayed measurements.

2. To develop distributed economic predictive control methods for large-scale non-

linear process networks.

3. To develop model predictive control methods for two-time-scale process networks.

The dissertation is organized as follows. In Chapter 2, we focus on distributed model predictive control of large scale nonlinear process systems in which several distinct sets of manipulated inputs are used to regulate the process. For each set of manipulated inputs, a different model predictive controller is used to compute the control actions, which is able to communicate with the rest of the controllers in making its decisions. Under the assumption that feedback of the state of the process is available to all the distributed controllers at each sampling time and a model of the plant is available, we propose two different distributed model predictive control architectures. In the first architecture, the distributed controllers use a one-directional communication strategy, are evaluated in sequence and each controller is evaluated only once at each sampling time; in the second architecture, the distributed controllers utilize a bi-directional communication strategy, are evaluated in parallel and iterate to improve closed-loop performance. In the design of the distributed model predictive controllers, Lyapunov-based model predictive control techniques are used. In order to ensure the stability of the closed-loop system, each model predictive controller in both architectures incorporates a stability constraint which is based on a suitable Lyapunov-based controller. We prove that the proposed distributed model predictive control architectures enforce practical stability in the closed-loop system and optimal performance. The theoretical results are illustrated through a catalytic alkylation of benzene process example.

In Chapter 3, we propose sequential and iterative DMPC schemes for large scale nonlinear systems subject to asynchronous and delayed state feedback [64, 14, 62]. In

the case of asynchronous feedback, under the assumption that there is an upper bound on the maximum interval between two consecutive measurements, we first extend the results obtained in Chapter 2 for sequential DMPC to include asynchronous feedback, and then re-design the iterative DMPC scheme presented in Chapter 2 to take explicitly into account asynchronous feedback. Subsequently, we design an iterative DMPC scheme for systems subject to asynchronous feedback that also involves time-delays under the assumption that there exists an upper bound on the maximum feedback delay. This design takes advantage of the bi-directional communication network used in the iterative DMPC framework. Sufficient conditions under which the proposed distributed control designs guarantee that the states of the closed-loop system are ultimately bounded in regions that contain the origin are provided. The theoretical results are illustrated through a catalytic alkylation of benzene process example.

In Chapter 4, we focus on the development and application of distributed and centralized Lyapunov economic MPC designs to a catalytic alkylation of benzene process network, which consists of four continuously stirred tank reactors and a flash separator. A new economic measure for the entire process network is proposed which accounts for a broad set of economic considerations on the process operation including reaction conversion, separation quality and energy efficiency. Subsequently, steady-state process optimization is first carried out to locate an economically optimal (with respect to the proposed economic measure) operating steady-state. Then, a sequential distributed economic model predictive control design method, suitable for large-scale process networks, is proposed and its closed-loop stability properties are established. Using the proposed method, economic, distributed as well as centralized, model predictive control systems are designed and are implemented on the process to drive the closed-loop system state close to the economically optimal steady-state. The closed-loop performance and time needed for control action calculation are evaluated

through simulations and compared with the ones of a centralized Lyapunov-based model predictive control design, which uses a conventional, quadratic cost function that includes penalty on the deviation of the states and inputs from their economically optimal steady-state values.

In Chapter 5, we focus on model predictive control of nonlinear singularly perturbed systems in standard form where the separation between the fast and slow state variables is explicit. A composite control system using multirate sampling (i.e., fast sampling of the fast state variables and slow sampling of the slow state variables) and consisting of a "fast" feedback controller that stabilizes the fast dynamics and a model predictive controller that stabilizes the slow dynamics and enforces desired performance objectives in the slow subsystem is designed. Using stability results for nonlinear singularly perturbed systems, the closed-loop system is analyzed and sufficient conditions for stability are derived. A large-scale nonlinear reactor-separator process network is used to demonstrate the application of the method including a distributed implementation of the predictive controller.

In Chapter 6, a composite control system comprised of a "fast" MPC acting to regulate the fast dynamics and a "slow" MPC acting to regulate the slow dynamics is designed. The composite MPC system uses multirate sampling of the plant state measurements, i.e., fast sampling of the fast state variables is used in the fast MPC and slow-sampling of the slow state variables is used in the slow MPC as well as in the fast MPC. Using singular perturbation theory, the stability and optimality of the closed-loop nonlinear singularly perturbed system are analyzed. The proposed fast-slow MPC design does not require communication between the two MPCs, and thus, it can be classified as decentralized in nature. A chemical process example which exhibits two-time-scale behavior is used to demonstrate the structure and implemen-

tation of the fast-slow MPC architecture in a practical setting. Extensive simulations are carried out to assess the performance and computational efficiency of the fast-slow MPC system.

Chapter 7 summaries the main results of the dissertation and discusses future research possibilities in distributed control system design.

Chapter 2

Sequential and Iterative Architectures for Distributed Model Predictive Control of Nonlinear Process Systems

2.1 Introduction

Model predictive control (MPC) is a popular control strategy based on using a model of the process to predict at each sampling time the future evolution of the system from the current state along a given prediction horizon. Using these predictions, the manipulated input trajectory that minimizes a given performance index is computed solving a suitable optimization problem. To obtain finite dimensional optimization problems, MPC optimizes over a family of piecewise constant trajectories with a fixed sampling time and a finite prediction horizon. Once the optimization problem

is solved, only the first manipulated input value is implemented, discarding the rest of the trajectory and repeating the optimization in the next sampling step [36, 86]. Typically, MPC is studied from a centralized control point of view in which all the manipulated inputs of a control system are optimized with respect to an objective function in a single optimization problem. When the number of the state variable and manipulated inputs of the process, however, becomes large, the computational burden of the centralized optimization problem may increase significantly and may impede the applicability of a centralized MPC, especially in the case where nonlinear process models are used in the MPC. One feasible alternative to overcome this problem is to utilize a distributed MPC architecture in which the manipulated inputs are computed by more than one optimization problems in a coordinated fashion. With respect to available results on distributed MPC architectures, several distributed MPC methods have been proposed in the literature that deal with the coordination of separate MPC controllers that communicate in order to obtain optimal input trajectories in a distributed manner; see [10, 89, 92] for reviews of results in this area. In [65], a distributed MPC architecture with one-directional communication was proposed for general nonlinear process systems. In this architecture, two separate MPC controllers designed via Lyapunov-based MPC (LMPC) were considered, in which one LMPC was used to guarantee the stability of the closed-loop system and the other LMPC was used to improve the closed-loop performance. Generally, the computational burden of these distributed MPC methods is smaller compared to the one of the corresponding centralized MPC because of the formulation of optimization problems with a smaller number of decision variables.

In this chapter, we focus on distributed model predictive control of large scale nonlinear process systems in which several distinct sets of manipulated inputs are used to regulate the process [63]. For each set of manipulated inputs, a different

model predictive controller is used to compute the control actions, which is able to communicate with the rest of the controllers in making its decisions. Under the assumption that feedback of the state of the process is available to all the distributed controllers at each sampling time and a model of the plant is available, we propose two different distributed model predictive control architectures designed via LMPC techniques. In the first architecture, the distributed controllers use a one-directional communication strategy, are evaluated in sequence and each controller is evaluated only once at each sampling time; in the second architecture, the distributed controllers utilize a bi-directional communication strategy, are evaluated in parallel and iterate to improve closed-loop performance. In order to ensure the stability of the closed-loop system, each model predictive controller in both architectures incorporates a stability constraint which is based on a suitable Lyapunov-based controller. We prove that the proposed distributed model predictive control architectures enforce practical stability in the closed-loop system and optimal performance. The theoretical results are illustrated through a catalytic alkylation of benzene process example.

2.2 Preliminaries

2.2.1 Control problem formulation

We consider nonlinear process systems described by the following state-space model:

$$\dot{x}(t) = f(x(t)) + \sum_{i=1}^{m} g_i(x(t))u_i(t) + k(x(t))w(t) \tag{2.1}$$

where $x(t) \in R^{n_x}$ denotes the vector of process state variables, $u_i(t) \in R^{m_{u_i}}$, $i = 1, \ldots, m$, are m sets of control (manipulated) inputs and $w(t) \in R^{n_w}$ denotes the vec-

tor of disturbance variables. The m sets of inputs are restricted to be in m nonempty convex sets $U_i \subseteq R^{m_{u_i}}$, $i = 1, \ldots, m$, which are defined as follows:

$$U_i := \{u_i \in R^{m_{u_i}} : |u_i| \leq u_i^{\max}\}^*, i = 1, \ldots, m$$

where $u_i^{\max}$, $i = 1, \ldots, m$, are the magnitudes of the input constraints. The disturbance vector is bounded, i.e., $w(t) \in W$ where

$$W := \{w \in R^{n_w} : |w| \leq \theta, \theta > 0\}.$$

We assume that f, g_i, $i = 1, \ldots, m$, and k are locally Lipschitz vector functions and that the origin is an equilibrium of the unforced nominal system (i.e., system of Eq. 2.1 with $u_i(t) = 0$, $i = 1, \ldots, m$, $w(t) = 0$ for all t) which implies that $f(0) = 0$. We also assume that the state x of the system is sampled synchronously and the time instants at which we have state measurement samplings are indicated by the time sequence $\{t_{k\geq 0}\}$ with $t_k = t_0 + k\Delta,\ k = 0, 1, \ldots$ where t_0 is the initial time and Δ is the sampling time.

Remark 2.1 *In general, distributed control systems are formulated based on the assumption that the controlled process systems consist of decoupled or partially coupled subsystems. However, we consider a fully coupled process model; this is a very common occurrence in chemical process control as we will illustrate in the example of section 2.4. In our future work, we will extend the proposed distributed control systems to the case in which only local state information is available to each distributed controller based on distributed state estimation.*

Remark 2.2 *Note that the assumption that f, g_i, $i = 1, \ldots, m$ and k are locally*

*$|\cdot|$ denotes Euclidean norm of a vector.

Lipschitz vector functions is a reasonable assumption for most of chemical processes. Note also that the assumption that the state x of the system is sampled synchronously is a widely used assumption in process control research. The proposed control system designs can be extended to the case where only part of the state x is measurable by designing an observer to estimate the whole state vector from output measurements and by designing the control system based on the measured and estimated states. In this case, the stability properties of the resulting output feedback control systems are affected by the convergence of the observer and need to be carefully studied.

2.2.2 Lyapunov-based controller

We assume that there exists a Lyapunov-based controller $h(x) = [h_1(x) \ \dots \ h_m(x)]^T$ with $u_i = h_i(x),\ i = 1, \dots, m$, which renders the origin of the nominal closed-loop system asymptotically stable while satisfying the input constraints for all the states x inside a given stability region. We note that this assumption is essentially equivalent to the assumption that the process is stabilizable or that the pair (A, B) in the case of linear systems is stabilizable. Using converse Lyapunov theorems [75, 60, 18], this assumption implies that there exist functions $\alpha_i(\cdot),\ i = 1, 2, 3, 4$ of class $\mathcal{K}^\dagger$ and a continuously differentiable Lyapunov function $V(x)$ for the nominal closed-loop

†A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$.

system, that satisfy the following inequalities:

$$\begin{gathered}\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \\ \frac{\partial V(x)}{\partial x}(f(x) + \sum_{i=1}^{m} g_i(x)h_i(x)) \leq -\alpha_3(|x|) \\ |\frac{\partial V(x)}{\partial x}| \leq \alpha_4(|x|) \\ h_i(x) \in U_i,\ i = 1, \ldots, m\end{gathered} \tag{2.2}$$

for all $x \in D \subseteq R^{n_x}$ where D is an open neighborhood of the origin. We denote the region ${\Omega_\rho}^{\ddagger} \subseteq D$ as the stability region of the closed-loop system under the Lyapunov-based controller $h(x)$. The construction of $V(x)$ can be carried out in a number of ways using systematic techniques like, for example, sum-of-squares methods.

By continuity, the local Lipschitz property assumed for the vector fields f, g_i, $i = 1, \ldots, m$, and k and taking into account that the manipulated inputs u_i, $i = 1, \ldots, m$, and the disturbance w are bounded in convex sets, there exists a positive constant M such that

$$|f(x) + \sum_{i=1}^{m} g_i(x)u_i + k(x)w| \leq M \tag{2.3}$$

for all $x \in \Omega_\rho$, $u_i \in U_i$, $i = 1, \ldots, m$, and $w \in W$. In addition, by the continuous differentiable property of the Lyapunov function $V(x)$ and the Lipschitz property assumed for the vector field f, there exist positive constants L_x, L_{u_i}, $i = 1, \ldots, m$, and L_w such that

$$\begin{gathered}|\frac{\partial V}{\partial x}f(x) - \frac{\partial V}{\partial x}f(x')| \leq L_x|x - x'| \\ |\frac{\partial V}{\partial x}g_i(x) - \frac{\partial V}{\partial x}g_i(x')| \leq L_{u_i}|x - x'|,\ i = 1, \ldots, m \\ |\frac{\partial V}{\partial x}k(x)| \leq L_w\end{gathered} \tag{2.4}$$

[‡]We use Ω_ρ to denote the set $\Omega_\rho := \{x \in R^{n_x} | V(x) \leq \rho\}$.

for all $x, x' \in \Omega_\rho$, $u_i \in U_i$, $i = 1, \ldots, m$, and $w \in W$.

Remark 2.3 *Different state feedback control laws for nonlinear systems have been developed using Lyapunov techniques; the reader may refer to [59, 53, 18, 28, 29] for results in this area including results on the design of bounded Lyapunov-based controllers by taking explicitly into account input constraints for broad classes of nonlinear systems.*

2.2.3 Centralized LMPC

To take advantage of all the sets of manipulated inputs, one option is to design a centralized MPC controller. In order to guarantee robust stability of the closed-loop system, the MPC controller must include a set of stability constraints. To do this, we propose to use the LMPC controller proposed in [77, 79] which guarantees practical stability of the closed-loop system, allows for an explicit characterization of the stability region and yields a reduced complexity optimization problem. LMPC is based on uniting receding horizon control with control Lyapunov functions and computes the manipulated input trajectory solving a finite horizon constrained optimal control problem. The LMPC controller is based on the Lyapunov-based controller $h(x)$. The controller $h(x)$ is used to define a stability constraint for the LMPC controller which guarantees that the LMPC controller inherits the stability and robustness properties of the Lyapunov-based controller $h(x)$. The LMPC controller introduced in [77, 79] is based on the following optimization problem:

$$\min_{u_{c1} \ldots u_{cm} \in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T(\tau) Q_c \tilde{x}(\tau) + \sum_{i=1}^{m} u_{ci}^T(\tau) R_{ci} u_{ci}(\tau)] d\tau \tag{2.5a}$$

$$\text{s.t. } \dot{\tilde{x}}(\tau) = f(\tilde{x}(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}(\tau)) u_{ci} \tag{2.5b}$$

$$u_{ci}(\tau) \in U_i,\ i = 1, \ldots, m \tag{2.5c}$$

$$\tilde{x}(0) = x(t_k) \tag{2.5d}$$

$$\sum_{i=1}^{m} \frac{\partial V(x)}{\partial x} g_i(x(t_k)) u_{ci}(0) \leq \sum_{i=1}^{m} \frac{\partial V(x)}{\partial x} g_i(x(t_k)) h_i(x(t_k)) \tag{2.5e}$$

where $S(\Delta)$ is the family of piece-wise constant functions with sampling period Δ, N is the prediction horizon, Q_c and R_{ci}, $i = 1, \ldots, m$, are positive definite weight matrices that define the cost, $x(t_k)$ is the state measurement obtained at t_k, $\tilde{x}$ is the predicted trajectory of the nominal system with u_i, $i = 1, \ldots, m$, the input trajectory computed by the LMPC of Eq. 2.5.

The optimal solution to this optimization problem is denoted by $u_{ci}^*(\tau|t_k)$, $i = 1, \ldots, m$, which is defined for $\tau \in [0, N\Delta)$. The LMPC controller is implemented with a receding horizon method; that is, at each sampling time t_k, the new state $x(t_k)$ is received from the sensors, the optimization problem of Eq. 2.5 is solved, and $u_{ci}^*(t - t_k|t_k)$, $i = 1, \ldots, m$ are applied to the closed-loop system for $t \in [t_k, t_{k+1})$.

The optimization problem of Eq. 2.5 does not depend on the uncertainty and guarantees that the system in closed-loop with the LMPC controller of Eq. 2.5 maintains the stability properties of the Lyapunov-based controller. The constraint of Eq. 2.5e guarantees that the value of the time derivative of the Lyapunov function at the initial evaluation time of the centralized LMPC controller is lower or equal to the value obtained if only the Lyapunov-based controller $h(x)$ is implemented in the closed-loop system in a sample-and-hold fashion. This is the constraint that allows proving that the centralized LMPC controller inherits the stability and robustness properties of the Lyapunov-based controller.

The manipulated inputs of the closed-loop system under the above centralized

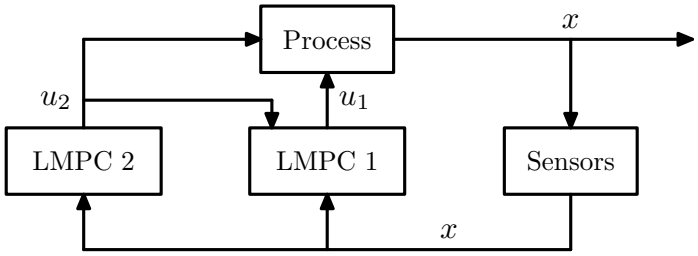


Figure 2.1: Distributed MPC scheme proposed in [65].

LMPC controller are defined as follows

$$u_i(t) = u_{ci}^*(t - t_k|t_k),\ i = 1, \ldots, m,\ \forall t \in [t_k, t_{k+1}). \tag{2.6}$$

In what follows, we refer to this controller as the centralized LMPC. The main property of the centralized LMPC controller is that the origin of the closed-loop system is practically stable for all initial states inside the stability region Ω_ρ for a sufficient small sampling time Δ and disturbance upper bound θ. This property is also guaranteed by the Lyapunov-based controller $h(x)$ when the controllers are implemented in a sample-and-hold fashion (see [22, 82] for results on sampled-data systems). The main advantage of LMPC approaches with respect to the Lyapunov-based controller is that optimality considerations can be taken explicitly into account (as well as constraints on the inputs and the states [79]) in the computation of the controller within an online optimization framework improving closed-loop performance.

2.3 Distributed model predictive control architectures

2.3.1 Introduction

In a previous work of our group [65], a distributed MPC architecture for nonlinear process systems based on the scheme shown in Fig. 2.1 was introduced. In this distributed MPC architecture, two MPC controllers designed via LMPC were considered. One of the two LMPC controllers (LMPC 1) was designed to guarantee the stability of the closed-loop system and the other LMPC controller (LMPC 2) was designed to improve the closed-loop performance while maintaining the closed-loop stability achieved by LMPC 1. This distributed MPC architecture required one-directional communication between the two distributed controllers and was proved that it guarantees practical stability of the closed-loop system and has the potential to maintain the closed-loop stability and performance in the face of new or failing controllers or actuators (for example, a zero input of LMPC 2 does not affect the closed-loop stability) and to reduce computational burden in the evaluation of the optimal manipulated inputs compared with a fully centralized LMPC controller of the same input/output-space dimension.

In this study, our objective is to extend our results in [65] and propose distributed MPC architectures including multiple MPCs for large scale nonlinear process systems. Specifically, we propose two different distributed MPC architectures. The first distributed MPC architecture is a direct extension of our previous work in [65] in which different MPC controllers are evaluated in sequence, only once at each sampling time and require only one-directional communication between consecutive distributed controllers (i.e., the distributed controllers are connected by pairs). In the second ar-

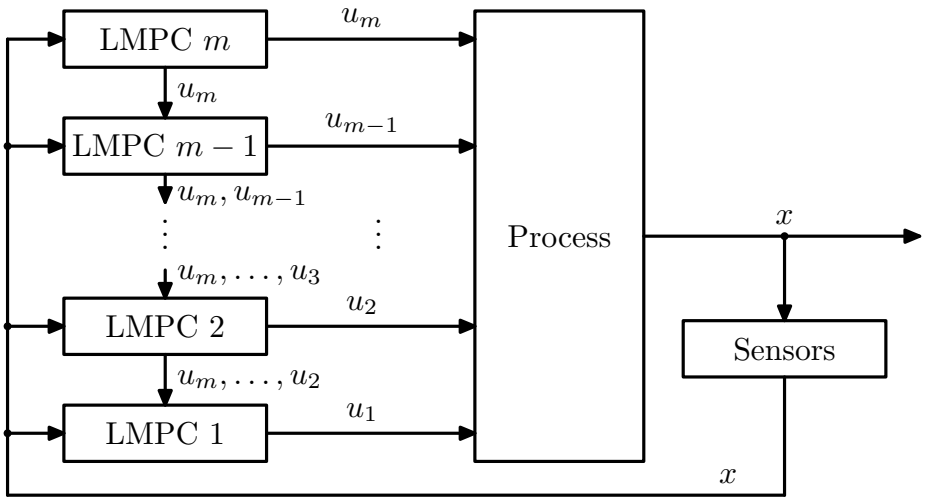


Figure 2.2: Sequential distributed LMPC architecture.

chitecture, different MPC controllers are evaluated in parallel, once or more than once at each sampling time depending on the number of iterations, and bi-directional communication among all the distributed controllers (i.e., the distributed controllers are all interconnected) is used.

In each proposed architecture, we will design m LMPC controllers to compute u_i, $i = 1, \ldots, m$, and refer to the LMPC controller computing the input trajectories of u_i as LMPC i.

2.3.2 Sequential distributed LMPC

In this section, we will discuss the direct extension of the results in [65] to include multiple LMPC controllers, in which different LMPC controllers are evaluated in sequence, once at each sampling time and one-directional communication between consecutive distributed controllers (i.e., the distributed controllers are connected by pairs) is used. A schematic of this architecture is shown in Fig. 2.2. We first present the proposed implementation strategy of this distributed MPC architecture and then

design the corresponding LMPC controllers. The proposed implementation strategy of this distributed MPC architecture is as follows:

1. At each sampling time t_k, all the LMPC controllers receive the state measurement $x(t_k)$ from the sensors.

2. For $j = m$ to 1

   2.1 LMPC j receives the entire future input trajectories of u_i, $i = m, \ldots, j+1$, from LMPC $j+1$ and evaluates the future input trajectory of u_j based on $x(t_k)$ and the received future input trajectories.

   2.2 LMPC j sends the first step input value of u_j to its actuators and the entire future input trajectories of u_i, $i = m, \ldots, j$, to LMPC $j-1$.

In this architecture, each LMPC controller only sends its future input trajectory and the future input trajectories it received to the next LMPC controller (i.e., LMPC j sends input trajectories to LMPC $j-1$). This implies that LMPC j, $j = m, \ldots, 2$, does not have any information about the values that u_i, $i = j-1, \ldots, 1$ will take when the optimization problems of the LMPC controllers are designed. In order to make a decision, LMPC j, $j = m, \ldots, 2$ must assume trajectories for u_i, $i = j-1, \ldots, 1$, along the prediction horizon. To this end, the Lyapunov-based controller $h(x)$ is used. In order to inherit the stability properties of the controller $h(x)$ (i.e., the stability properties of $h(x)$), each control input u_i, $i = 1, \ldots, m$ must satisfy a constraint that guarantees a given minimum contribution to the decrease rate of the Lyapunov function $V(x)$. Specifically, the proposed design of the LMPC j, $j = 1, \ldots, m$, is based on the following optimization problem:

$$u^*_{s,j}(\tau|t_k) = \arg \min_{u_{s,j} \in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T(\tau) Q_c \tilde{x}(\tau) + \sum_{i=1}^{m} u_{s,i}(\tau)^T R_{ci} u_{s,i}(\tau)] d\tau \tag{2.7a}$$

$$\text{s.t. } \dot{\tilde{x}}(\tau) = f(\tilde{x}(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}(\tau))u_{s,i} \tag{2.7b}$$

$$u_{s,i}(\tau) = h_i(\tilde{x}(j\Delta)),\ i = 1, \ldots, j-1,\ \forall\ \tau \in [l\Delta, (l+1)\Delta),\ l = 0, ..., N-1 \tag{2.7c}$$

$$u_{s,i}(\tau) = u^*_{s,i}(\tau|t_k), i = j+1, \ldots, m \tag{2.7d}$$

$$u_{s,j}(\tau) \in U_j \tag{2.7e}$$

$$\tilde{x}(0) = x(t_k) \tag{2.7f}$$

$$\frac{\partial V(x)}{\partial x} g_j(x(t_k))u_{s,j}(0) \leq \frac{\partial V(x)}{\partial x} g_j(x(t_k))h_j(x(t_k)) \tag{2.7g}$$

where $\tilde{x}$ is the predicted trajectory of the nominal system with u_i, $i = j+1, \ldots, m$, the input trajectory computed by the LMPC controllers of Eq. 2.7 evaluated before LMPC j, u_i, $i = 1, \ldots, j-1$, the corresponding elements of $h(x)$ applied in a sample-and-hold fashion and $u^*_{s,i}(\tau|t_k)$ denotes the future input trajectory of u_i obtained by LMPC i of the form of Eq. 2.7. The optimal solution to the optimization problem of Eq. 2.7 is denoted $u^*_{s,j}(\tau|t_k)$ which is defined for $\tau \in [0, N\Delta)$.

The constraint of Eq. 2.7b is the nominal model of the system of Eq. 2.1, which is used to predict the future evolution of the system; the constraints of Eq. 2.7c define the value of the inputs evaluated after u_j (i.e., u_i with $i = 1, \ldots, j-1$); the constraints of Eq. 2.7d define the value of the inputs evaluated before u_j (i.e., u_i with $i = j+1, \ldots, m$); the constraint of Eq. 2.7e is the constraint on the manipulated input u_j; the constraint of Eq. 2.7f sets the initial state for the optimization problem; the constraint of Eq. 2.7g guarantees that the contribution of input u_j to the decrease rate of the time derivative of the Lyapunov function at the initial evaluation time, if

$u_j = u^*_{s,j}(0|t_k)$ is applied, is bigger or equal to the value obtained when $u_j = h_j(x(t_k))$ is applied. This constraint allows proving the closed-loop stability properties of the proposed controller.

The manipulated inputs of the proposed control design of Eq. 2.7 are defined as follows:

$$u_i(t) = u^*_{s,i}(t - t_k|t_k),\ i = 1, \ldots, m,\ \forall t \in [t_k, t_{k+1}). \tag{2.8}$$

In what follows, we refer to this distributed LMPC architecture as the sequential distributed LMPC.

Remark 2.4 *Note that, in order to simplify the description of the implementation strategy proposed above in this section, we do not distinguish LMPC m and LMPC 1 from the others. We note that LMPC m does not receive any information from the other controllers and LMPC* 1 *does not have to send information to any other controller.*

The proposed distributed LMPC architecture of Eqs. 2.7-2.8 computes the inputs u_i, $i = 1, \ldots, m$, applied to the system of Eq. 2.1 in a way such that in the closed-loop system, the value of the Lyapunov function at time instant t_k (i.e., $V(x(t_k))$) is a decreasing sequence of values with a lower bound. Following Lyapunov arguments, this property guarantees practical stability of the closed-loop system. This is achieved due to the constraint of Eq. 2.7g. This property is presented in Theorem 2.1 below.

Theorem 2.1 *Consider the system of Eq. 2.1 in closed-loop under the distributed LMPC of Eqs. 2.7-2.8 based on the controller $h(x)$ that satisfies the condition of Eq. 2.2 with class $\mathcal{K}$ functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$. Let $\epsilon_w > 0$, $\Delta > 0$ and $\rho > \rho_s > 0$*

satisfy the following constraint:

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + L^* \leq -\epsilon_w/\Delta \tag{2.9}$$

where $L^* = (L_x + \sum_{i=1}^{m} L_{u_i} u_i^{\max})M + L_w\theta$ *with* M, L_x, L_{u_i} $(i = 1, \ldots, m)$ *and* L_w *being defined in Eqs. 2.3-2.4. For any* $N \geq 1$, *if* $x(t_0) \in \Omega_\rho$ *and if* $\rho^* \leq \rho$ *where*

$$\rho^* = \max\{V(x(t + \Delta)) : V(x(t)) \leq \rho_s\}, \tag{2.10}$$

then the state $x(t)$ *of the closed-loop system is ultimately bounded in* Ω_{ρ^*}.

Proof 2.1 The proof consists of two parts. We first prove that the optimization problem of Eq. 2.7 is feasible for all $j = 1, \ldots, m$ and $x \in \Omega_\rho$. Then we prove that, under the proposed distributed LMPC of Eqs. 2.7-2.8, the state of the system of Eq. 2.1 is ultimately bounded in Ω_{ρ^*}. Note that the constraint of Eq. 2.7g of each distributed controller is independent from the decisions that the rest of the distributed controllers make.

Part 1: In order to prove the feasibility of the optimization problem of Eq. 2.7, we only have to prove that there exists a $u_{s,j}(0)$ which satisfies the input constraint of Eq. 2.7e and the constraint of Eq. 2.7g. This is because the constraint of Eq. 2.7g is only enforced on the first prediction step of $u_{s,j}(\tau)$ and in the prediction time $\tau \in [\Delta, N\Delta)$, the input constraint of Eq. 2.8 can be easily satisfied with $u_{s,j}(\tau)$ being any value in the convex set U_j.

We assume that $x(t_k) \in \Omega_\rho$ ($x(t)$ is bounded in Ω_ρ which will be proved in *Part 2*). It is easy to verify that the value of $u_{s,j}$ such that $u_{s,j}(0) = h_j(x(t_k))$ satisfies the input constraint of Eq. 2.7e (assumed property of $h(x)$ for $x \in \Omega_\rho$) and the constraint of Eq. 2.7g, thus, the feasibility of the optimization problem of LMPC j, $j = 1, \ldots, m$,

is guaranteed.

Part 2: From the condition of Eq. 2.2 and the constraint of Eq. 2.7g, if $x(t_k) \in \Omega_\rho$, it follows that

$$\begin{aligned} \frac{\partial V}{\partial x}(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k))u^*_{s,i}(0|t_k)) &\leq \frac{\partial V}{\partial x}(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k))h_i(x(t_k)) \\ &\leq -\alpha_3(|x(t_k)|). \end{aligned} \tag{2.11}$$

The time derivative of the Lyapunov function V along the actual state trajectory $x(t)$ of the system of Eq. 2.1 in $t \in [t_k, t_{k+1})$ is given by

$$\dot{V}(x(t)) = \frac{\partial V}{\partial x}(f(x(t)) + \sum_{i=1}^{m} g_i(x(t))u^*_{s,i}(0|t_k) + k(x(t))w(t)). \tag{2.12}$$

Adding and subtracting $\frac{\partial V}{\partial x}(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k))u^*_{s,i}(0|t_k))$ and taking into account Eq. 2.11, we obtain the following inequality

$$\begin{aligned} \dot{V}(x(t)) \leq\ & -\alpha_3(|x(t_k)|) + \frac{\partial V}{\partial x}(f(x(t)) + \sum_{i=1}^{m} g_i(x(t))u^*_{s,i}(0|t_k) + k(x(t))w(t)) \\ & -\frac{\partial V}{\partial x}(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k))u^*_{s,i}(0|t_k)). \end{aligned} \tag{2.13}$$

Taking into account Eqs. 2.2 and 2.3, the following inequality if obtained for all $x(t_k) \in \Omega_\rho / \Omega_{\rho_s}$[§] from Eq. 2.13

$$\dot{V}(x(t)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + (L_x + \sum_{i=1}^{m} L_{u_i} u^*_{s,i}(0|t_k))|x(t) - x(t_k)| + L_w|w(t)|. \tag{2.14}$$

Taking into account Eq. 2.3 and the continuity of $x(t)$, the following bound can be

[§]The operator "/" is used to denote set subtraction, i.e., $A/B := \{x \in R^{n_x} : x \in A, x \notin B\}$.

written for all $t \in [t_k, t_{k+1})$

$$|x(t) - x(t_k)| \leq M\Delta.$$

Using this expression, the bounds on the disturbance $w(t)$ and the inputs u_i, $i = 1, \ldots, m$, and Eq. 2.14, we obtain the following bound on the time derivative of the Lyapunov function for $t \in [t_k, t_{k+1})$, for all initial states $x(t_k) \in \Omega_\rho / \Omega_{\rho_s}$

$$\dot{V}(x(t)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + (L_x + \sum_{i=1}^{m} L_{u_i} u_i^{\max})M + L_w\theta. \tag{2.15}$$

If the condition of Eq. 2.9 is satisfied, then there exists $\epsilon_w > 0$ such that the following inequality holds for $x(t_k) \in \Omega_\rho / \Omega_{\rho_s}$

$$\dot{V}(x(t)) \leq -\epsilon_w / \Delta \tag{2.16}$$

for $t \in [t_k, t_{k+1})$. Integrating the inequality of Eq. 2.16 on $t \in [t_k, t_{k+1})$, we obtain that

$$V(x(t_{k+1}) \leq V(x(t_k)) - \epsilon_w \tag{2.17}$$

and

$$V(x(t)) \leq V(x(t_k)),\ \forall t \in [t_k, t_{k+1}) \tag{2.18}$$

for all $x(t_k) \in \Omega_\rho / \Omega_{\rho_s}$. Using Eqs. 2.17 and 2.18 recursively it can be proved that, if $x(t_0) \in \Omega_\rho / \Omega_{\rho_s}$, the state converges to Ω_{ρ_s} in a finite number of sampling times without leaving the stability region. Once the state converges to $\Omega_{\rho_s} \subseteq \Omega_{\rho^*}$, it remains inside Ω_{ρ^*} for all times. This statement holds because of the definition of ρ^*. This proves that the closed-loop system under the proposed distributed LMPC of Eqs. 2.7-2.8 is ultimately bounded in Ω_{ρ^*}.

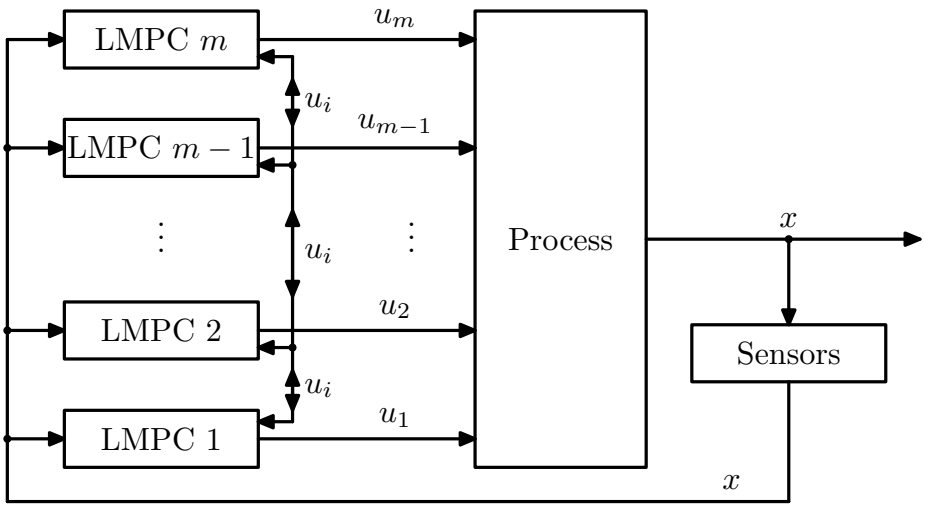


Figure 2.3: Iterative distributed LMPC architecture.

2.3.3 Iterative distributed LMPC

An alternative to the sequential distributed LMPC architecture presented in the previous section is to evaluate all the distributed LMPCs in parallel and iterate to improve closed-loop performance. A schematic of this control architecture is shown in Fig. 2.3. In this architecture, each distributed LMPC controller must be able to communicate with all the other controllers (i.e., the distributed controllers are all interconnected). More specifically, when a new state measurement is available at a sampling time, each distributed LMPC controller evaluates and obtains its future input trajectory; and then each LMPC controller broadcasts its latest obtained future input trajectory to all the other controllers. Based on the newly received input trajectories, each LMPC controller evaluates its future input trajectory again and this process is repeated until a certain termination condition is satisfied. Specifically, we proposed to use the following implementation strategy:

1. At each sampling time t_k, all the LMPC controllers receive the state measurement $x(t_k)$ from the sensors.

2. At iteration c ($c \geq 1$):

   2.1 All the distributed LMPC controllers exchange their latest future input trajectories.

   2.2 Each LMPC controller evaluates its own future input trajectory based on $x(t_k)$ and the latest received input trajectories of all the other LMPC controllers.

3. If the termination condition is satisfied, each LMPC controller sends the first step input value of its latest input trajectory to its actuators; if the termination condition is not satisfied, go to step 2 ($c = c + 1$).

Note that at the initial iteration, all the LMPC controllers use $h(x)$ to estimate the input trajectories of all the other controllers. Note also that the number of iterations c can be variable and it does not affect the closed-loop stability of the proposed distributed LMPC architecture; a point that will be made clear below. For the iterations in this distributed LMPC architecture, there are different choices of the termination condition. For example, the number of iterations c may be restricted to be smaller than a maximum iteration number $c_{\max}$ (i.e., $c \leq c_{\max}$) or the iterations may be terminated when the difference of the performance or the solution between two consecutive iterations is smaller than a threshold value or the iterations maybe terminated when a maximum computational time is reached.

In order to proceed, we define $\hat{x}(\tau|t_k)$ for $\tau \in [0, N\Delta)$ as the nominal sampled trajectory of the system of Eq. 2.1 associated with the feedback control law $h(x)$ and sampling time Δ starting from $x(t_k)$. This nominal sampled trajectory is obtained

by integrating recursively the following differential equation:

$$\dot{\hat{x}}(\tau|t_k) = f(\hat{x}(\tau|t_k)) + \sum_{i=1}^{m} g_i(\hat{x}(\tau|t_k))h_i(\hat{x}(l\Delta|t_k)),\ \forall \tau \in [(l\Delta,(l+1)\Delta)),$$

$$l = 0,\ldots,N-1.$$

Based on $\hat{x}(\tau|t_k)$, we can define the following variable

$$u_{p,j}^{*,0}(\tau|t_k) \quad = \quad h_j(\hat{x}(l\Delta|t_k)),\ j=1,\ldots,m,\ \forall \tau \in [(l\Delta,(l+1)\Delta)),\ l=0,\ldots,N-1.$$

which will be used as the initial guess of the trajectory of u_j.

The proposed design of the LMPC j, $j = 1,\ldots,m$, at iteration c is based on the following optimization problem:

$$u_{p,j}^{*,c}(\tau|t_k) = \arg\min_{u_{p,j}\in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T(\tau)Q_c\tilde{x}(\tau) + \sum_{i=1}^{m} u_{p,i}(\tau)^T R_{ci} u_{p,i}(\tau)]d\tau \tag{2.20a}$$

$$\text{s.t. } \dot{\tilde{x}}(\tau) = f(\tilde{x}(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}(\tau))u_{p,i} \tag{2.20b}$$

$$u_{p,i}(\tau) = u_{p,i}^{*,c-1}(\tau|t_k),\ \forall i \neq j \tag{2.20c}$$

$$u_{p,j}(\tau) \in U_j \tag{2.20d}$$

$$\tilde{x}(0) = x(t_k) \tag{2.20e}$$

$$\frac{\partial V(x)}{\partial x} g_j(x(t_k))u_{p,j}(0) \leq \frac{\partial V(x)}{\partial x} g_j(x(t_k))h_j(x(t_k)) \tag{2.20f}$$

where $\tilde{x}$ is the predicted trajectory of the nominal system with u_k, the input trajectory, computed by the LMPCs of Eq. 2.20 and all the other inputs are the optimal input

trajectories at iteration $c-1$ of the rest of distributed controllers. The optimal solution to the optimization problem of Eq. 2.20 is denoted $u_{p,j}^{*,c}(\tau|t_k)$ which is defined for $\tau \in [0, N\Delta)$. Accordingly, we define the final optimal input trajectory of LMPC j (that is, the optimal trajectories computed at the last iteration) as $u_{p,j}^{*}(\tau|t_k)$ which is also defined for $\tau \in [0, N\Delta)$.

The manipulated inputs of the proposed control design of Eq. 2.20 are defined as follows:

$$u_i(t) = u_{p,i}^{*}(t - t_k|t_k),\ i = 1, \ldots, m,\ \forall t \in [t_k, t_{k+1}). \tag{2.21}$$

In what follows, we refer to this distributed LMPC architecture as the iterative distributed LMPC. The stability property of the iterative distributed LMPC is stated in the following Theorem 2.2.

Theorem 2.2 *Consider the system of Eq. 2.1 in closed-loop under the distributed LMPC of Eqs. 2.20-2.21 based on the controller $h(x)$ that satisfies the condition of Eq. 2.2. Let $\epsilon_w > 0$, $\Delta > 0$ and $\rho > \rho_s > 0$ satisfy the constraint of Eq. 2.9. For any $N \geq 1$ and $c \geq 1$, if $x(t_0) \in \Omega_\rho$ and if $\rho^* \leq \rho$ where ρ^* is defined as in Eq. 2.10, then the state $x(t)$ of the closed-loop system is ultimately bounded in Ω_{ρ^*}.*

Proof 2.2 Similar to the proof of Theorem 2.1, the proof of Theorem 2.2 also consists of two parts. We first prove that the optimization problem of Eq. 2.20 is feasible for each iteration c and $x \in \Omega_\rho$. Then we prove that, under the proposed distributed LMPC scheme of Eqs. 2.20-2.21, the state of the system of Eq. 2.1 is ultimately bounded in Ω_{ρ^*}.

Part 1: In order to prove the feasibility of the optimization problem of Eq. 2.20, we only have to prove that there exists a $u_{p,j}(0)$ which satisfies the input constraint of Eq. 2.20d and the constraint of Eq. 2.20f. This is because the constraint of Eq. 2.20f

is only enforced on the first prediction step of $u_{p,j}(\tau)$ and in the prediction time $\tau \in [\Delta, N\Delta)$, the input constraint of Eq. 2.21 can be easily satisfied with $u_{p,j}(\tau)$ being any value in the convex set U_j.

We assume that $x(t_k) \in \Omega_\rho$ ($x(t)$ is bounded in Ω_ρ which will be proved in *Part 2*). It is easy to verify that the value of $u_{p,j}$ such that $u_{p,j}(0) = h_j(x(t_k))$ satisfies the input constraint of Eq. 2.20d (assumed property of $h(x)$ for $x \in \Omega_\rho$) and the constraint of Eq. 2.20f for all possible c, thus, the feasibility of LMPC j, $j = 1, \ldots, m$, is guaranteed.

Part 2: By adding the constraint of Eq. 2.20f of each LMPC together, we have

$$\sum_{j=1}^{m} \frac{\partial V(x)}{\partial x} g_j(x(t_k)) u_{p,j}^{*,c}(0|t_k) \leq \sum_{j=1}^{m} \frac{\partial V(x)}{\partial x} g_j(x(t_k)) h_j(x(t_k))$$

It follows from the above inequality and condition of Eq. 2.2 that

$$\begin{aligned} \frac{\partial V}{\partial x}(f(x(t_k)) + \sum_{j=1}^{m} g_j(x(t_k)) u_{p,j}^{*,c}(0|t_k)) &\leq \frac{\partial V}{\partial x}(f(x(t_k)) + \sum_{j=1}^{m} g_j(x(t_k)) h_j(x(t_k))) \\ &\leq -\alpha_3(|x(t_k)|). \end{aligned} \tag{2.22}$$

Following the same approach as in the proof of Theorem 2.1, we know that if condition of Eq. 2.9 is satisfied, then the state of the closed-loop system can be proved to be maintained in Ω_{ρ^*} under the proposed distributed LMPC architecture of Eqs. 2.20-2.21.

Remark 2.5 *Note that the distributed LMPC designs have the same stability region Ω_ρ as the one of the Lyapunov-based controller $h(x)$. When the stability of the Lyapunov-based controller $h(x)$ is global (i.e., the stability region is the entire state space), then the stability of the distributed LMPC designs is also global. Note also that for any initial condition in Ω_ρ, the distributed LMPC designs are proved to be*

feasible.

Remark 2.6 *We do not consider delays introduced into the system by the communication network or by the time needed to solve the optimization problems. In future works, these delays will be taken into account in the formulation of the controllers. In this work, state constraints have also not been considered but the proposed designs can be extended to handle state constraints by restricting the closed-loop stability region further to satisfy the state constraints.*

Remark 2.7 *The choice of the horizon of the distributed LMPC designs does not affect the stability of the closed-loop system. For any horizon length* $N \geq 1$*, the closed-loop stability is guaranteed by the constraints of Eqs. 2.7g and 2.20f. However, the choice of the horizon does affect the performance of the distributed LMPC designs.*

Remark 2.8 *Note that because the manipulated inputs enter the dynamics of the system of Eq. 2.1 in an affine manner, the constraints designed in the LMPC optimization problems of Eqs. 2.7 and 2.20 to guarantee the closed-loop stability can be decoupled for different distributed controllers as in Eqs. 2.7g and 2.20f.*

Remark 2.9 *In the sequential distributed LMPC architecture presented in Section 2.3.2, the distributed controllers are evaluated in sequence, which implies that the minimal time to obtain a set of solutions to all the LMPC controllers is the sum of the evaluation times of all the LMPC controllers; whereas in the iterative distributed LMPC architecture proposed in Section 2.3.3, the distributed controllers are evaluated in parallel, which implies that the minimal time to obtain a set of solutions to all the LMPC controllers in each iteration is the largest evaluation time among all the LMPCs.*

Remark 2.10 *Note that the sequential (or iterative) distributed LMPC is not a direct decomposition of the centralized LMPC because the set of constraints of Eq. 2.7g (or Eq. 2.20f) for* $j = 1, \ldots, m$ *in the distributed LMPC formulation of Eq. 2.7 (or Eq. 2.20) imposes a different feasibility region from the one of the centralized LMPC of Eq. 2.5 which has a single constraint (Eq. 2.5e).*

Remark 2.11 *In general, there is no guaranteed convergence of the optimal cost or solution of an iterated distributed MPC (e.g., the distributed MPC architecture discussed in Section 2.3.3) to the optimal cost or solution of a centralized MPC for general nonlinear constrained systems because of the non-convexity of the MPC optimization problems. The reader may refer to [10, 91] for discussions on the conditions under which convergence of the solution of a distributed linear or convex MPC design to the solution of a centralized MPC or a Pareto optimal solution is ensured in the context of linear systems.*

Remark 2.12 *Note also that in general there is no guarantee that the closed-loop performance of one (centralized or distributed) MPC architecture discussed in this work should be superior than the others since the solutions provided by these MPC architectures are proved to be feasible and stabilizing but the superiority of the performance of one MPC architecture over another is not established. This is because the MPC designs are implemented in a receding horizon scheme and the prediction horizon is finite; and also because the different MPC designs are not equivalent as we discussed in Remark 2.10 and the non-convexity property as we discussed in Remark 2.11. In applications of these MPC architectures, especially for chemical process control in which non-convex problems is a very common occurrence, simulations should be conducted before making decisions as to which architecture should be used.*

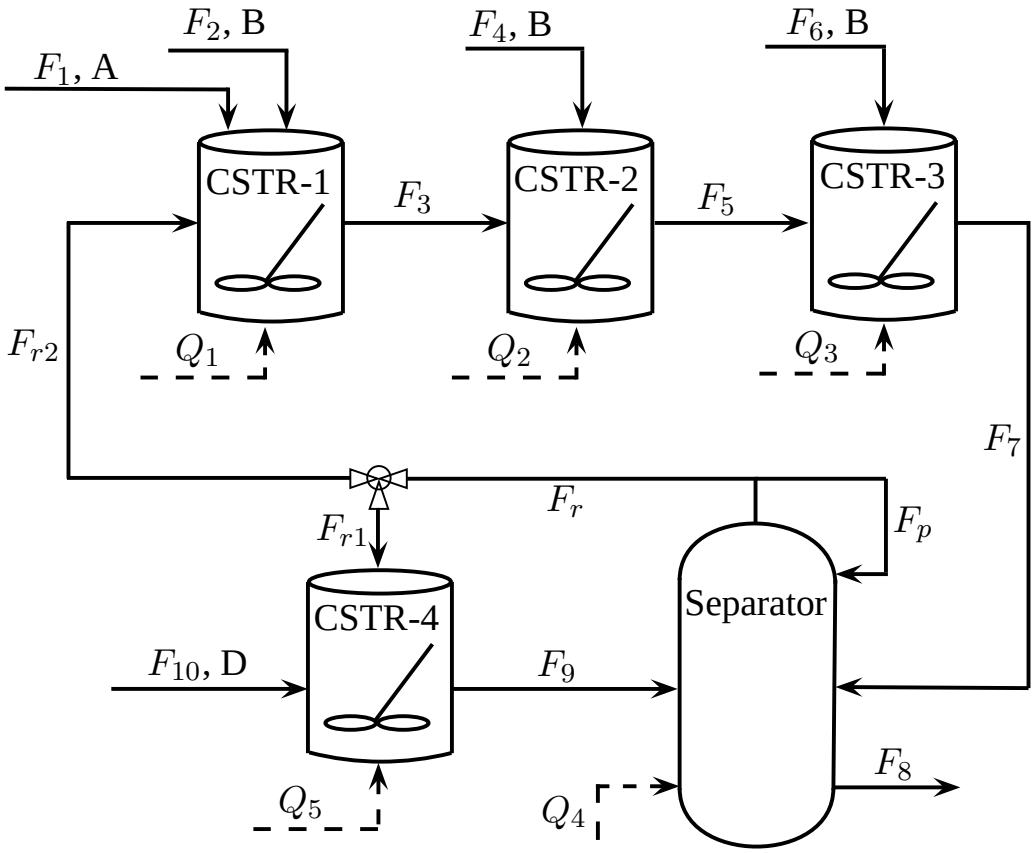


Figure 2.4: Process flow diagram of alkylation of benzene.

2.4 Application to a chemical process

2.4.1 Process description and modeling

The process of alkylation of benzene with ethylene to produce ethylbenzene is widely used in the petrochemical industry. Dehydration of the product produces styrene, which is the precursor to polystyrene and many copolymers. Over the last two decades, several methods and simulation results of alkylation of benzene with catalysts have been reported in the literature. The process model developed in this section is based on these references [35, 58, 84, 110]. More specifically, the process considered in this work consists of four continuously stirred tank reactors (CSTRs)

and a flash tank separator, as shown in Fig. 2.4. The CSTR-1, CSTR-2 and CSTR-3 are in series and involve the alkylation of benzene with ethylene. Pure benzene is fed from stream F_1 and pure ethylene is fed from streams F_2, F_4 and F_6. Two catalytic reactions take place in CSTR-1, CSTR-2 and CSTR-3. Benzene (A) reacts with ethylene (B) and produces the required product ethylbenzene (C) (reaction 1); ethylbenzene can further react with ethylene to form 1,3-diethylbenzene (D) (reaction 2) which is the byproduct. The effluent of CSTR-3, including the products and leftover reactants, is fed to a flash tank separator, in which most of benzene is separated overhead by vaporization and condensation techniques and recycled back to the plant and the bottom product stream is removed. A portion of the recycle stream F_{r2} is fed back to CSTR-1 and another portion of the recycle stream F_{r1} is fed to CSTR-4 together with an additional feed stream F_{10} which contains 1,3-diethylbenzene from further distillation process that we do not consider in this example. In CSTR-4, reaction 2 and catalyzed transalkylation reaction in which 1,3-diethylbenzene reacts with benzene to produce ethylbenzene (reaction 3) takes place. All chemicals left from CSTR-4 eventually pass into the separator. All the materials in the reactions are in liquid phase due to high pressure. The dynamic equations describing the behavior of the process, obtained through material and energy balances under standard modeling assumptions, are given below:

$$\frac{dC_{A1}}{dt} = \frac{F_1 C_{A0} + F_{r2} C_{Ar} - F_3 C_{A1}}{V_1} - r_1(T_1, C_{A1}, C_{B1}) \tag{2.23a}$$

$$\frac{dC_{B1}}{dt} = \frac{F_2 C_{B0} + F_{r2} C_{Br} - F_3 C_{B1}}{V_1} - r_1(T_1, C_{A1}, C_{B1}) - r_2(T_1, C_{B1}, C_{C1}) \tag{2.23b}$$

$$\frac{dC_{C1}}{dt} = \frac{F_{r2} C_{Cr} - F_3 C_{C1}}{V_1} + r_1(T_1, C_{A1}, C_{B1}) - r_2(T_1, C_{B1}, C_{C1}) \tag{2.23c}$$

$$\frac{dC_{D1}}{dt} = \frac{F_{r2}C_{Dr} - F_3C_{D1}}{V_1} + r_2(T_1, C_{B1}, C_{C1}) \tag{2.23d}$$

$$\frac{dT_1}{dt} = \frac{Q_1 + F_1C_{A0}H_A(T_{A0}) + F_2C_{B0}H_B(T_{B0})}{\sum_i^{A,B,C,D} C_{i1}C_{pi}V_1}$$

$$+ \frac{\sum_i^{A,B,C,D} (F_{r2}C_{ir}H_i(T_4) - F_3C_{i1}H_i(T_1))}{\sum_i^{A,B,C,D} C_{i1}C_{pi}V_1}$$

$$+ \frac{-\Delta H_{r1}r_1(T_1, C_{A1}, C_{B1}) - \Delta H_{r2}r_2(T_1, C_{B1}, C_{C1})}{\sum_i^{A,B,C,D} C_{i1}C_{pi}V_1} \tag{2.23e}$$

$$\frac{dC_{A2}}{dt} = \frac{F_3C_{A1} - F_5C_{A2}}{V_2} - r_1(T_2, C_{A2}, C_{B2}) \tag{2.23f}$$

$$\frac{dC_{B2}}{dt} = \frac{F_3C_{B1} + F_4C_{B0} - F_5C_{B2}}{V_2} - r_1(T_2, C_{A2}, C_{B2}) - r_2(T_2, C_{B2}, C_{C2}) \tag{2.23g}$$

$$\frac{dC_{C2}}{dt} = \frac{F_3C_{C1} - F_5C_{C2}}{V_2} + r_1(T_2, C_{A2}, C_{B2}) - r_2(T_2, C_{B2}, C_{C2}) \tag{2.23h}$$

$$\frac{dC_{D2}}{dt} = \frac{F_3C_{D1} - F_5C_{R2}}{V_2} + r_2(T_2, C_{B2}, C_{C2}) \tag{2.23i}$$

$$\frac{dT_2}{dt} = \frac{Q_2 + F_4C_{B0}H_B(T_{B0}) + \sum_i^{A,B,C,D} (F_3C_{i1}H_i(T_1) - F_5C_{i2}H_i(T_2))}{\sum_i^{A,B,C,D} C_{i2}C_{pi}V_2}$$

$$+ \frac{-\Delta H_{r1}r_1(T_2, C_{A2}, C_{B2}) - \Delta H_{r2}r_2(T_2, C_{A2}, C_{B2})}{\sum_i^{A,B,C,D} C_{i2}C_{pi}V_2} \tag{2.23j}$$

$$\frac{dC_{A3}}{dt} = \frac{F_5C_{A2} - F_7C_{A3}}{V_3} - r_1(T_3, C_{A3}, C_{B3}) \tag{2.23k}$$

$$\frac{dC_{B3}}{dt} = \frac{F_5C_{B2} + F_6C_{B0} - F_7C_{B3}}{V_3} - r_1(T_3, C_{A3}, C_{B3}) - r_2(T_3, C_{B3}, C_{C3}) \tag{2.23l}$$

$$\frac{dC_{C3}}{dt} = \frac{F_5 C_{C2} - F_7 C_{C3}}{V_3} + r_1(T_3, C_{A3}, C_{B3}) - r_2(T_3, C_{B3}, C_{C3}) \tag{2.23m}$$

$$\frac{dC_{D3}}{dt} = \frac{F_5 C_{D2} - F_7 C_{D3}}{V_3} + r_2(T_3, C_{B3}, C_{C3}) \tag{2.23n}$$

$$\frac{dT_3}{dt} = \frac{Q_3 + F_6 C_{B0} H_B(T_{B0}) + \sum_{i}^{A,B,C,D} (F_5 C_{i2} H_i(T_2) - F_7 C_{i3} H_i(T_3))}{\sum_{i}^{A,B,C,D} C_{i3} C_{pi} V_3}$$

$$+ \frac{-\Delta H_{r1} r_1(T_3, C_{A3}, C_{B3}) - \Delta H_{r2} r_2(T_3, C_{B3}, C_{C3})}{\sum_{i}^{A,B,C,D} C_{i3} C_{pi} V_3} \tag{2.23o}$$

$$\frac{dC_{A4}}{dt} = \frac{F_7 C_{A3} + F_9 C_{A5} - F_r C_{Ar} - F_8 C_{A4}}{V_4} \tag{2.23p}$$

$$\frac{dC_{B4}}{dt} = \frac{F_7 C_{B3} + F_9 C_{B5} - F_r C_{Br} - F_8 C_{B4}}{V_4} \tag{2.23q}$$

$$\frac{dC_{C4}}{dt} = \frac{F_7 C_{C3} + F_9 C_{C5} - F_r C_{Cr} - F_8 C_{C4}}{V_4} \tag{2.23r}$$

$$\frac{dC_{D4}}{dt} = \frac{F_7 C_{D3} + F_9 C_{D5} - F_r C_{Dr} - F_8 C_{D4}}{V_4} \tag{2.23s}$$

$$\frac{dT_4}{dt} = \frac{Q_4 + \sum_{i}^{A,B,C,D} (F_7 C_{i3} H_i(T_3) + F_9 C_{i5} H_i(T_5) - F_r C_{ir} H_i(T_4)}{\sum_{i}^{A,B,C,D} C_{i4} C_{pi} V_4}$$

$$- \frac{\sum_{i}^{A,B,C,D} (F_8 C_{i4} H_i(T_4) + F_r C_{ir} H_{vapi})}{\sum_{i}^{A,B,C,D} C_{i4} C_{pi} V_4} \tag{2.23t}$$

$$\frac{dC_{A5}}{dt} = \frac{F_{r1} C_{Ar} - F_9 C_{A5}}{V_5} - r_3(T_5, C_{A5}, C_{D5}) \tag{2.23u}$$

$$\frac{dC_{B5}}{dt} = \frac{F_{r1} C_{Br} - F_9 C_{B5}}{V_5} - r_2(T_5, C_{B5}, C_{C5}) \tag{2.23v}$$

$$\frac{dC_{C5}}{dt} = \frac{F_{r1} C_{Cr} - F_9 C_{C5}}{V_5} - r_2(T_5, C_{B5}, C_{C5}) + 2r_3(T_5, C_{A5}, C_{D5}) \tag{2.23w}$$

$$\frac{dC_{D5}}{dt} = \frac{F_{r1}C_{Dr} + F_{10}C_{D0} - F_9 C_{D5}}{V_5} + r_2(T_5, C_{B5}, C_{C5}) - r_3(T_5, C_{A5}, C_{D5}) \tag{2.23x}$$

$$\frac{dT_5}{dt} = \frac{Q_5 + F_{10}C_{D0}H_D(T_{D0}) + \sum_i^{A,B,C,D}(F_r C_{ir} H_i(T_4) - F_9 C_{i5} H_i(T_5))}{\sum_i^{A,B,C,D} C_{i5} C_{pi} V_5}$$

$$+ \frac{-\Delta H_{r2} r_2(T_5, C_{B5}, C_{C5}) - \Delta H_{r3} r_3(T_5, C_{A5}, C_{D5})}{\sum_i^{A,B,C,D} C_{i5} C_{pi} V_5} \tag{2.23y}$$

where $r_1,\ r_2$ and r_3 are the reaction rates of reactions 1, 2 and 3 respectively and H_i, $i = A,\ B,\ C,\ D$, are the enthalpies of the reactants. The reaction rates are related to the concentrations of the reactants and the temperature in each reactor as follows:

$$r_1(T, C_A, C_B) = k_{r1} C_A^{0.32} C_B^{1.5}$$

$$r_2(T, C_B, C_C) = \frac{k_{r2} C_B^{2.5} C_C^{0.5}}{(1 + k_{EB2} C_D)}$$

$$r_3(T, C_A, C_D) = \frac{k_{r3} C_A^{1.0218} C_D}{(1 + k_{EB3} C_A)}$$

with

$$k_{r1} = 0.0840 e^{(-9502/RT)}, k_{r2} = 0.0850 e^{(-20640/RT)}, k_{r3} = 237.8 e^{(-61280/RT)}$$

$$k_{EB2} = 0.0152 e^{(-3933/RT)}, k_{EB3} = 0.4901 e^{(-50870/RT)}.$$

The heat capacities of the species are assumed to be constants and the molar enthalpies have a linear dependence on temperature as follows:

$$H_i(T) = H_{iref} + C_{pi}(T - T_{ref}), \quad i = A, B, C, D$$

where C_{pi}, $i = A,\ B,\ C,\ D$ are heat capacities.

The model of the flash tank separator is developed under the assumption that the relative volatility of each species has a linear correlation with the temperature of the vessel within the operating temperature range of the flash tank, as shown below:

$$\begin{aligned} \alpha_A &= 0.0449T_4 + 10, \quad \alpha_B &= 0.0260T_4 + 10 \\ \alpha_C &= 0.0065T_4 + 0.5, \quad \alpha_D &= 0.0058T_4 + 0.25 \end{aligned}$$

where α_i, $i = A,\ B,\ C, D$, represent the relative volatilities. It has also been assumed that there is a negligible amount of reaction taking place in the separator. The following algebraic equations model the composition of the overhead stream relative to the composition of the liquid holdup in the flash tank:

$$M_i = (F_7 C_{i3} + F_9 C_{i5}) \frac{\alpha_i (F_7 C_{i3} + F_9 C_{i5})}{\sum_{k}^{A,B,C,D} \alpha_k (F_7 C_{k3} + F_9 C_{k5})}, \quad i = A, B, C, D$$

where M_i, $i = A,\ B,\ C,\ D$ are the molar flow rates of the overhead reactants. Based on M_i, $i = A,\ B,\ C,\ D$, we can calculate the concentration of the reactants in the recycle streams as follows:

$$C_{ir} = \frac{M_i}{\sum_{k}^{A,B,C,D} M_i / C_{k0}}, \quad i = A, B, C, D$$

where C_{k0}, $k = A, B, C, D$, are the mole densities of pure reactants. The condensation of vapor takes place overhead, and a portion of the condensed liquid is purged back to separator to keep the flow rate of the recycle stream at a fixed value. The temperature of the condensed liquid is assumed to be the same as the temperature of the vessel.

The definitions for the variables used in the above model can be found in Table 2.1, with the parameter values given in Table 2.2.

Each of the tanks has an external heat/coolant input. The manipulated inputs to the process are the heat injected to or removed from the five vessels, Q_1, Q_2, Q_3, Q_4 and Q_5, and the feed stream flow rates to CSTR-2 and CSTR-3, F_4 and F_6.

The states of the process consist of the concentrations of A, B, C, D in each of the five vessels and the temperatures of the vessels. The state of the process is assumed to be available continuously to the controllers. We consider a stable steady state (operating point), x_s, of the process which is defined by the steady-state inputs Q_{1s}, Q_{2s}, Q_{3s}, Q_{4s}, Q_{5s}, F_{4s} and F_{6s} which are shown in Table 2.3 with corresponding steady-state values shown in Table 2.4.

The control objective is to regulate the system from an initial state to the steady state. The initial state values are shown in Table 2.6.

The first distributed controller (LMPC 1) will be designed to decide the values of Q_1, Q_2 and Q_3, the second distributed controller (LMPC 2) will be designed to decide the values of Q_4 and Q_5, and the third distributed controller (LMPC 3) will be designed to decide the values of F_4 and F_6. Taking this into account, the process model of Eq. 2.23 belongs to the following class of nonlinear systems:

$$\dot{x}(t) = f(x) + g_1(x)u_1(x) + g_2(x)u_2(x) + g_3(x)u_3(x)$$

where the state x is the deviation of the state of the process from the steady state,

Table 2.1: Process variables of the alkylation of benzene process of Eqs. 2.23

C_{A1}, C_{B1}, C_{C1}, C_{D1}	Concentrations of $A,\ B,\ C,\ D$ in CSTR-1
C_{A2}, C_{B2}, C_{C2} ,C_{D2}	Concentrations of $A,\ B,\ C,\ D$ in CSTR-2
C_{A3}, C_{B3}, C_{C3}, C_{D3}	Concentrations of $A,\ B,\ C,\ D$ in CSTR-3
C_{A4}, C_{B4}, C_{C4}, C_{D4}	Concentrations of $A,\ B,\ C,\ D$ in separator
C_{A5}, C_{B5}, C_{C5}, C_{D5}	Concentrations of $A,\ B,\ C,\ D$ in CSTR-4
C_{Ar}, C_{Br}, C_{Cr}, C_{Dr}	Concentrations of $A,\ B,\ C,\ D$ in F_r, F_{r1}, F_{r2}
T_1, T_2, T_3, T_4, T_5	Temperatures in each vessel
T_ref	Reference temperature
F_3, F_5, F_7, F_8, F_9	Effluent flow rates from each vessel
F_1, F_2, F_4, F_6, F_10	Feed flow rates to each vessel
F_r, F_{r1}, F_{r2}	Recycle flow rates
H_{vapA}, H_{vapB}, H_{vapC}, H_{vapD}	Enthalpies of vaporization of $A,\ B,\ C,\ D$
H_{Aref}, H_{Bref}, H_{Cref}, H_{Dref}	Enthalpies of $A,\ B,\ C,\ D$ at T_{ref}
ΔH_{r1}, ΔH_{r2}, ΔH_{r3}	Heat of reactions 1, 2 and 3
V_1, V_2, V_3, V_4, V_5	Volume of each vessel
Q_1, Q_2, Q_3, Q_4, Q_5	External heat/coolant inputs to each vessel
C_{pA}, C_{pB}, C_{pC}, C_{pD}	Heat capacity of $A,\ B,\ C,\ D$ at liquid phase
α_A, α_B, α_C, α_D	Relative volatilities of $A,\ B,\ C,\ D$
C_{A0}, C_{B0}, C_{C0}, C_{D0}	Molar densities of pure $A,\ B,\ C,\ D$
T_{A0}, T_{B0}, T_{D0}	Feed temperatures of pure $A,\ B,\ D$

Table 2.2: Parameter values of the alkylation of benzene process of Eqs. 2.23

F_1	7.1×10^{-3} $[m^3/s]$	F_r	0.012 $[m^3/s]$
F_2	8.697×10^{-4} $[m^3/s]$	F_{r1}	0.006 $[m^3/s]$
F_{r2}	0.006 $[m^3/s]$	V_1	1 $[m^3]$
F_{10}	2.31×10^{-3} $[m^3/s]$	V_2	1 $[m^3]$
H_{vapA}	3.073×10^4 $[J/mole]$	V_3	1 $[m^3]$
H_{vapB}	1.35×10^4 $[J/mole]$	V_4	3 $[m^3]$
H_{vapC}	4.226×10^4 $[J/mole]$	V_5	1 $[m^3]$
H_{vapD}	4.55×10^4 $[J/mole]$	C_{pA}	184.6 $[J/mole \cdot K]$
H_{Aref}	7.44×10^4 $[J/mole]$	H_{Bref}	5.91×10^4 $[J/mole]$
H_{Cref}	2.02×10^4 $[J/mole]$	H_{Bref}	-2.89×10^4 $[J/mole]$
ΔH_{r1}	-1.536×10^5 $[J/mole]$	C_{pB}	59.1 $[J/mole \cdot K]$
ΔH_{r2}	-1.118×10^5 $[J/mole]$	C_{pC}	247 $[J/mole \cdot K]$
ΔH_{r3}	4.141×10^5 $[J/mole]$	C_{pD}	301.3 $[J/mole \cdot K]$
C_{A0}	1.126×10^4 $[mole/m^3]$	T_{ref}	450 $[K]$
C_{B0}	2.028×10^4 $[mole/m^3]$	T_{A0}	473 $[K]$
C_{C0}	8174 $[mole/m^3]$	T_{B0}	473 $[K]$
C_{D0}	6485 $[mole/m^3]$	T_{D0}	473 $[K]$
k	0.8		

Table 2.3: Steady-state input values for x_s of the alkylation of benzene process of Eqs. 2.23

Q_{1s}	-4.4×10^6 $[J/s]$	Q_{2s}	-4.6×10^6 $[J/s]$
Q_{3s}	-4.7×10^6 $[J/s]$	Q_{4s}	9.2×10^6 $[J/s]$
Q_{5s}	5.9×10^6 $[J/s]$	F_{4s}	8.697×10^{-4} $[m^3/s]$
F_{4s}	8.697×10^{-4} $[m^3/s]$		

Table 2.4: Steady-state values for x_s of the alkylation of benzene process of Eqs. 2.23

C_{A1}	9.101×10^3 $[mole/m^3]$	C_{A2}	7.548×10^3 $[mole/m^3]$
C_{B1}	22.15 $[mole/m^3]$	C_{B2}	23.46 $[mole/m^3]$
C_{C1}	1.120×10^3 $[mole/m^3]$	C_{C2}	1.908×10^3 $[mole/m^3]$
C_{D1}	2.120×10^2 $[mole/m^3]$	C_{D2}	3.731×10^2 $[mole/m^3]$
T_1	4.772×10^2 $[K]$	T_2	4.77×10^2 $[K]$
C_{A3}	6.163×10^3 $[mole/m^3]$	C_{A4}	1.723×10^3 $[mole/m^3]$
C_{B3}	24.84 $[mole/m^3]$	C_{B4}	13.67 $[mole/m^3]$
C_{C3}	2.616×10^3 $[mole/m^3]$	C_{C4}	5.473×10^3 $[mole/m^3]$
C_{D3}	5.058×10^2 $[mole/m^3]$	C_{D4}	7.044×10^2 $[mole/m^3]$
T_3	4.735×10^2 $[K]$	T_4	4.706×10^2 $[K]$
C_{A5}	5.747×10^3 $[mole/m^3]$	C_{D5}	1.537×10^2 $[mole/m^3]$
C_{B5}	3.995 $[mole/m^3]$	T_5	4.783×10^2 $[K]$
C_{C5}	3.830×10^3 $[mole/m^3]$		

Table 2.5: Manipulated input constraints of the alkylation of benzene process of Eqs. 2.23

$\lvert u_{11}\rvert \leq 7.5 \times 10^5\ [J/s]$	$\lvert u_{12}\rvert \leq 5 \times 10^5\ [J/s]$
$\lvert u_{13}\rvert \leq 5 \times 10^5\ [J/s]$	$\lvert u_{21}\rvert \leq 6 \times 10^5\ [J/s]$
$\lvert u_{22}\rvert \leq 5 \times 10^5\ [J/s]$	$\lvert u_{31}\rvert \leq 4.93 \times 10^{-5}\ [m^3/s]$
$\lvert u_{32}\rvert \leq 4.93 \times 10^{-5}\ [m^3/s]$	

Table 2.6: Initial state values of the alkylation of benzene process of Eqs. 2.23

C_{A1}	$9.112 \times 10^3\ [mole/m^3]$	C_{A2}	$7.557 \times 10^3\ [mole/m^3]$
C_{B1}	$25.09\ [mole/m^3]$	C_{B2}	$27.16\ [mole/m^3]$
C_{C1}	$1.113 \times 10^3\ [mole/m^3]$	C_{C2}	$1.905 \times 10^3\ [mole/m^3]$
C_{D1}	$2.186 \times 10^2\ [mole/m^3]$	C_{D2}	$3.695 \times 10^2\ [mole/m^3]$
T_1	$4.430 \times 10^2\ [K]$	T_2	$4.371 \times 10^2\ [K]$
C_{A3}	$6.170 \times 10^3\ [mole/m^3]$	C_{A4}	$1.800 \times 10^3\ [mole/m^3]$
C_{B3}	$29.45\ [mole/m^3]$	C_{B4}	$16.35\ [mole/m^3]$
C_{C3}	$2.617 \times 10^3\ [mole/m^3]$	C_{C4}	$5.321 \times 10^3\ [mole/m^3]$
C_{D3}	$5.001 \times 10^2\ [mole/m^3]$	C_{D4}	$7.790 \times 10^2\ [mole/m^3]$
T_3	$4.284 \times 10^2\ [K]$	T_4	$4.331 \times 10^2\ [K]$
C_{A5}	$5.889 \times 10^3\ [mole/m^3]$	C_{D5}	$2.790 \times 10^2\ [mole/m^3]$
C_{B5}	$5.733\ [mole/m^3]$	T_5	$4.576 \times 10^2\ [K]$
C_{C5}	$3.566 \times 10^3\ [mole/m^3]$		

$u_1^T = [u_{11}\ u_{12}\ u_{13}] = [Q_1 - Q_{1s}\ Q_2 - Q_{2s}\ Q_3 - Q_{3s}]$, $u_2^T = [u_{21}\ u_{22}] = [Q_4 - Q_{4s}\ Q_5 - Q_{5s}]$ and $u_3^T = [u_{31}\ u_{32}] = [F_4 - F_{4s}\ F_6 - F_{6s}]$ are the manipulated inputs which are subject to the constraints shown in Table 2.5.

In the control of the process, u_1 and u_2 are necessary to keep the stability of the closed-loop system, while u_3 can be used as an extra manipulated input to improve the closed-loop performance. To illustrate the theoretical results, we first design the Lyapunov-based controller $h(x) = [h_1(x)\ h_2(x)\ h_3(x)]^T$. Specifically, $h_1(x)$ and $h_2(x)$ are designed as follows [93]:

$$h_i(x) = \begin{cases} -\dfrac{L_f V + \sqrt{(L_f V)^2 + (L_{g_i} V)^4}}{(L_{g_i} V)^2} L_{g_i} V & \text{if } L_{g_i} V \neq 0 \\ 0 & \text{if } L_{g_i} V = 0 \end{cases} \tag{2.24}$$

where $i = 1,\ 2$, $L_f V = \dfrac{\partial V}{\partial x} f(x)$ and $L_{g_i} V = \dfrac{\partial V}{\partial x} g_i(x)$ denote the Lie derivatives of the scalar function V with respect to the vector fields f and g_i ($i = 1,\ 2$), respectively. The controller $h_3(x)$ is chosen to be $h_3(x) = [0\ 0]^T$ because the input set u_3 is not needed to stabilize the process. We consider a Lyapunov function $V(x) = x^T P x$ with P being the following weight matrix

$$P = diag^{\P}([1\ 1\ 1\ 1\ 10\ 1\ 1\ 1\ 1\ 10\ 1\ 1\ 1\ 1\ 10\ 1\ 1\ 1\ 1\ 10\ 1\ 1\ 1\ 1\ 10]).$$

The weights in P are chosen by a trail-and-error procedure. The basic idea behind this procedure is that more weight should be put on the temperatures of the five vessels because temperatures have more significant effect on the overall control performance, and the Lyapunov-based controller $h(x)$ should be able to stabilize the closed-loop system asymptotically with continuous feedback and actuation.

¶ $diag(v)$ denotes a matrix with its diagonal elements being the elements of vector v and all the other elements being zeros.

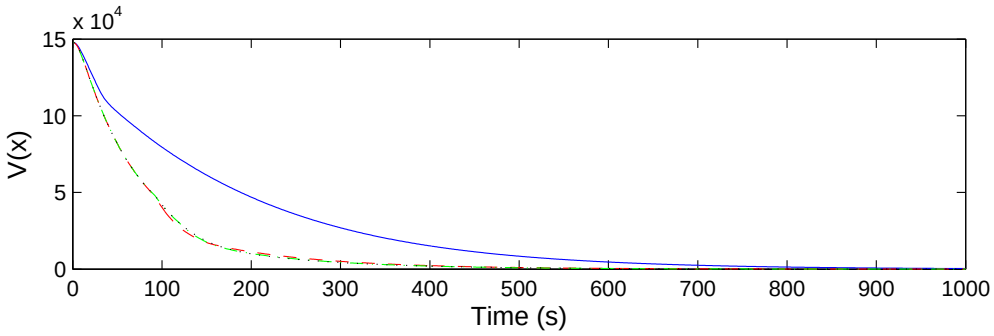


Figure 2.5: Trajectories of the Lyapunov function $V(x)$ of the alkylation of benzene process of Eqs. 2.23 under the controller $h(x)$ of Eq. 2.24 implemented in a sample-and-hold fashion (solid line), the centralized LMPC of Eqs. 2.5 (dashed line), the sequential DMPC of Eqs. 2.7 (dash-dotted line) and the iterative DMPC of Eqs. 2.20 with $c = 1$ (dotted line)

Based on $h(x)$, we design the centralized LMPC, the sequential distributed LMPC and the iterative distributed LMPC. The sampling time used is $\Delta = 30\ s$ and the weight matrices

$$Q_c = diag([1\ 1\ 1\ 1\ 10^3\ 1\ 1\ 1\ 1\ 10^3\ 10\ 10\ 10\ 10\ 10^4\ 1\ 1\ 1\ 1\ 10^3\ 1\ 1\ 1\ 1\ 10^3]).$$

and $R_{c1} = diag([10^{-8}\ 10^{-8}\ 10^{-8}])$, $R_{c2} = diag([10^{-8}\ 10^{-8}])$ and $R_{c3} = diag([1\ 1])$.

2.4.2 Simulation results

First, we carried out a set of simulations which demonstrate that the nonlinear control law $h(x)$ and the different schemes of LMPCs can all stabilize the closed-loop system asymptotically. Figure 2.5 shows the trajectories of the Lyapunov function $V(x)$ under the different control schemes. Note that because of the constraints of Eqs. 2.5e, 2.7g and 2.20f, the trajectories of the Lyapunov function of the closed-loop system under the centralized LMPC, the sequential DMPC and the iterative DMPC are guaranteed to be bounded by the corresponding Lyapunov function trajectory under

Table 2.7: Mean evaluation time of different LMPC optimization problems for 100 evaluations

		$N = 1$ (s)	$N = 3$ (s)	$N = 6$ (s)
Centralized LMPC		2.192	8.694	27.890
Sequential	LMPC 1	0.472	2.358	6.515
	LMPC 2	0.497	1.700	4.493
	LMPC 3	0.365	1.453	3.991
Iterative	LMPC 1	0.484	2.371	6.280
	LMPC 2	0.426	1.716	4.413
	LMPC 3	0.185	0.854	2.355

the controller $h(x)$ implemented in a sample-and-hold fashion with the sampling time Δ until $V(x)$ converges to a small region around the origin (i.e., $\Omega_{\rho_{\min}}$). This point is also illustrated in Figure 2.5.

Next, we compare the mean evaluation times of the centralized LMPC optimization problem and the sequential and iterative DMPC optimization problems. Each LMPC optimization problem was evaluated 100 times at different conditions. Different prediction horizons were considered in this set of simulations. The simulations were carried out using JAVA™ programming language in a PENTIUM® 3.20 GHz computer. The optimization problems were solved using the open source interior point optimizer Ipopt [101]. The results are shown in Table 2.7. From Table 2.7, we can see that in all cases, the time needed to solve the centralized LMPC is much larger than the time needed to solve the sequential or iterative DMPCs. This is because the centralized LMPC has to solve a much larger (in terms of decision variables) optimization problem than the DMPCs. We can also see that the evaluation time of

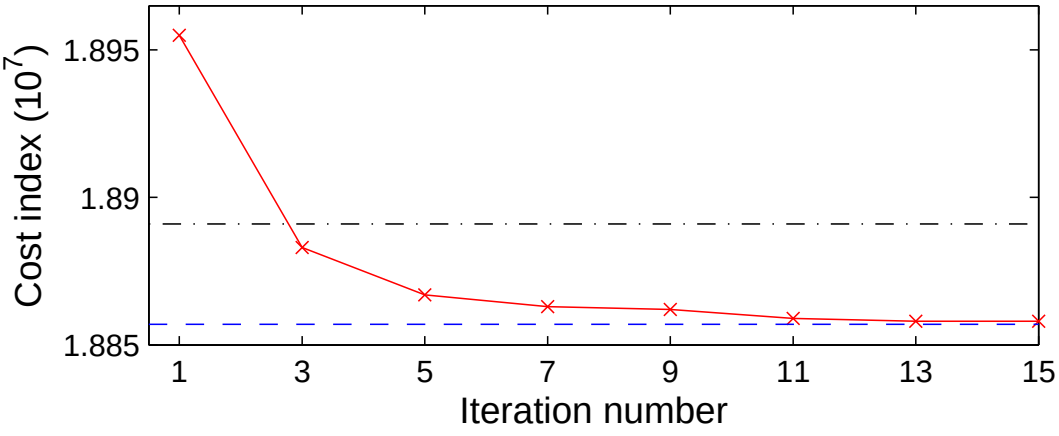


Figure 2.6: Total performance costs along the closed-loop trajectories of the alkylation of benzene process of Eqs. 2.23 under centralized LMPC of Eqs. 2.5 (dashed line), sequential DMPC of Eqs. 2.7 (dash-dotted line) and iterative DMPC of Eqs. 2.20 (solid line)

the centralized LMPC is even larger than the sum of evaluation times of LMPC 1, LMPC 2 and LMPC 3 in the sequential DMPC, and the times needed to solve the DMPCs in both sequential and iterative distributed schemes are of the same order of magnitude.

In the following set of simulations, we compare the centralized LMPC and the two DMPC schemes from a performance index point of view. In this set of simulations, the prediction horizon is $N = 1$. To carry out this comparison, the same initial condition and parameters were used for the different control schemes and the total cost under each control scheme was computed as follows:

$$J = \int_{t_0}^{t_M} \left[\|x(\tau)\|_{Q_c} + \|u_1(\tau)\|_{R_{c1}} + \|u_2(\tau)\|_{R_{c2}} + \|u_3(\tau)\|_{R_{c3}} \right] d\tau \tag{2.25}$$

where $t_0 = 0$ is the initial time of the simulations and $t_M = 1000\ s$ is the end of the simulations. Table 2.8 shows the total cost along the closed-loop system trajectories (trajectories I) under the different control schemes. For the iterative DMPC design,

Table 2.8: Total performance costs along the closed-loop trajectories I of the alkylation of benzene process of Eqs. 2.23

	J ($\times 10^7$)							
Centralized	1.8858							
Sequential	1.8891							
$c_{\max}$	1	3	5	7	9	11	13	15
Iterative	1.8955	1.8883	1.8867	1.8863	1.8862	1.8859	1.8858	1.8858

different maximum number of iterations, $c_{\max}$, are used. From Table 2.8, we can see that in this set of simulations, the centralized LMPC gives the lowest performance cost, the sequential DMPC gives lower cost than the iterative DMPC when there is no iteration ($c_{\max} = 1$). However, as the iteration number c increases, the performance cost given by the iterative DMPC decreases and converges to the cost of the one corresponding to the centralized LMPC. This point is also shown in Figure 2.6.

Note that the above set of simulations only represents one case of many possible cases. As we discussed in Remarks 2.11 and 2.12, there is no guaranteed convergence of the performance of distributed MPC to the performance of a centralized MPC and there is also no guaranteed superiority of the performance of one DMPC scheme over the others. In the following, we show two sets of simulations to illustrate these points. In both sets of simulations, we chose different matrices R_{c1} and R_{c2}, and all the other parameters (Q_c, R_{c3}, Δ, N) remained the same as the previous set of simulations. In the first set of simulations, we picked $R_{c1} = diag([5 \times 10^{-5}\ 5 \times 10^{-5}\ 5 \times 10^{-5}])$, $R_{c2} = diag([5 \times 10^{-5}\ 5 \times 10^{-5}])$. The total performance cost along the closed-loop system trajectories (trajectories II) under this simulation setting are shown in Table 2.9.

Table 2.9: Total performance costs along the closed-loop trajectories II of the alkylation of benzene process of Eqs. 2.23

	J ($\times 10^7$)			
Centralized	5.052			
Sequential	7.039			
$c_{\max}$	1	3	5	6
Iterative	7.2286	7.2241	7.2240	7.2240

Table 2.10: Total performance costs along the closed-loop trajectories III of the alkylation of benzene process of Eqs. 2.23

	J ($\times 10^7$)		
Centralized	3.8564		
Sequential	3.6755		
$c_{\max}$	1	3	4
Iterative	3.6663	3.6639	3.6639

From Table 2.9, we can see that the centralized LMPC provides a much lower cost than both the sequential and iterative distributed LMPCs. We can also see that as the number of iterations increases, the iterative distributed LMPC converges to a value which is different from the one obtained by the centralized LMPC. In the second set of simulations, we picked $R_{c1} = diag([1 \times 10^{-4} \ 1 \times 10^{-4} \ 1 \times 10^{-4}])$, $R_{c2} = diag([1 \times 10^{-4} \ 1 \times 10^{-4}])$ and the total performance cost along the closed-loop system trajectories (trajectories III) are shown in Table 2.10 from which we can see that the centralized LMPC provides a higher cost than both distributed LMPCs.

2.5 Conclusions

In this chapter, we presented two different architectures of distributed MPC for nonlinear process systems: sequential distributed LMPC and iterative distributed LMPC. In both architectures, the MPC controllers were designed via LMPC techniques. In the sequential distributed LMPC architecture, the distributed LMPC controllers adopt a one-directional communication strategy and are evaluated in sequence and once at each sampling time; in the iterative distributed LMPC architecture, the distributed LMPC controllers utilize a bi-directional communication strategy, are evaluated in parallel and iterate to improve closed-loop performance. Each LMPC controller in both architectures incorporates a suitable stability constraint which ensures that the state of the closed-loop system under the proposed distributed MPC architectures is ultimately bounded in an invariant set. Extensive simulations using a catalytic alkylation of benzene process example were carried out to compare the proposed distributed MPC architectures with existing centralized LMPC algorithms from computational time and closed-loop performance points of view.

Chapter 3

Sequential and Iterative Distributed Model Predictive Control of Nonlinear Systems: Handling Asynchronous and Delayed Measurements

3.1 Introduction

The development of DMPC schemes is of particular interest for the process industries because of the possibility of augmenting the sensor and actuation capabilities of control systems using hybrid communication networks that take advantage of an efficient integration of the existing, point-to-point communication networks (i.e., wired connections from each actuator or sensor to the control system using dedicated lo-

cal area networks) and additional wired or wireless networked actuator or sensor devices [108, 24, 83, 17]. However, the design of networked control systems has to account for the dynamics introduced by the communication network which may include time-varying delays or data losses in feedback. On the other hand, measuring difficulties of some system states, for example, species concentrations, in chemical processes also introduce the presence of asynchronous and delayed feedback in control loops. Previous work on MPC design for systems subject to asynchronous or delayed feedback has primarily focused on centralized MPC designs [45, 61, 80, 67] and little attention has been given to the design of DMPC for systems subject to asynchronous or delayed measurements. In a recent paper [34], the issue of delays in the communication between distributed controllers was addressed. In the work [66], sequential DMPC schemes for nonlinear systems subject to asynchronous and delayed state feedback were developed; however, the design of sequential DMPC involving two distributed controllers was only considered, and the design of sequential DMPC involving multiple controllers subject to asynchronous state feedback or iterative DMPC for nonlinear system subject to asynchronous and delayed state feedback was not addressed.

Motivated by the above considerations, in this chapter, we propose sequential and iterative DMPC schemes for large scale nonlinear systems subject to asynchronous and delayed state feedback [64, 14, 62]. In the case of asynchronous feedback, under the assumption that there is an upper bound on the maximum interval between two consecutive measurements, we first extend the results obtained in [66] for sequential DMPC to include multiple distributed controllers, and then re-design the iterative DMPC scheme presented in [63] to take explicitly into account asynchronous feedback. Subsequently, we design an iterative DMPC scheme for systems subject to asynchronous feedback that also involve time-delays under the assumption that there

exists an upper bound on the maximum feedback delay. This design takes advantage of the bi-directional communication network used in the iterative DMPC framework. Sufficient conditions under which the proposed distributed control designs guarantee that the states of the closed-loop system are ultimately bounded in regions that contain the origin are provided. The theoretical results are illustrated through a catalytic alkylation of benzene process example.

3.2 Preliminaries

3.2.1 Problem formulation

We consider nonlinear process systems described by the following state-space model:

$$\dot{x}(t) = f(x(t)) + \sum_{i=1}^{m} g_i(x(t))u_i(t) + k(x(t))w(t) \tag{3.1}$$

where $x(t) \in R^{n_x}$ denotes the vector of process state variables, $u_i(t) \in R^{m_{u_i}}$, $i = 1, \ldots, m$, are m sets of control (manipulated) inputs and $w(t) \in R^{n_w}$ denotes the vector of disturbance variables. The m sets of inputs are restricted to be in m nonempty convex sets $U_i \subseteq R^{m_{u_i}}$, $i = 1, \ldots, m$, which are defined as follows:

$$U_i := \{u_i \in R^{m_{u_i}} : |u_i| \leq u_i^{\max}\}^*, i = 1, \ldots, m$$

where $u_i^{\max}$, $i = 1, \ldots, m$, are the magnitudes of the input constraints. The disturbance vector is bounded, i.e., $w(t) \in W$ where

$$W := \{w \in R^{n_w} : |w| \leq \theta, \theta > 0\}.$$

*$|\cdot|$ denotes Euclidean norm of a vector.

We assume that $f(x)$, $g_i(x)$, $i = 1, \ldots, m$, and $k(x)$ are locally Lipschitz vector functions and that the origin is an equilibrium of the unforced nominal system (i.e., system of Eq. 3.1 with $u_i(t) = 0$, $i = 1, \ldots, m$, $w(t) = 0$ for all t) which implies that $f(0) = 0$.

3.2.2 Lyapunov-based controller

We assume that there exists a Lyapunov-based controller $h(x) = [h_1(x) \ \ldots \ h_m(x)]^T$ with $u_i = h_i(x)$, $i = 1, \ldots, m$, which renders (under continuous state feedback) the origin of the nominal closed-loop system asymptotically stable while satisfying the input constraints for all the states x inside a given stability region. We note that this assumption is essentially equivalent to the assumption that the process is stabilizable or that the pair (A, B) in the case of linear systems is stabilizable. Using converse Lyapunov theorems [60, 18], this assumption implies that there exist functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$ of class $\mathcal{K}$[†] and a continuously differentiable Lyapunov function $V(x)$ for the nominal closed-loop system, that satisfy the following inequalities:

$$\begin{gathered} \alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \\ \frac{\partial V(x)}{\partial x}\left(f(x) + \sum_{i=1}^{m} g_i(x) h_i(x)\right) \leq -\alpha_3(|x|) \\ \left|\frac{\partial V(x)}{\partial x}\right| \leq \alpha_4(|x|) \\ h_i(x) \in U_i, \ i = 1, \ldots, m \end{gathered} \tag{3.2}$$

for all $x \in O \subseteq R^{n_x}$ where O is an open neighborhood of the origin. We denote the region $\Omega_\rho \subseteq O$[‡] as the stability region of the closed-loop system under the Lyapunov-

[†]A continuous function $\alpha : [0, a) \to [0, \infty)$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and $\alpha(0) = 0$.

[‡]We use Ω_ρ to denote the set $\Omega_\rho := \{x \in R^{n_x} : V(x) \leq \rho\}$.

based controller $h(x)$. The construction of $V(x)$ can be carried out in a number of ways using systematic techniques like, for example, sum-of-squares methods.

By continuity, the local Lipschitz property assumed for the vector fields $f(x)$, $g_i(x)$, $i = 1, \ldots, m$, and $k(x)$ and taking into account that the manipulated inputs u_i, $i = 1, \ldots, m$, and the disturbance w are bounded in convex sets, there exists a positive constant M such that

$$\left| f(x) + \sum_{i=1}^{m} g_i(x) u_i + k(x) w \right| \leq M \tag{3.3}$$

for all $x \in \Omega_\rho$, $u_i \in U_i$, $i = 1, \ldots, m$, and $w \in W$. In addition, by the continuous differentiable property of the Lyapunov function $V(x)$ and the Lipschitz property assumed for the vector field $f(x)$, there exist positive constants L_x, L_{u_i}, $i = 1, \ldots, m$, and L_w such that

$$\begin{gathered} \left| \frac{\partial V}{\partial x} f(x) - \frac{\partial V}{\partial x} f(x') \right| \leq L_x \left| x - x' \right| \\ \left| \frac{\partial V}{\partial x} g_i(x) - \frac{\partial V}{\partial x} g_i(x') \right| \leq L_{u_i} |x - x'|, \; i = 1, \ldots, m \\ \left| \frac{\partial V}{\partial x} k(x) \right| \leq L_w \end{gathered} \tag{3.4}$$

for all $x, x' \in \Omega_\rho$, $u_i \in U_i$, $i = 1, \ldots, m$, and $w \in W$.

3.3 DMPC with asynchronous measurements

3.3.1 Modeling of asynchronous measurement

We assume that the state of the system of Eq. 3.1, $x(t)$, is available asynchronously at time instants t_k where $\{t_{k \geq 0}\}$ is a random increasing sequence of times. The

distribution of $\{t_{k\geq 0}\}$ characterizes the time needed to obtain a new measurement. In general, if there exists the possibility of arbitrarily large periods of time in which a new measurement is not available, then it is not possible to provide guaranteed stability properties. In order to study the stability properties in a deterministic framework, in the present work, we assume that there exists an upper bound T_m on the interval between two successive measurements, i.e., $\max_k\{t_{k+1} - t_k\} \leq T_m$. This assumption is reasonable from process control and networked control systems perspectives [102, 103]. This model of asynchronous measurements is of relevance to systems subject to asynchronous measurement samplings and to networked control systems, where the asynchronous property is introduced by data losses in the communication network connecting the sensors/actuators and the controllers.

In this section, we design sequential and iterative DMPC schemes, taking into account asynchronous measurements explicitly in their designs, that provide deterministic closed-loop stability properties. In each proposed architecture, we will design m Lyapunov-based MPC (LMPC) controllers to compute u_i, $i = 1, \ldots, m$, and refer to the LMPC computing the input trajectories of u_i as LMPC i.

3.3.2 Sequential DMPC formulation

A schematic diagram of the proposed sequential DMPC design for a system subject to asynchronous measurements is shown in Fig. 3.1. We propose to take advantage of the MPC scheme when feedback is lost to update the control inputs based on a state prediction obtained by the model and to have the control actuators store and implement the last computed optimal input trajectories [80, 66]. Specifically, the proposed implementation strategy is as follows:

1. When a new measurement is available at t_k, all the LMPCs receive the state

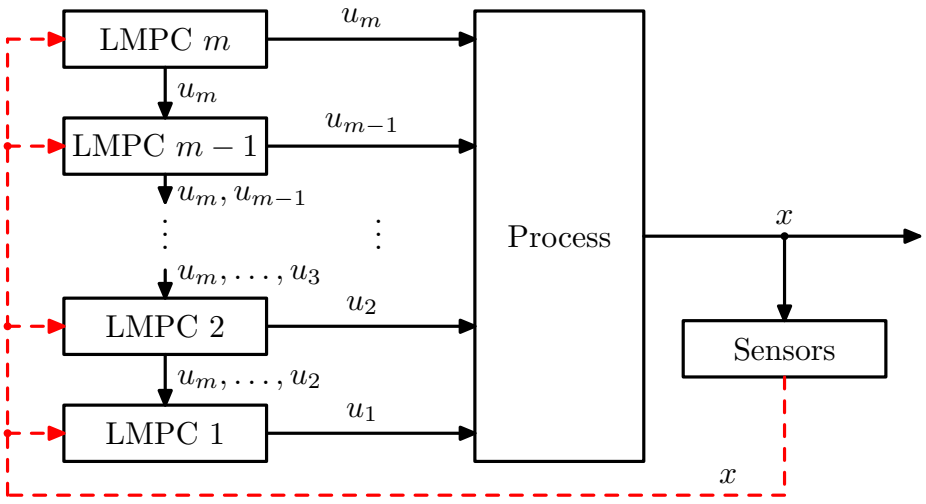


Figure 3.1: Sequential DMPC for nonlinear systems subject to asynchronous measurements.

measurement $x(t_k)$ from the sensors.

2. For $j = m$ to 1

   2.1. LMPC j receives the entire future input trajectories of u_i, $i = m, \ldots, j+1$, from LMPC $j+1$ and evaluates the future input trajectory of u_j based on $x(t_k)$ and the received future input trajectories.

   2.2. LMPC j sends the entire input trajectories of u_j to its actuators and the entire input trajectories of u_i, $i = m, \ldots, j$, to LMPC $j-1$.

Note that in the above implementation strategy, each LMPC sends its own computed input trajectories and the other input trajectories it received to the next LMPC controller (i.e., LMPC j sends input trajectories to LMPC $j-1$). This implies that LMPC j, $j = m, \ldots, 2$, does not have any information about the values that u_i, $i = j-1, \ldots, 1$ will take when the optimization problems of the LMPC controllers are evaluated. In order to make a decision, LMPC j, $j = m, \ldots, 2$ must assume trajectories for u_i, $i = j-1, \ldots, 1$, along the prediction horizon. To this end, the

Lyapunov-based controller $h(x)$ is used. In order to inherit the stability properties of the controller $h(x)$, each control input u_i, $i = 1, \ldots, m$ must satisfy a set of constraints that guarantee a given minimum contribution to the decrease rate of the Lyapunov function $V(x)$ in the case of asynchronous measurements.

In order to proceed, we define $\hat{x}(\tau|t_k)$ for $\tau \in [0, N\Delta]$ as the nominal sampled trajectory of the system of Eq. 3.1 associated with the feedback control law $h(x)$ and sampling time Δ starting from $x(t_k)$. This nominal sampled trajectory is obtained by integrating the following differential equation recursively:

$$\begin{aligned} \dot{\hat{x}}(\tau|t_k) = f(\hat{x}(\tau|t_k)) \quad + \quad & \sum_{i=1}^{m} g_i(\hat{x}(\tau|t_k)) h_i(\hat{x}(l\Delta|t_k)), \\ & \forall \tau \in [(l\Delta, (l+1)\Delta)),\ l = 0, \ldots, N-1. \end{aligned} \tag{3.5}$$

Based on $\hat{x}(\tau|t_k)$, we can define the following input trajectories:

$$u_{n,j}(\tau|t_k) \quad = \quad h_j(\hat{x}(l\Delta|t_k)),\ j = 1, \ldots, m,\ \forall \tau \in [l\Delta, (l+1)\Delta),\ l = 0, \ldots, N-1 \tag{3.6}$$

which will be used in the design of the LMPCs. Specifically, the design of LMPC j, $j = 1, \ldots, m$, is based on the following optimization problem:

$$\min_{u_{s,j} \in S(\Delta)} \int_0^{N\Delta} \left[\tilde{x}^j(\tau)^T Q_c \tilde{x}^j(\tau) + \textstyle\sum_{i=1}^{m} u_{s,i}(\tau)^T R_{ci} u_{s,i}(\tau) \right] d\tau \tag{3.7a}$$

$$\text{s.t. } \dot{\tilde{x}}^j(\tau) = f(\tilde{x}^j(\tau)) + \textstyle\sum_{i=1}^{m} g_i(\tilde{x}^j(\tau)) u_{s,i} \tag{3.7b}$$

$$\dot{\hat{x}}^j(\tau) = f(\hat{x}^j(\tau)) + \textstyle\sum_{i=1}^{j} g_i(\hat{x}^j(\tau)) u_{n,i}(\tau|t_k) + \sum_{i=j+1}^{m} g_i(\hat{x}^j(\tau)) u_{s,i} \tag{3.7c}$$

$$u_{s,i}(\tau) = u_{n,i}(\tau|t_k), i = 1, \ldots, j-1 \tag{3.7d}$$

$$u_{s,i}(\tau) = u^*_{s,i}(\tau|t_k), i = j+1, \ldots, m \tag{3.7e}$$

$$u_{s,j}(\tau) \in U_j \tag{3.7f}$$

$$\tilde{x}^j(0) = \hat{x}^j(0) = x(t_k) \tag{3.7g}$$

$$V(\tilde{x}^j(\tau)) \leq V(\hat{x}^j(\tau)),\ \forall \tau \in [0, N_R\Delta] \tag{3.7h}$$

where $S(\Delta)$ is the family of piece-wise constant functions with sampling time Δ, N is the prediction horizon, Q_c and R_{ci}, $i = 1, \ldots, m$, are positive definite weighting matrices, and N_R is the smallest integer satisfying $T_m \leq N_R\Delta$. The vector $\tilde{x}^j$ is the predicted trajectory of the nominal system with u_j computed by the above optimization problem (i.e., LMPC j) and the other control inputs defined by Eqs. 3.7d-3.7e. The vector $\hat{x}^j$ is the predicted trajectory of the nominal system with $u_j = u_{n,j}(\tau|t_k)$ and the other control inputs defined by Eqs. 3.7d-3.7e. In order to fully take advantage of the prediction, we choose $N \geq N_R$. The optimal solution to this optimization problem is denoted $u^*_{s,j}(\tau|t_k)$ and is defined for $\tau \in [0, N\Delta)$.

The constraint of Eq. 3.7b is the nominal model of the system, which is used to generate the trajectory $\tilde{x}^j$; the constraint of Eq. 3.7c defines a reference trajectory of the nominal system (i.e., $\hat{x}^j$) when the input u_j is defined by $u_{n,j}(\tau|t_k)$; the constraint of Eq. 3.7d defines the value of the inputs evaluated after u_j (i.e., u_i with $i = 1, \ldots, j-1$); the constraint of Eq. 3.7e defines the value of the inputs evaluated before u_j (i.e., u_i with $i = j+1, \ldots, m$); the constraint of Eq. 3.7f is the constraint on the manipulated input u_j; the constraint of Eq. 3.7g sets the initial state for the optimization problem; and the constraint of Eq. 3.7h guarantees that the contribution of input u_j to the decrease rate of the time derivative of the Lyapunov function from t_k to $t_k + N_R\Delta$, if $u_j = u^*_{s,j}(\tau|t_k)$, $\tau \in [0, N_R\Delta)$ is applied, is bigger or equal to the value obtained when $u_j = u_{n,j}(t - t_k|t_k)$, $t \in [t_k, t_k + N_R\Delta)$ is applied. This constraint guarantees that

the proposed sequential DMPC design maintains the stability of the Lyapunov-based controller $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation in the presence of asynchronous measurements.

The manipulated inputs of the closed-loop system under the above sequential DMPC are defined as follows:

$$u_i(t) = u^*_{s,i}(t - t_k|t_k),\ i = 1, \ldots, m, \forall t \in [t_k, t_{k+1}). \tag{3.8}$$

The proposed sequential DMPC design of Eqs. 3.7-3.8 maintains the closed-loop stability properties of the Lyapunov-based controller $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation in the presence of asynchronous measurements. This property is presented in Theorem 3.1 below. To state this theorem, we need the following propositions.

Proposition 3.1 (c.f. [80, 63]) *Consider the nominal sampled trajectory $\hat{x}$ of the system of Eq. 3.1 in closed-loop with the Lyapunov-based controller $h(x)$ applied in a sample-and-hold fashion and with open-loop state estimation. Let $\Delta, \epsilon_s > 0$ and $\rho > \rho_s > 0$ satisfy*

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + L^* M \leq -\epsilon_s/\Delta \tag{3.9}$$

with $L^* = L_x + \sum_{i=1}^{m} L_{u_i} u_i^{\max}$. Then, if $\rho_{\min} < \rho$ where

$$\rho_{\min} = \max\{V(\hat{x}(t + \Delta)) : V(\hat{x}(t)) \leq \rho_s\} \tag{3.10}$$

and $\hat{x}(0) \in \Omega_\rho$, the following inequality holds:

$$V(\hat{x}(k\Delta)) \leq \max\{V(\hat{x}(0)) - k\epsilon_s, \rho_{\min}\}. \tag{3.11}$$

Proposition 3.1 ensures that if the nominal system under the control $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation starts in Ω_ρ, then it is ultimately bounded in $\Omega_{\rho_{\min}}$. The following proposition provides an upper bound on the deviation of the state trajectory obtained using the nominal model, from the actual state trajectory when the same control actions are applied.

Proposition 3.2 (c.f. [69, 68]) *Consider the systems*

$$\begin{aligned} \dot{x}_a(t) &= f(x_a(t)) + \sum_{i=1}^{m} g_i(x_a(t))u_i(t) + k(x_a(t))w(t)) \\ \dot{x}_b(t) &= f(x_b(t)) + \sum_{i=1}^{m} g_i(x_b(t))u_i(t) \end{aligned} \tag{3.12}$$

with initial states $x_a(t_0) = x_b(t_0) \in \Omega_\rho$. There exists a class $\mathcal{K}$ function $f_W(\cdot)$ such that

$$|x_a(t) - x_b(t)| \leq f_W(t - t_0), \tag{3.13}$$

for all $x_a(t), x_b(t) \in \Omega_\rho$ and all $w(t) \in W$ with $f_W(\tau) = R_w\theta(e^{R_x\tau} - 1)/R_x$ and R_w, R_x are positive numbers.

Proposition 3.3 bounds the difference between the magnitudes of the Lyapunov function of two states in Ω_ρ.

Proposition 3.3 (c.f. [69, 68]) *Consider the Lyapunov function $V(\cdot)$ of the system of Eq. 3.1. There exists a quadratic function $f_V(\cdot)$ such that*

$$V(x) \leq V(\hat{x}) + f_V(|x - \hat{x}|) \tag{3.14}$$

for all $x, \hat{x} \in \Omega_\rho$ with $f_V(s) = \alpha_4(\alpha_1^{-1}(\rho))s + M_v s^2$ and $M_v > 0$.

In Theorem 3.1 below, we provide sufficient conditions under which the DMPC design of Eqs. 3.7-3.8 guarantees that the state of the closed-loop system is ultimately bounded in a region that contains the origin.

Theorem 3.1 *Consider the system of Eq. 3.1 in closed-loop with the DMPC design of Eqs. 3.7-3.8 based on the controller $h(x)$ that satisfies the conditions of Eq. 3.2 with class $\mathcal{K}$ functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$. Let $\Delta, \epsilon_s > 0$, $\rho > \rho_{\min} > 0$, $\rho > \rho_s > 0$ and $N \geq N_R \geq 1$ satisfy the conditions of Eqs. 3.9 and 3.10 and the following inequality:*

$$-N_R\epsilon_s + f_V(f_W(N_R\Delta)) < 0 \tag{3.15}$$

with N_R being the smallest integer satisfying $N_R\Delta \geq T_m$. If the initial state of the closed-loop system $x(t_0) \in \Omega_\rho$, then $x(t)$ is ultimately bounded in $\Omega_{\rho_a} \subseteq \Omega_\rho$ where

$$\rho_a = \rho_{\min} + f_V(f_W(N_R\Delta)).$$

Proof 3.1 In order to prove that the state of the closed-loop system is ultimately bounded in a region that contains the origin, we prove that $V(x(t_k))$ is a decreasing sequence of values with a lower bound. Specifically, we focus on the time interval $t \in [t_k, t_{k+1}]$ and prove that $V(x(t_{k+1}))$ is reduced compared with $V(x(t_k))$ or is maintained in an invariant set containing the origin.

To simplify the notation, we assume that all the signals used in this proof refer to the different optimization problems solved at t_k with the initial condition $x(t_k)$, and the trajectory $\tilde{x}^j(t)$, $j = 1, \ldots, m$, is corresponding to the optimal input $u^*_{s,j+1}(\tau|t_k)$. We also note that the predicted trajectories $\tilde{x}^{j+1}(t)$ and $\hat{x}^j(t)$ generated in the optimization problems of LMPC $j+1$ and LMPC j are identical. This property will be used in the proof.

Part 1: In this part, we prove that the stability results stated in Theorem 3.1 hold in the case that $t_{k+1} - t_k = T_m$ for all k and $T_m = N_R\Delta$. This case corresponds to the worst possible situation in the sense that the controllers need to operate in open-loop for the maximum possible amount of time. By Proposition 3.1 and the fact that $t_{k+1} = t_k + N_R\Delta$, the following inequality can be obtained:

$$V(\hat{x}(t_{k+1})) \leq \max\{V(\hat{x}(t_k)) - N_R\epsilon_s, \rho_{\min}\}. \tag{3.16}$$

From the constraints of Eq. 3.7h in the LMPCs, the following inequality can be written for all $t \in [t_k, t_k + N_R\Delta]$:

$$V(\tilde{x}^j(t)) \;\; \leq \;\; V(\hat{x}^j(t)),\; j = 1, \ldots, m. \tag{3.17}$$

By the fact that $\tilde{x}^{j+1}(t)$ and $\hat{x}^j(t)$ are identical, the following equations can be written for all $t \in [t_k, t_k + N_R\Delta]$:

$$V(\hat{x}^j(t)) \;\; = \;\; V(\tilde{x}^{j+1}(t)),\; j = 1, \ldots, m-1. \tag{3.18}$$

From the inequalities of Eqs. 3.17 and 3.18, the following inequalities are obtained for all $t \in [t_k, t_k + N_R\Delta]$:

$$V(\tilde{x}^1(t)) \leq \ldots \leq V(\tilde{x}^j(t)) \leq \ldots \leq V(\tilde{x}^m(t)) \leq V(\hat{x}^m(t)). \tag{3.19}$$

Note that the trajectory $\tilde{x}^1$ is the nominal trajectory (i.e., $\tilde{x}$) of the closed-loop system under the control of the sequential DMPC. Note also that the trajectory $\hat{x}^m$ is the nominal sampled trajectory (i.e., $\hat{x}$) of the closed-loop system defined in Eq. 3.5.

Therefore, the following trajectory can be written:

$$V(\tilde{x}(t)) \leq V(\hat{x}(t)), \forall\, t \in [t_k, t_k + N_R\Delta). \tag{3.20}$$

From the inequalities of Eq. 3.16 and 3.20 and the fact that $\hat{x}(t_k) = x(t_k)$, the following inequality is obtained:

$$V(\tilde{x}(t_{k+1})) \leq \max\{V(x(t_k)) - N_R\epsilon_s, \rho_{\min}\}. \tag{3.21}$$

When $x(t) \in \Omega_\rho$ for all times (this point will be proved below), we can apply Proposition 3.3 to obtain the following inequality:

$$V(x(t_{k+1})) \leq V(\tilde{x}(t_{k+1})) + f_V(|\tilde{x}(t_{k+1}) - x(t_{k+1})|). \tag{3.22}$$

Applying Proposition 3.2 we obtain the following upper bound on the deviation of $\tilde{x}(t)$ from $x(t)$:

$$|x(t_{k+1}) - \tilde{x}(t_{k+1})| \leq f_W(N_R\Delta). \tag{3.23}$$

From the inequalities of Eqs. 3.22 and 3.23, the following upper bound on $V(x(t_{k+1}))$ can be written:

$$V(x(t_{k+1})) \leq V(\tilde{x}(t_{k+1})) + f_V(f_W(N_R\Delta)). \tag{3.24}$$

Using the inequality of Eq. 3.21, we can re-write the inequality of Eq. 3.24 as follows:

$$V(x(t_{k+1})) \leq \max\{V(x(t_k)) - N_R\epsilon_s, \rho_{\min}\} + f_V(f_W(N_R\Delta)). \tag{3.25}$$

If the condition of Eq. 3.15 is satisfied, from the inequality of Eq. 3.25, we know that

there exists $\epsilon_w > 0$ such that the following inequality holds

$$V(x(t_{k+1})) \leq \max\{V(x(t_k)) - \epsilon_w, \rho_a\} \tag{3.26}$$

which implies that if $x(t_k) \in \Omega_\rho / \Omega_{\rho_a}$, then $V(x(t_{k+1})) < V(x(t_k))$, and if $x(t_k) \in \Omega_{\rho_a}$, then $V(x(t_{k+1})) \leq \rho_a$.

Because the upper bound on the difference between the Lyapunov function of the actual trajectory x and the nominal trajectory $\tilde{x}$ is a strictly increasing function of time (see Propositions 3.2 and 3.3 for the expressions of f_V and f_W), the inequality of Eq. 3.26 also implies that

$$V(x(t)) \leq \max\{V(x(t_k)), \rho_a\}, \ \forall t \in [t_k, t_{k+1}]. \tag{3.27}$$

Using the inequality of Eq. 3.27 recursively, it can be proved that if $x(t_0) \in \Omega_\rho$, then the closed-loop trajectories of the system of Eq. 3.1 under the proposed sequential DMPC design of Eqs. 3.7-3.8 stay in Ω_ρ for all times (i.e., $x(t) \in \Omega_\rho, \ \forall t$). Moreover, using the inequality of Eq. 3.27 recursively, it can be proved that if $x(t_0) \in \Omega_\rho$, the closed-loop trajectories of the system of Eq. 3.1 under the proposed sequential DMPC design satisfy

$$\limsup_{t \to \infty} V(x(t)) \leq \rho_a.$$

This proves that $x(t) \in \Omega_\rho$ for all times and $x(t)$ is ultimately bounded in Ω_{ρ_a} for the case when $t_{k+1} - t_k = T_m$ for all k and $T_m = N_R \Delta$.

Part 2: In this part, we extend the results proved in Part 1 to the general case, that is, $t_{k+1} - t_k \leq T_m$ for all k and $T_m \leq N_R \Delta$ which implies that $t_{k+1} - t_k \leq N_R \Delta$. Because f_V and f_W are strictly increasing functions of their arguments and f_V is convex, following similar steps as in Part 1, it can be shown that the inequality of

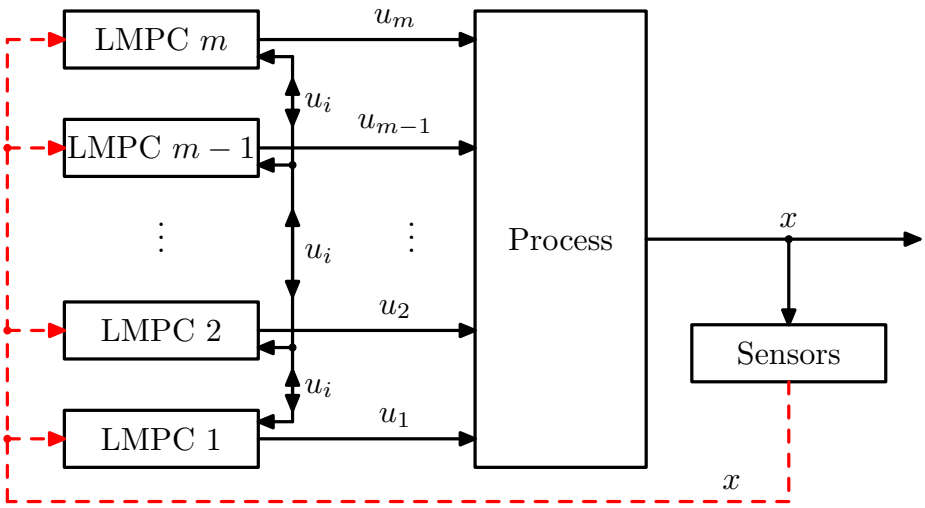


Figure 3.2: Iterative DMPC for nonlinear systems subject to asynchronous measurements.

Eq. 3.25 still holds. This proves that the stability results stated in Theorem 3.1 hold.

3.3.3 Iterative DMPC formulation

In contrast to the one-directional communication of the sequential DMPC architecture, the iterative DMPC architecture utilizes a bi-directional communication strategy in which all the distributed controllers are able to share their future input trajectories information after each iteration. In the presence of asynchronous measurements, the iterative DMPC in [63] cannot guarantee closed-loop stability. In this subsection, we modify the implementation strategy and the formulation of the distributed controllers to take into account asynchronous measurements. The proposed implementation strategy is as follows:

1. When a new measurement is available at t_k, all the LMPCs receive the state measurement $x(t_k)$ from the sensors.

2. At iteration c ($c \geq 1$):

2.1. All the distributed LMPCs exchange their latest future input trajectories.

2.2. Each LMPC evaluates its own future input trajectory based on $x(t_k)$ and the latest received input trajectories of all the other LMPCs.

3. If a termination condition is satisfied, each LMPC sends its entire future input trajectory to its actuators; if the termination condition is not satisfied, go to step 2 ($c \leftarrow c+1$).

For the iterations in this DMPC design, there are different choices of the termination condition. For example, the number of iterations c may be restricted to be smaller than a maximum iteration number $c_{\max}$ (i.e., $c \leq c_{\max}$) or the iterations may be terminated when the difference of the performance or the solution between two consecutive iterations is smaller than a threshold value or the iterations maybe terminated when a maximum computational time is reached.

The proposed design of the LMPC j, $j = 1, \ldots, m$, at iteration c is based on the following optimization problem:

$$\min_{u_{p,j} \in S(\Delta)} \int_0^{N\Delta} \left[\tilde{x}^j(\tau)^T Q_c \tilde{x}^j(\tau) + \sum_{i=1}^m u_{p,i}(\tau)^T R_{ci} u_{p,i}(\tau)\right] d\tau \tag{3.28a}$$

$$\text{s.t. } \dot{\tilde{x}}^j(\tau) = f(\tilde{x}^j(\tau)) + \sum_{i=1}^m g_i(\tilde{x}^j(\tau)) u_{p,i} \tag{3.28b}$$

$$u_{p,i}(\tau) = u_{p,i}^{*,c-1}(\tau | t_k),\ \forall i \neq j \tag{3.28c}$$

$$\left| u_{p,j}(\tau) - u_{p,j}^{*,c-1}(\tau | t_k) \right| \leq \Delta u_j,\ \forall \tau \in [0, N_R \Delta] \tag{3.28d}$$

$$u_{p,j}(\tau) \in U_j \tag{3.28e}$$

$$\tilde{x}^j(0) = x(t_k) \tag{3.28f}$$

$$\frac{\partial V(\tilde{x}^j)}{\partial \tilde{x}^j}\left(\frac{1}{m}f(\tilde{x}^j(\tau))+g_j(\tilde{x}^j(\tau))u_{p,j}(\tau)\right)\leq$$

$$\frac{\partial V(\hat{x})}{\partial \hat{x}}\left(\frac{1}{m}f(\hat{x}(\tau|t_k))+g_j(\hat{x}(\tau|t_k))u_{n,j}(\tau|t_k)\right), \forall \tau \in [0, N_R\Delta] \quad (3.28g)$$

where $\tilde{x}^j$ is the predicted trajectory of the nominal system with u_j computed by the LMPC of Eq. 3.28 and all the other inputs are the optimal input trajectories at iteration $c-1$ of the rest of the distributed controllers. The optimal solution to this optimization problem is denoted $u_{p,j}^{*,c}(\tau|t_k)$ which is defined for $\tau \in [0, N\Delta)$. Accordingly, we define the final optimal input trajectory of LMPC j (that is, the optimal trajectories computed at the last iteration) as $u_{p,j}^{*}(\tau|t_k)$ which is also defined for $\tau \in [0, N\Delta)$.

Note that for the first iteration of each distributed LMPC, the input trajectories defined in Eq. 3.6 based on the trajectory generated in Eq. 3.5 are used as the initial input trajectory guesses; that is, $u_{p,i}^{*,0} = u_{n,i}$ with $i = 1, \ldots, m$.

The constraint of Eq. 3.28d puts a limit on the input change in two consecutive iterations. This constraint allows LMPC j to take advantage of the input trajectories received in the last iteration (i.e., $u_{p,i}^{*,c-1}$, $\forall i \neq j$) to predict the future evolution of the system state without introducing big errors. For LMPC j (i.e., u_j), the magnitude of input change in two consecutive iterations is restricted to be smaller than a positive constant Δu_j. The constraint of Eq. 3.28g is used to guarantee the closed-loop stability.

The manipulated inputs of the closed-loop system under the above iterative DMPC are defined as follows:

$$u_i(t) = u_{p,i}^{*}(t - t_k|t_k),\ i = 1, \ldots, m, \forall t \in [t_k, t_{k+1}). \quad (3.29)$$

The iterative DMPC design of Eqs. 3.28-3.29 takes into account asynchronous measurements explicitly in the controller design and the implementation strategy. It maintains the closed-loop stability properties of the Lyapunov-based controller $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation. This property is presented in Theorem 3.2. To state this theorem, we need another proposition.

Proposition 3.4 *Consider the systems*

$$\begin{aligned} \dot{x}_a(t) &= f(x_a(t)) + \sum_{i=1}^{m} g_i(x_a(t))u_i^c(t) \\ \dot{x}_b(t) &= f(x_b(t)) + \sum_{i=1}^{m,\ i\neq j} g_i(x_b(t))u_i^{c-1}(t) + g_j(x_b(t))u_j^c(t) \end{aligned}$$

with initial states $x_a(t_0) = x_b(t_0) \in \Omega_\rho$. There exists a class $\mathcal{K}$ function $f_{X,j}(\cdot)$ such that

$$|x_a(t) - x_b(t)| \leq f_{X,j}(t - t_0) \tag{3.30}$$

for all $x_a(t)$, $x_b(t) \in \Omega_\rho$, and $u_i^c(t)$, $u_i^{c-1} \in U_i$ and $|u_i^c(t) - u_i^{c-1}(t)| \leq \Delta u_i$ with $i = 1, \ldots, m$.

Proof 3.2 Define the error vector as $e(t) = x_a(t) - x_b(t)$. The time derivative of the error is given by

$$\dot{e}(t) = f(x_a(t)) - f(x_b(t)) + \sum_{i=1}^{m,\ i\neq j} g_i(x_a(t))u_i^c(t) - \sum_{i=1}^{m,\ i\neq j} g_i(x_b(t))u_i^{c-1}(t).$$

Adding and subtracting $\sum_{i=1}^{m,\ i\neq j} g_i(x_b(t))u_i^c(t)$ to/from the right-hand-side of the

above equation, we obtain the following equation:

$$\begin{aligned}\dot{e}(t) &= f(x_a(t)) - f(x_b(t)) + \sum_{i=1}^{m,\ i\neq j} (g_i(x_a(t))u_i^c(t) - g_i(x_b(t))u_i^c(t)) \\ &\quad + \sum_{i=1}^{m,\ i\neq j} (g_i(x_b(t))u_i^c(t) - g_i(x_b(t))u_i^{c-1}(t)).\end{aligned}$$

By local Lipschitz properties assumed for the vector fields $g_i(\cdot)$, $i = 1, \ldots, m$, there exist positive constants $M_{g,i}$, $i = 1, \ldots, m$ such that $|g_i(x)| \leq M_{g,i}$, $i = 1, \ldots, m$. Moreover, by continuity and the local Lipschitz properties assumed for the vector field $f(\cdot)$, the fact that the manipulated inputs are bounded in convex sets and the difference between $u_i^c(t)$ and $u_i^{c-1}(t)$ is bounded, there exist positive constants C_x and $C_{g,i}$ $(i = 1, \ldots, m)$ that satisfy the following inequality:

$$\begin{aligned}|\dot{e}(t)| &\leq C_x\,|x_a(t) - x_b(t)| + \sum_{i=1}^{m,\ i\neq j} C_{g,i}\,|x_a(t) - x_b(t)|\,|u_i^c(t)| \\ &\quad + \sum_{i=1}^{m,\ i\neq j} |g_i(x_b(t))|\,\left|u_i^c(t) - u_i^{c-1}(t)\right| \\ &\leq C_x|e(t)| + \sum_{i=1}^{m,\ i\neq j} C_{g,i}u_i^{\max}\,|e(t)| + \sum_{i=1}^{m,\ i\neq j} M_{g,i}\Delta u_i.\end{aligned}$$

Denoting $C_{1,j} = C_x + \sum_{i=1}^{m,\ i\neq j} C_{g,i}u_i^{\max}$ and $C_{2,j} = \sum_{i=1}^{m,\ i\neq j} M_{g,i}\Delta u_i$, we can obtain

$$|\dot{e}(t)| \leq C_{1,j}|e(t)| + C_{2,j}.$$

Integrating $|\dot{e}(t)|$ with initial condition $e(t_0) = 0$ (recall that $x_a(t_0) = x_b(t_0)$), the following bound on the norm of the error vector is obtained:

$$|e(t)| \leq \frac{C_{2,j}}{C_{1,j}}(e^{C_{1,j}(t-t_0)} - 1).$$

This implies that Eq. 3.30 holds for

$$f_{X,j}(\tau) = \frac{C_{2,j}}{C_{1,j}}(e^{C_{1,j}\tau} - 1).$$

Proposition 3.4 bounds the difference between the nominal state trajectory under the optimized control inputs and the predicted nominal state trajectory generated in each LMPC optimization problem. To simplify the proof of Theorem 3.2, we define a new function $f_X(\tau)$ based on $f_{X,i}$, $i = 1, \ldots, m$, as follows:

$$f_X(\tau) = \sum_{i=1}^{m} \left(\frac{1}{m} L_x + L_{u_i} u_i^{\max} \right) \left(\frac{1}{C_{1,i}} f_{X,i}(\tau) - \frac{C_{2,i}}{C_{1,i}} \tau \right).$$

It is easy to verify that $f_X(\tau)$ is a strictly increasing and convex function of its argument.

In Theorem 3.2 below, we provide sufficient conditions under which the iterative DMPC guarantees that the state of the closed-loop system is ultimately bounded in a region that contains the origin.

Theorem 3.2 *Consider the system of Eq. 3.1 in closed-loop with the DMPC design of Eqs. 3.28-3.29 based on the controller $h(x)$ that satisfies the conditions of Eq. 3.2 with class $\mathcal{K}$ functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$. Let $\Delta, \epsilon_s > 0$, $\rho > \rho_{\min} > 0$, $\rho > \rho_s > 0$ and $N \geq N_R \geq 1$ satisfy the conditions of Eqs. 3.9 and 3.10 and the following inequality:*

$$-N_R \epsilon_s + f_X(N_R \Delta) + f_V(f_W(N_R \Delta)) < 0 \tag{3.31}$$

with N_R being the smallest integer satisfying $N_R \Delta \geq T_m$. If the initial state of the

closed-loop system $x(t_0) \in \Omega_\rho$, then $x(t)$ is ultimately bounded in $\Omega_{\rho_b} \subseteq \Omega_\rho$ where

$$\rho_b = \rho_{\min} + f_X(N_R\Delta) + f_V(f_W(N_R\Delta)).$$

Proof 3.3 We follow a similar strategy to the one in the proof of Theorem 3.1. In order to simplify the notation, we assume that all the signals used in this proof refer to the different optimization variables of the problems solved at t_k with the initial condition $x(t_k)$. This proof also includes two parts.

Part I: In this part, we prove that the stability results stated in Theorem 3.2 hold in the case that $t_{k+1} - t_k = T_m$ for all k and $T_m = N_R\Delta$. The derivative of the Lyapunov function of the nominal system of Eq. 3.1 under the control of the iterative DMPC of Eqs. 3.28-3.29 from t_k to t_{k+1} is expressed as follows:

$$\dot{V}(\tilde{x}(\tau)) = \frac{\partial V}{\partial x}\left(f(\tilde{x}(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}(\tau))u^*_{p,i}(\tau|t_k)\right),\ \forall \tau \in [0, N_R\Delta].$$

Adding the above inequality and the constraints of Eq. 3.28g in each LMPC together, we can obtain the following inequality:

$$\begin{aligned}\dot{V}(\tilde{x}(\tau)) \leq\ & \frac{\partial V}{\partial x}\left(f(\tilde{x}(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}(\tau))u^*_{p,i}(\tau|t_k)\right) \\ & + \frac{\partial V}{\partial x}\left(f(\hat{x}(\tau|t_k)) + \sum_{i=1}^{m} g_i(\hat{x}(\tau|t_k))u_{n,i}(\tau|t_k)\right) \\ & - \frac{\partial V}{\partial x}\left(\frac{1}{m}f(\tilde{x}^1(\tau)) + g_1(\tilde{x}^1(\tau))u^*_{p,1}(\tau|t_k)\right) - \dots \\ & - \frac{\partial V}{\partial x}\left(\frac{1}{m}f(\tilde{x}^m(\tau)) + g_m(\tilde{x}^m(\tau))u^*_{p,m}(\tau|t_k)\right),\ \forall \tau \in [0, N_R\Delta].\end{aligned} \tag{3.32}$$

Reworking the above inequality, the following inequality can be obtained for $\tau \in$

$[0, N_R\Delta]$:

$$\begin{aligned}\dot{V}(\tilde{x}(\tau)) \leq & \frac{\partial V}{\partial x}\left(f(\hat{x}(\tau|t_k)) + \sum_{i=1}^{m} g_i(\hat{x}(\tau|t_k))u_{n,i}(\tau|t_k)\right) \\ & + \frac{\partial V}{\partial x}\left(\frac{1}{m}f(\tilde{x}(\tau)) + g_1(\tilde{x})u^*_{p,1}(\tau|t_k)\right) \\ & - \frac{\partial V}{\partial x}\left(\frac{1}{m}f(\tilde{x}^1(\tau)) + g_1(\tilde{x}^1(\tau))u^*_{p,1}(\tau|t_k)\right) \\ & + \ldots \\ & + \frac{\partial V}{\partial x}\left(\frac{1}{m}f(\tilde{x}(\tau)) + g_m(\tilde{x})u^*_{p,m}(\tau|t_k)\right) \\ & - \frac{\partial V}{\partial x}\left(\frac{1}{m}f(\tilde{x}^m(\tau)) + g_m(\tilde{x}^m(\tau))u^*_{p,m}(\tau|t_k)\right).\end{aligned} \tag{3.33}$$

By the continuity and locally Lipschitz properties assumed for the vector fields $f(\cdot)$, $g_i(\cdot)$, $i = 1, \ldots, m$, and the constants defined in Eq. 3.3, the following inequality can be obtained for $\tau \in [0, N_R\Delta]$:

$$\begin{aligned}\dot{V}(\tilde{x}(\tau)) \quad \leq \quad & \dot{V}(\hat{x}(\tau|t_k)) + \left(\frac{1}{m}L_x + L_{u_1}u^*_{p,1}(\tau|t_k)\right)|\tilde{x}(\tau) - \tilde{x}^1(\tau)| + \ldots \\ & + \left(\frac{1}{m}L_x + L_{u_m}u^*_{p,m}(\tau|t_k)\right)|\tilde{x}(\tau) - \tilde{x}^m(\tau)|.\end{aligned} \tag{3.34}$$

Applying Proposition 3.4 to the above inequality of Eq. 3.34, we obtain the following inequality for $\tau \in [0, N_R\Delta]$:

$$\dot{V}(\tilde{x}(\tau)) \leq \dot{V}(\hat{x}(\tau|t_k)) + \left(\frac{1}{m}L_x + L_{u_1}u_1^{\max}\right)f_{X,1}(\tau) + \ldots + \left(\frac{1}{m}L_x + L_{u_m}u_m^{\max}\right)f_{X,m}(\tau). \tag{3.35}$$

Integrating the inequality of Eq. 3.35 from $\tau = 0$ to $\tau = N_R\Delta$ and taking into account

that $\tilde{x}(t_k) = \hat{x}(t_k)$, and $t_{k+1} - t_k = N_R\Delta$, the following inequality can be obtained:

$$\begin{aligned} V(\tilde{x}(t_{k+1})) \;\leq\; & V(\hat{x}(t_{k+1})) \\ & + \left(\frac{1}{m}L_x + L_{u_1}u_1^{\max}\right)\left(\frac{1}{C_{1,1}}f_{X,1}(N_R\Delta) - \frac{C_{2,1}}{C_{1,1}}N_R\Delta\right) + \ldots \\ & + \left(\frac{1}{m}L_x + L_{u_m}u_m^{\max}\right)\left(\frac{1}{C_{1,m}}f_{X,m}(N_R\Delta) - \frac{C_{2,m}}{C_{1,m}}N_R\Delta\right). \end{aligned} \tag{3.36}$$

From the definition of $f_X(\cdot)$, we have

$$V(\tilde{x}(t_{k+1})) \leq V(\hat{x}(t_{k+1})) + f_X(N_R\Delta). \tag{3.37}$$

By Propositions 3.1 and 3.3 and following similar calculations to the ones in the proof of Theorem 3.1, we obtain the following inequality

$$V(x(t_{k+1})) \leq \max\{V(x(t_k)) - N_R\epsilon_s, \rho_{\min}\} + f_X(N_R\Delta) + f_V(f_W(N_R\Delta)). \tag{3.38}$$

If the condition of Eq. 3.31 is satisfied, we know that there exists $\epsilon_w > 0$ such that the following inequality holds

$$V(x(t_{k+1})) \leq \max\{V(x(t_k)) - \epsilon_w, \rho_b\} \tag{3.39}$$

which implies that if $x(t_k) \in \Omega_\rho/\Omega_{\rho_b}$, then $V(x(t_{k+1})) < V(x(t_k))$, and if $x(t_k) \in \Omega_{\rho_b}$, then $V(x(t_{k+1})) \leq \rho_b$.

Because the upper bound on the difference between the Lyapunov function of the actual trajectory x and the nominal trajectory $\tilde{x}$ is a strictly increasing function of time, the inequality of Eq. 3.39 also implies that

$$V(x(t)) \leq \max\{V(x(t_k)) - \epsilon_w, \rho_b\}, \;\; \forall t \in [t_k, t_{k+1}]. \tag{3.40}$$

Using the inequality of Eq. 3.40 recursively, it can be proved that if $x(t_0) \in \Omega_\rho$, then the closed-loop trajectories of the system of Eq. 3.1 under the proposed iterative DMPC design stay in Ω_ρ for all times (i.e., $x(t) \in \Omega_\rho$ for all t). Moreover, if $x(t_0) \in \Omega_\rho$, the closed-loop trajectories of the system of Eq. 3.1 under the proposed iterative DMPC design satisfy

$$\limsup_{t\to\infty} V(x(t)) \leq \rho_b.$$

This proves that $x(t) \in \Omega_\rho$ for all times and $x(t)$ is ultimately bounded in Ω_{ρ_b} for the case when $t_{k+1} - t_k = T_m$ for all k and $T_m = N_R\Delta$.

Part 2: In this part, we extend the results proved in Part 1 to the general case, that is, $t_{k+1} - t_k \leq T_m$ for all k and $T_m \leq N_R\Delta$ which implies that $t_{k+1} - t_k \leq N_R\Delta$. Because f_V, f_W and f_X are strictly increasing functions of their arguments and f_X, f_V are convex, following similar steps as in Part 1, it can be shown that the inequality of Eq. 3.38 still holds. This proves that the stability results stated in Theorem 3.2 hold.

Remark 3.1 *Referring to the design of the LMPC of Eq. 3.28, the constraint of Eq. 3.28d ensures that the deviation of the predicted future state evolution (using input trajectories obtained in the last iteration) from the actual system state evolution is bounded. It also ensures that the results stated in Theorem 3.2 do not depend on the iteration number c which means the iteration of the DMPC can be terminated at any iteration and the stability properties stated in Theorem 3.2 continue to hold. The constraint of Eq. 3.28d can be also imposed as the termination condition of the iterative DMPC; that is, the DMPC stops iterating when* $|u_{p,i}(\tau) - u_{p,i}^{*,c-1}(\tau|t_k)| \leq \Delta u_i$, $i = 1, \ldots, m$, *for all* $\tau \in [0, N_R\Delta]$. *In this case, however, the stability properties stated in Theorem 3.2 have dependence on the iteration number c in a way that they hold only after the termination condition of Eq. 3.28d is satisfied.*

3.4 DMPC with delayed measurements

In this section, we consider the design of DMPC for systems subject to delayed measurements. In our previous work [66], we pointed out that in order to obtain a good estimate of the current system state from a delayed state measurement, a DMPC design should have bi-directional communication among the distributed controllers. Consequently, we focus on the design of DMPC for nonlinear systems subject to delayed measurements in an iterative DMPC framework.

3.4.1 Modeling of delayed measurements

We assume that the state of the system of Eq. 3.1 is received by the controllers at asynchronous time instants t_k where $\{t_{k\geq 0}\}$ is a random increasing sequence of times and that there exists an upper bound T_m on the interval between two successive measurements. We also assume that there are delays in the measurements received by the controllers due to delays in the sampling process and data transmission. In order to model delays in measurements, another auxiliary variable d_k is introduced to indicate the delay corresponding to the measurement received at time t_k, that is, at time t_k, the measurement $x(t_k - d_k)$ is received. In general, if the sequence $\{d_{k\geq 0}\}$ is modeled using a random process, there exists the possibility of arbitrarily large delays. In this case, it is improper to use all the delayed measurements to estimate the current state and decide the control inputs, because when the delays are too large, they may introduce enough errors to destroy the stability of the closed-loop system. In order to study the stability properties in a deterministic framework, we assume that the delays associated with the measurements are smaller than an upper bound D, which is, in general, relevant to measurement sensors and data transmission networks. This model is of relevance to systems subject to delayed measurements and

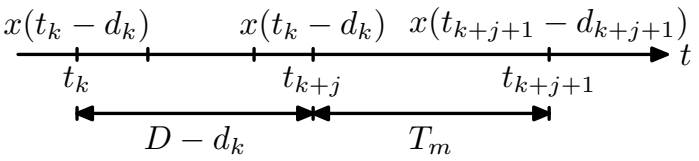


Figure 3.3: A possible sequence of delayed measurements.

to networked control systems, where the delay is introduced by the communication network connecting the sensors and the controllers.

Note that because the delays are time-varying, it is possible that at a time instant t_k, the controllers may receive a measurement $x(t_k - d_k)$ which does not provide new information (i.e., $t_k - d_k < t_{k-1} - d_{k-1}$); that is, the controller has already received a measurement of the state after time $t_k - d_k$. We assume that each measurement is time-labeled and hence the controllers are able to discard a newly received measurement if $t_k - d_k < t_{k-1} - d_{k-1}$. Figure 3.3 shows part of a possible sequence of $\{t_{k\geq 0}\}$. At time t_k, the state measurement $x(t_k - d_k)$ is received. There exists a possibility that between t_k and t_{k+j}, with $t_{k+j} - t_k = D - d_k$ and j being an unknown integer, all the measurements received do not provide new information. Note that any measurements received after t_{k+j} provide new information because the maximum delay is D and the latest received measurement is $x(t_k - d_k)$. The maximum possible time interval between t_{k+j} and t_{k+j+1} is T_m. Therefore, the maximum amount of time the system might operate in open-loop following t_k is $D + T_m - d_k$. This upper bound will be used in the formulation of the iterative DMPC design for systems subject to delayed measurements below.

3.4.2 Iterative DMPC formulation

As in the DMPC designs for systems subject to asynchronous measurements, we propose to take advantage of the system model both to estimate the current system

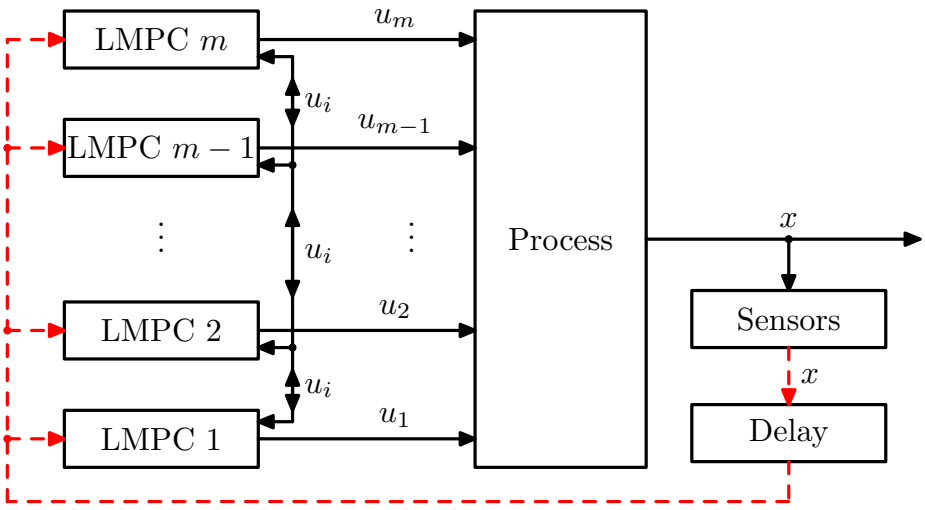


Figure 3.4: Iterative DMPC for nonlinear systems subject to delayed measurements.

state from a delayed measurement and to control the system in open-loop when new information is not available. To this end, when a delayed measurement is received, the distributed controllers use the system model and the input trajectories that have been applied to the system to get an estimate of the current state and then based on the estimate, MPC optimization problems are solved to compute the optimal future input trajectory that will be applied until new measurements are received. A schematic of the proposed iterative DMPC for systems subject to delayed measurements is shown in Fig. 3.4. The proposed implementation strategy for the proposed iterative DMPC design is as follows

1. When a measurement $x(t_k - d_k)$ is available at t_k, all the distributed controllers receive the state measurement and check whether the measurement provides new information. If $t_k - d_k > \max_{l<k} t_l - d_l$, go to step 2. Else the measurement does not contain new information and is discarded, go to step 3.

2. All the distributed controllers exchange their last implemented input trajectories and estimate the current state of the system $x^e(t_k)$.

3. At iteration c ($c \geq 1$):

   3.1. All the distributed controllers exchange their latest future input trajectories.

   3.2. Each controller evaluates its own future input trajectory based on $x^e(t_k)$ and the latest received input trajectories of all the other distributed controllers.

4. If a termination condition is satisfied, each distributed controller sends its entire future input trajectory to its actuators; if the termination condition is not satisfied, go to step 3 ($c \leftarrow c + 1$).

In order to estimate the current system state $x^e(t_k)$ based on a delayed measurement $x(t_k - d_k)$, the distributed controllers take advantage of the input trajectories that have been applied to the system from $t_k - d_k$ to t_k and the system model of Eq. 3.1. Let us denote the input trajectories that have been applied to the system as $u^*_{d,i}(t)$, $i = 1, \ldots, m$. Therefore, $x^e(t_k)$ is evaluated by integrating the following equation:

$$\dot{x}^e(t) = f(x^e(t)) + \sum_{i=1}^{m} g_i(x^e(t)) u^*_{d,i}(t), \ \forall t \in [t_k - d_k, t_k) \tag{3.41}$$

with $x^e(t_k - d_k) = x(t_k - d_k)$.

Before going to the design of the iterative DMPC, we need to define another nominal sampled trajectory $\check{x}(\tau|t_k)$ for $\tau \in [0, N\Delta]$, which is obtained by replacing $\hat{x}(\tau|t_k)$ with $\check{x}(\tau|t_k)$ in Eq. 3.5 and then integrating the equation with $\check{x}(t_k) = x^e(t_k)$. Based on $\check{x}(\tau|t_k)$, we define a new input trajectory as follows:

$$u^e_{n,j}(\tau|t_k) = h_j(\check{x}(l\Delta|t_k)), \ j = 1, \ldots, m, \ \forall \tau \in [l\Delta, (l+1)\Delta), \ l = 0, \ldots, N-1$$

which will be used in the design of the LMPC to construct the stability constraint and used as the initial input guess for iteration 1 (i.e., $u^{*,0}_{d,i} = u^e_{n,i}$ for $i = 1, \ldots, m$).

Specifically, the design of LMPC j, $j = 1, \ldots, m$, at iteration c is based on the following optimization problem:

$$\min_{u_{d,j} \in S(\Delta)} \int_0^{N\Delta} \left[\tilde{x}^j(\tau)^T Q_c \tilde{x}^j(\tau) + \textstyle\sum_{i=1}^m u_{d,i}(\tau)^T R_{ci} u_{d,i}(\tau)\right] d\tau \tag{3.42a}$$

$$\text{s.t. } \dot{\tilde{x}}^j(\tau) = f(\tilde{x}^j(\tau)) + \textstyle\sum_{i=1}^m g_i(\tilde{x}^j(\tau)) u_{d,i}(\tau) \tag{3.42b}$$

$$u_{d,i}(\tau) = u^{*,c-1}_{d,i}(\tau | t_k), \ \forall i \neq j \tag{3.42c}$$

$$\left| u_{d,j}(\tau) - u^{*,c-1}_{d,j}(\tau | t_k) \right| \leq \Delta u_j, \ \forall \tau \in [0, N_{Dk}\Delta] \tag{3.42d}$$

$$u_{d,j}(\tau) \in U_j \tag{3.42e}$$

$$\tilde{x}^j(0) = x^e(t_k) \tag{3.42f}$$

$$\frac{\partial V(\tilde{x}^j)}{\partial \tilde{x}^j} \left(\frac{1}{m} f(\tilde{x}^j(\tau)) + g_j(\tilde{x}^j(\tau)) u_{d,j}(\tau) \right) \leq$$

$$\frac{\partial V(\check{x})}{\partial \check{x}} \left(\frac{1}{m} f(\check{x}(\tau | t_k)) + g_j(\check{x}(\tau | t_k)) u^e_{n,j}(\tau | t_k) \right), \forall \tau \in [0, N_{Dk}\Delta] \tag{3.42g}$$

where N_{Dk} is the minimum integer satisfying $N_{Dk}\Delta \geq T_m + D - d_k$. The optimal solution to the optimization problem of Eq. 3.42 is denoted $u^{*,c}_{d,j}(\tau | t_k)$ which is defined for $\tau \in [0, N\Delta)$. Accordingly, we define the final optimal input trajectory of LMPC j as $u^*_{d,j}(\tau | t_k)$ which is also defined for $\tau \in [0, N\Delta)$. Note that the length of the constraint N_{Dk} depends on the current delay d_k, so it may have different values at different time instants and has to be updated before solving the optimization problems.

The manipulated inputs of the closed-loop system under the above iterative DMPC for systems subject to delayed measurements are defined as follows:

$$u_i(t) = u^*_{d,i}(t - t_k|t_k),\ i = 1, \ldots, m, \forall t \in [t_k, t_{k+q}) \tag{3.43}$$

for all t_k such that $t_k - d_k > \max_{l<k} t_l - d_l$ and for a given t_k, the variable q denotes the smallest integer that satisfies $t_{k+q} - d_{k+q} > t_k - d_k$. The stability properties of the iterative DMPC of Eqs. 3.42-3.43 are stated in the following theorem.

Theorem 3.3 *Consider the system of Eq. 3.1 in closed-loop with the DMPC design of Eqs. 3.42-3.43 based on the controller $h(x)$ that satisfies the conditions of Eq. 3.2 with class $\mathcal{K}$ functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$. Let $\Delta, \epsilon_s > 0$, $\rho > \rho_{\min} > 0$, $\rho > \rho_s > 0$, $N \geq 1$ and $D \leq 0$ satisfy the conditions of Eqs. 3.9 and 3.10 and the following inequality:*

$$-N_R\epsilon_s + f_X(N_D\Delta) + f_V(f_W(N_D\Delta)) + f_V(f_W(D)) < 0 \tag{3.44}$$

with N_D being the smallest integer satisfying $N_D\Delta \geq T_m + D$ and N_R being the smallest integer satisfying $N_R\Delta \geq T_m$. If the initial state of the closed-loop system $x(t_0) \in \Omega_\rho$, $N \geq N_D$ and $d_0 = 0$, then $x(t)$ is ultimately bounded in $\Omega_{\rho_d} \subseteq \Omega_\rho$ where

$$\rho_d = \rho_{\min} + f_X(N_D\Delta) + f_V(f_W(N_D\Delta)) + f_V(f_W(D)).$$

Proof 3.4 We assume that at t_k, a delayed measurement $x(t_k - d_k)$ containing new information is received, and that the next measurement with new state information is not received until t_{k+i}. This implies that $t_{k+i} - d_{k+i} > t_k - d_k$ and that the iterative DMPC of Eqs. 3.42-3.43 is solved at t_k and the optimal input trajectories $u^*_{d,i}(\tau|t_k)$,

$i = 1, \ldots, m$, are applied from t_k to t_{k+i}. In this proof, we will refer to $\tilde{x}(t)$ for $t \in [t_k, t_{k+i}]$ as the state trajectory of the nominal system of Eq. 3.1 under the control of the iterative DMPC of Eqs. 3.42-3.43 with $\tilde{x}(t_k) = x^e(t_k)$.

Part I: In this part, we prove that the stability results stated in Theorem 3.3 hold for $t_{k+i} - t_k = N_{Dk}\Delta$ and all $d_k \leq D$. By Proposition 3.1 and taking into account that $\check{x}(t_k) = x^e(t_k)$, the following inequality can be obtained:

$$V(\check{x}(t_{k+i})) \leq \max\{V(x^e(t_k)) - N_{Dk}\epsilon_s, \rho_{\min}\}. \tag{3.45}$$

By Proposition 3.2 and taking into account that $x^e(t_k - d_k) = x(t_k - d_k)$, $\tilde{x}(t_k) = x^e(t_k)$ and $N_D\Delta \geq N_{Dk}\Delta + d_k$, the following inequalities can be obtained:

$$\begin{aligned} |x^e(t_k) - x(t_k)| &\leq f_W(d_k) \\ |\tilde{x}(t_{k+i}) - x(t_{k+i})| &\leq f_W(N_D\Delta). \end{aligned} \tag{3.46}$$

When $x(t) \in \Omega_\rho$ for all times (this point will be proved below), we can apply Proposition 3.3 to obtain the following inequalities:

$$\begin{aligned} V(x^e(t_k)) &\leq V(x(t_k)) + f_V(f_W(d_k)) \\ V(x(t_{k+i})) &\leq V(\tilde{x}(t_{k+i})) + f_V(f_W(N_D\Delta)). \end{aligned} \tag{3.47}$$

From Eqs. 3.45 and 3.47, the following inequality is obtained:

$$V(\check{x}(t_{k+i})) \leq \max\{V(x(t_k)) - N_{Dk}\epsilon_s, \rho_{\min}\} + f_V(f_W(d_k)). \tag{3.48}$$

By Proposition 3.4 and following similar steps as in the proof of Theorem 3.2, the

following inequality can be obtained:

$$V(\tilde{x}(t_{k+i})) \leq V(\breve{x}(t_{k+i})) + f_X(N_{Dk}\Delta). \tag{3.49}$$

From Eqs. 3.47, 3.48 and 3.49, the following inequality is obtained:

$$V(x(t_{k+i})) \leq \max\{V(x(t_k)) - N_{Dk}\epsilon_s, \rho_{\min}\} + f_V(f_W(d_k)) + f_V(f_W(N_D\Delta)) + f_X(N_{Dk}\Delta). \tag{3.50}$$

In order to prove that the Lyapunov function is decreasing between two consecutive new measurements, the following inequality must hold

$$N_{Dk}\epsilon_s > f_V(f_W(d_k)) + f_V(f_W(N_D\Delta)) + f_X(N_{Dk}\Delta) \tag{3.51}$$

for all possible $0 \leq d_k \leq D$. Taking into account that f_W, f_V and f_X are strictly increasing functions of their arguments, N_{Dk} is a decreasing function of the delay d_k and that if $d_k = D$ then $N_{D_k} = N_R$, then if the condition of Eq. 3.44 is satisfied, the condition of Eq. 3.51 holds for all possible d_k and there exists $\epsilon_w > 0$ such that the following inequality holds

$$V(x(t_{k+i})) \leq \max\{V(x(t_k)) - \epsilon_w, \rho_d\} \tag{3.52}$$

which implies that if $x(t_k) \in \Omega_\rho / \Omega_{\rho_d}$, then $V(x(t_{k+i})) < V(x(t_k))$, and if $x(t_k) \in \Omega_{\rho_d}$, then $V(x(t_{k+i})) \leq \rho_d$.

Because the upper bound on the difference between the Lyapunov function of the actual trajectory x and the nominal trajectory $\tilde{x}$ is a strictly increasing function of

time, the inequality of Eq. 3.52 also implies that

$$V(x(t)) \leq \max\{V(x(t_k)), \rho_d\},\ \forall t \in [t_k, t_{k+i}]. \tag{3.53}$$

Using the inequality of Eq. 3.53 recursively, it can be proved that if $x(t_0) \in \Omega_\rho$, then the closed-loop trajectories of the system of Eq. 3.1 under the proposed iterative DMPC stay in Ω_ρ for all times (i.e., $x(t) \in \Omega_\rho,\ \forall t$). Moreover, using the inequality of Eq. 3.53 recursively, it can be proved that if $x(t_0) \in \Omega_\rho$, the closed-loop trajectories of the system of Eq. 3.1 under the proposed iterative DMPC satisfy

$$\limsup_{t \to \infty} V(x(t)) \leq \rho_d.$$

This proves that $x(t) \in \Omega_\rho$ for all times and $x(t)$ is ultimately bounded in Ω_{ρ_d} when $t_{k+i} - t_k = N_{Dk}\Delta$.

Part 2: In this part, we extend the results proved in Part 1 to the general case, that is, $t_{k+i} - t_k \leq N_{Dk}\Delta$. Taking into account that f_V, f_W and f_X are strictly increasing functions of their arguments and following similar steps as in Part 1, it can be readily proved that the inequality of Eq. 3.51 holds for all possible $d_k \leq D$ and $t_{k+i} - t_k \leq N_{Dk}\Delta$. Using this inequality and following the same line of argument as in the previous part, the stability results stated in Theorem 3.3 can be proved.

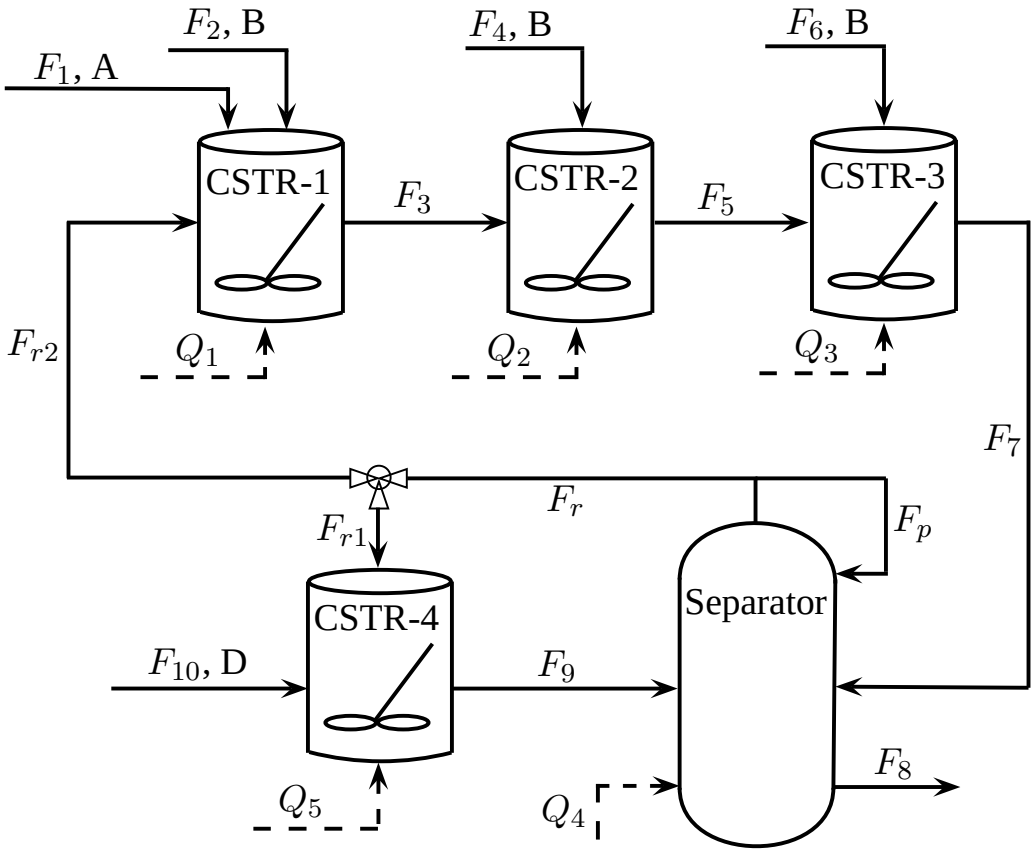


Figure 3.5: Process flow diagram of alkylation of benzene.

3.5 Application to an alkylation of benzene process

3.5.1 Process and control problem description

Consider the alkylation of benzene process of Eqs. 2.23 described in Section 2.4. The control objective is to drive the system from the initial condition as shown in Table 2.6 to the desired steady-state as shown in Table 2.4. The manipulated inputs are the heat injected to or removed from the five vessels, Q_1, Q_2, Q_3, Q_4 and Q_5, and the feed stream flow rates to CSTR-2 and CSTR-3, F_4 and F_6, whose steady-state input values are shown in Table 2.3. We design three distributed LMPCs to manipulate the 7 inputs. Similarly, the first distributed controller (LMPC 1) will be designed to decide the values of Q_1, Q_2 and Q_3, the second distributed controller (LMPC 2) will be designed to decide the values of Q_4 and Q_5, and the third distributed controller (LMPC 3) will be designed to decide the values of F_4 and F_6. The deviations of these inputs from their corresponding steady-state values are subject to the constraints shown in Table 2.5. We use the same design of $h(x)$ as in Section 2.4 with a quadratic Lyapunov function $V(x) = x^T P x$ with P being the following weight matrix:

$$P = diag\left([1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1\ 1]\right). \tag{3.54}$$

Based on $h(x)$, we design the sequential DMPC of Eqs. 3.7 and the iterative DMPC of Eqs. 3.28 with the following weighting matrices:

$$Q_c = diag\left(\left[1\ 1\ 1\ 1\ 10^3\ 1\ 1\ 1\ 1\ 10^3\ 10\ 10\ 10\ 10\ 3000\ 1\ 1\ 1\ 1\ 10^3\ 1\ 1\ 1\ 1\ 10^3\right]\right) \tag{3.55}$$

and $R_{c1} = diag\,([1 \times 10^{-8}\ 1 \times 10^{-8}\ 1 \times 10^{-8}])$, $R_{c2} = diag\,([1 \times 10^{-8}\ 1 \times 10^{-8}])$ and $R_{c3} = diag\,([10\ 10])$. The sampling time of the LMPCs is chosen to be $\Delta = 30$ *sec.* For the iterative DMPC of Eqs. 3.28, Δu_i is chosen to be $0.25u_i^{\max}$ for all the distributed LMPCs and maximum iteration numbers (i.e., $c \leq c_{\max}$) are applied as the termination conditions. In all the simulations, bounded process noise is added to the right hand side of the ordinary differential equations of the process model to simulate disturbances/model uncertainty.

3.5.2 Asynchronous measurements without delay

In this subsection, we consider that the state of the process of Eq. 2.23 is sampled asynchronously and that the maximum interval between two consecutive measurements is $T_m = 75$ *sec.* The asynchronous nature of the measurements is introduced by the measurement difficulties of the full state given the presence of several species concentration measurements. We will compare the proposed sequential and iterative DMPC for systems subject to asynchronous measurements with a centralized LMPC which takes into account asynchronous measurements explicitly [80]. The centralized LMPC uses the same weighting matrices, sampling time and prediction horizon as used in the DMPCs. To model the time sequence $\{t_{k\geq 0}\}$, we apply an upper bounded random Poisson process. The Poisson process is defined by the number of events per unit time W. The interval between two successive state sampling times is given by $\Delta_a = \min\{-ln\chi/W,\ T_m\}$, where χ is a random variable with uniform probability distribution between 0 and 1. This generation ensures that $\max_k\{t_{k+1} - t_k\} \leq T_m$. In the simulations, W is chosen to be 30 and the time sequence generated by this bounded Poisson process is shown in Fig. 3.6. For this set of simulations, we choose the prediction horizon of all the LMPCs to be $N = 3$ and choose $N_R = N$ so that

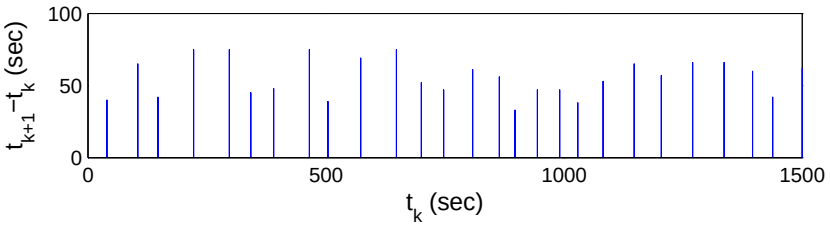
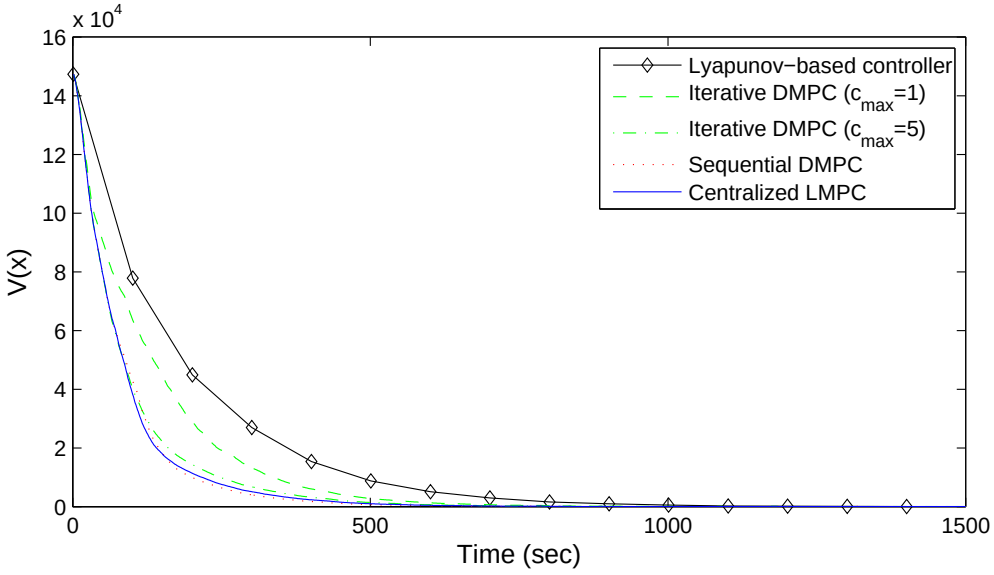


Figure 3.6: Asynchronous measurement sampling times $\{t_{k\geq 0}\}$ with $T_m = 75$ *sec*: the $x-$axis indicates $\{t_{k\geq 0}\}$ and the $y-$axis indicates the size of the interval between t_k and t_{k-1}.

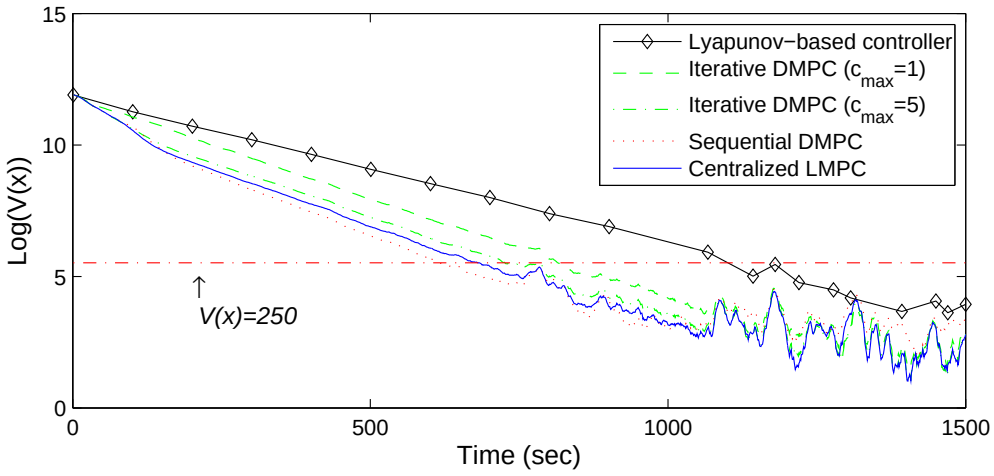
$N_R\Delta \geq T_m$.

We first compare the proposed DMPC designs for systems subject to asynchronous measurements with the centralized LMPC from a stability point of view. Figure 3.7 shows the trajectory of the Lyapunov function $V(x)$ under these control designs. From Fig. 3.7, we see that the proposed DMPC designs as well as the centralized LMPC design are able to drive the system state to a region very close to the desired steady state. From Fig. 3.7, we can also see that the sequential DMPC, the centralized LMPC and the iterative DMPC with $c_{\max} = 5$ give very similar trajectories of $V(x)$. Another important aspect we can see from Fig. 3.7(b) is that at the early stage of the closed-loop system simulation, because of the strong driving force related to the difference between the set-point and the initial condition, the process noise/disturbance has small influence on the process dynamics, even though the controller(s) has/have to operate in the presence of asynchronous measurements. When the states are getting close to the set-point, the Lyapunov function starts to fluctuate due to the domination of noise/disturbance over the vanishing driving force. However, the proposed DMPC designs are able to maintain practical stability of the closed-loop system and keep the trajectory of the Lyapunov function in a bounded region ($V(x) \leq 250$) very close to the steady state.

Next, we compare the evaluation times of the LMPCs in these control designs.



(a)



(b)

Figure 3.7: Trajectories of the Lyapunov function under the Lyapunov-based controller $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation, the iterative DMPC of Eqs. 3.28 with $c_{\max} = 1$ and $c_{\max} = 5$, the sequential DMPC of Eqs. 3.7 and the centralized LMPC in [80]: (a) $V(x)$; (b) $Log(V(x))$.

The simulations are carried out by Java programming language in a *Pentium 3.20 GHz* computer. The optimization problems are solved by the open source interior point optimizer Ipopt [101]. We evaluate the LMPC optimization problems for 100 runs. The mean evaluation time of the centralized LMPC, which optimizes all the inputs in one optimization problem, is about 23.7 *sec*. The mean evaluation time for the sequential DMPC architecture, which is the sum of the evaluation times (1.9 *sec*, 3.6 *sec* and 3.2 *sec*) of the three LMPCs, is about 8.7 *sec*. The mean evaluation time of the iterative DMPC architecture with one iteration is 6.3 *sec* which is the largest evaluation time among the evaluation times (1.6 *sec*, 6.3 *sec* and 4.3 *sec*) of the three LMPCs. The mean evaluation time of the iterative DMPC architecture with four iterations is 18.7 *sec* with the evaluation times of the three LMPCs being 6.9 *sec*, 18.7 *sec* and 14.0 *sec*. From this set of simulations, we can see that the proposed DMPC designs lead to a significant reduction in the controller evaluation time compared with a centralized LMPC design though they provide a very similar performance.

3.5.3 Asynchronous measurements subject to delays

In this subsection, we consider that the state of the process of Eq. 2.23 is sampled at asynchronous time instants $\{t_{k\geq 0}\}$ with an upper bound $T_m = 50$ *sec* on the interval between two successive measurements. Moreover, we consider that there are delays involved in the measurement samplings and the upper bound on the maximum delay is $D = 40$ *sec*. The delays in measurements can naturally arise in the context of species concentration measurements. We will compare the proposed iterative DMPC design of Eqs. 3.42-3.43 with a centralized LMPC which takes into account delayed measurements explicitly [67]. The centralized LMPC uses the same weighting matri-

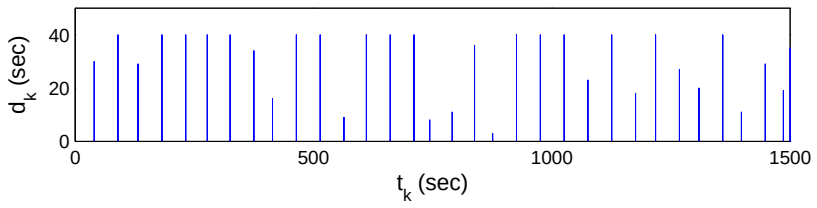
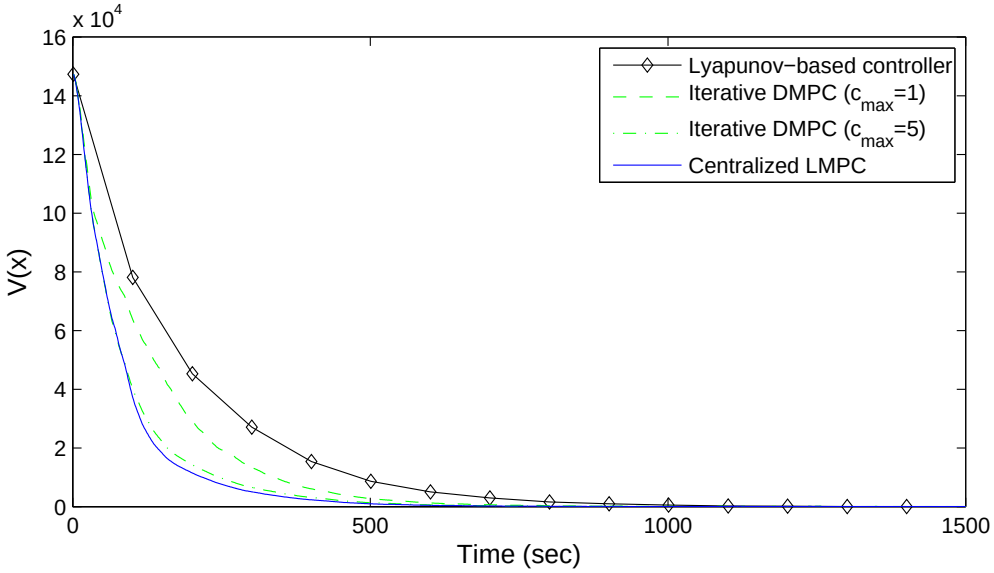


Figure 3.8: Asynchronous time sequence $\{t_{k\geq 0}\}$ and corresponding delay sequence $\{d_{k\geq 0}\}$ with $T_m = 50$ *sec* and $D = 40$ *sec*: the $x-$axis indicates $\{t_{k\geq 0}\}$ and the $y-$axis indicates the size of d_k.

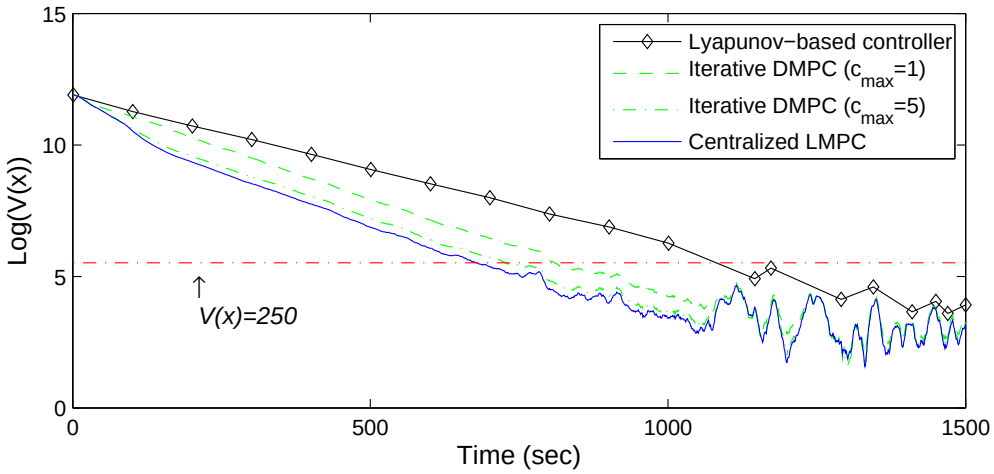
ces, sampling time and prediction horizon as used in the DMPCs. In order to model the sampling time instants, the same Poisson process is used to generate $\{t_{k\geq 0}\}$ with $W = 30$ and $T_m = 50$ *sec* and another random process is used to generate the associated delay sequence $\{d_{k\geq 0}\}$ with $D = 40$ *sec*. For this set of simulations, we also choose the prediction horizon of all the LMPCs to be $N = 3$ so that the horizon covers the maximum possible open-loop operation interval. Figure 3.8 shows the time instants when new state measurements are received and the associated delay sizes. Note that for all the control designs considered in this subsection, the same state estimation strategy shown in Eq. 3.41 is used.

Figure 3.9 shows the trajectory of the Lyapunov function $V(x)$ under different control designs. From Fig. 3.9, we see that both the proposed iterative DMPC for systems subject to delayed measurements and the centralized LMPC design in [67] are able to drive the system state to a region very close to the desired steady state ($V(x) \leq 250$); the trajectories of $V(x)$ generated by the iterative DMPC design are bounded by the corresponding trajectory of $V(x)$ under the Lyapunov-based controller $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation. From Fig. 3.9, we can also see that the centralized LMPC and the iterative DMPC with $c_{\max} = 5$ give very similar trajectories of $V(x)$.

In the final set of simulations, we compare the centralized LMPC and the two



(a)



(b)

Figure 3.9: Trajectories of the Lyapunov function under the Lyapunov-based controller $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation, the iterative DMPC of Eqs. 3.42-3.43 with $c_{\max} = 1$ and $c_{\max} = 5$ and the centralized LMPC in [67]: (a) $V(x)$; (b) $Log(V(x))$.

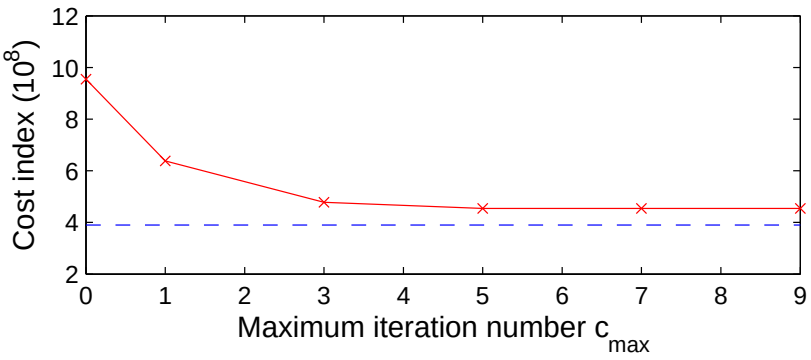


Figure 3.10: Total performance cost along the closed-loop system trajectories of centralized LMPC in [67] (dashed line) and iterative DMPC of Eqs. 3.42-3.43 (solid line).

distributed LMPC schemes from a performance index point of view. To carry out this comparison, the same initial condition and parameters were used for the different control schemes and the total cost under each control scheme was computed as follows:

$$J = \int_0^{t_f} (x(t)^T Q_c x(t) + u_1(t)^T R_{c1} u_1(t) + u_2(t)^T R_{c2} u_2(t) + u_3(t)^T R_{c3} u_3(t)) dt$$

where $t_f = 1500$ sec is the final simulation time. Figure 3.10 shows the total cost along the closed-loop system trajectories under the iterative DMPC of Eqs. 3.42-3.43 and the centralized LMPC in [67]. For the iterative DMPC design, different maximum numbers of iterations, $c_{\max}$, are used. From Fig. 3.10, we can see that as the iteration number c increases, the performance cost given by the iterative DMPC design decreases and converges to a value which is very close to the cost of the one corresponding to the centralized LMPC. However, we note that there is no guaranteed convergence of the performance of iterative DMPC design to the performance of a centralized MPC because of the non-convexity of the LMPC optimization problems, and the different stability constraints imposed in the centralized LMPC and the iterative DMPC design.

3.6 Conclusions

In this chapter, we designed sequential and iterative DMPC schemes for large scale nonlinear systems subject to asynchronous and delayed state feedback. First, we focused on nonlinear systems subject to asynchronous measurements without delays. In this case, we first extended our previous sequential DMPC design (Chapter 2) to include asynchronous measurements and then re-designed the iterative DMPC scheme proposed in Chapter 2 to take explicitly into account asynchronous feedback. Following that, we focused on the design of an iterative DMPC scheme for nonlinear systems subject to delayed measurements. This design taked advantage of the bi-directional communication network already used in the iterative DMPC framework. Mathematical analysis were carried out to derive sufficient conditions under which the proposed distributed control designs guarantee that the states of the closed-loop system are ultimately bounded in regions that contain the origin. Through a catalytic alkylation of benzene process example, we successfully compared the proposed DMPC designs with two centralized LMPC designs from stability, evaluation time and performance points of view.

Chapter 4

Distributed Economic MPC: Application to a Nonlinear Chemical Process Network

4.1 Introduction

The traditional, and currently dominant, approach to the achievement of economic optimization considerations of a plant relies on the use of a two-layer approach in which the upper layer is used to compute optimal process operation points taking into account economic considerations and using steady-state process models, and the lower-layer (i.e., process control layer) employs automatic feedback control systems to force the process to operate at the economically optimal steady-state computed by the upper layer. In the process control layer, classical control schemes wherever appropriate, as well as model predictive control (MPC) due to its ability to deal with multivariable constrained control problems and to account for optimization consid-

erations [36, 76], are widely used in industry. In the context of MPC, the standard approach is to use a quadratic cost function that involves penalties on the deviations of the state variables and of the control actions from their economically-optimal steady-state values over a finite prediction horizon. This consideration, together with appropriate stability constraints, allows standard MPC schemes to drive the state of the closed-loop system to the economically optimal steady-state for a suitable set of initial conditions. While this approach to enforcing economic considerations in the context of standard MPC formulations is widely used, there is room to improve upon the incorporation of economic considerations in the control layer and the computation of the control action.

To this end, there have been several authors within process control advocating the tighter integration of MPC and economic optimization of processes (e.g., [74, 2, 87, 47]). In [44], two economic MPC schemes were proposed for cyclic processes and nominal stability of the closed-loop system was established via Lyapunov techniques. MPC designs using an economics-based cost function were proposed in [26] and the closed-loop stability properties were established via a suitable Lyapunov function through adoption of a terminal constraint which requires that the closed-loop system state is driven to a steady-state at the end of the prediction horizon. Even though a rigorous stability analysis is included in [26], it is difficult, in general, to characterize, a priori, the set of initial conditions starting from where feasibility and closed-loop stability of the proposed MPC scheme are guaranteed. In contrast, in [40], a Lyapunov-based centralized economic MPC (LEMPC) scheme which has two different operation modes was proposed. The first operation mode corresponds to the period in which the cost function should be optimized while the second operation mode corresponds to operation in which the system is driven by the economic MPC to an appropriate ideally economically optimal steady-state. The design proposed in

[40] took advantage of the pre-defined Lyapunov-based controller to achieve feasibility and characterize the closed-loop stability region.

All of the above economic MPC designs are centralized in nature, that is the optimal manipulated input trajectories are computed from the solution of a single optimization problem. This approach is clearly effective in a number of applications but it may be limited in the context of large-scale nonlinear process networks that involve a large number of manipulated inputs. Distributed MPC (DMPC) has emerged as a feasible alternative to reduce the computational complexity of centralized MPC by solving multiple, reduced-order optimization problems in a parallel, iterative fashion; the reader may refer to ([10, 90, 92, 19]) for recent results in this area as well as to Chapter 2 and 3 of this thesis. However, all the existing DMPC methods utilize quadratic cost function that penalize the deviation of the states and inputs from their operating steady-state values, and generally, they do not explicitly account for economic objectives.

Motivated by the above, we focus on the development and application of distributed and centralized LEMPC designs to a catalytic alkylation of benzene process network [13], which consists of four continuously stirred tank reactors and a flash separator (See Section 2.4). A new economic measure for the entire process network is proposed which accounts for a broad set of economic considerations on the process operation including reaction conversion, separation quality and energy efficiency. Subsequently, steady-state process optimization is first carried out to locate an economically optimal (with respect to the proposed economic measure) operating steady-state. Then, a sequential distributed economic model predictive control design method, suitable for large-scale process networks, is proposed and its closed-loop stability properties are established. Using the proposed method, economic, distributed

as well as centralized, model predictive control systems are designed and are implemented on the process to drive the closed-loop system state close to the economically optimal steady-state. The closed-loop performance and time needed for control action calculation are evaluated through simulations and compared with the ones of a centralized Lyapunov-based model predictive control design, which uses a conventional, quadratic cost function that includes penalty on the deviation of the states and inputs from their economically optimal steady-state values.

4.2 Preliminaries

4.2.1 Notation

The notation $|\cdot|$ is used to denote the Euclidean norm of a vector, and a continuous function $\alpha : [0, a) \to [0, \overline{a})$ is said to belong to class $\mathcal{K}$ if it is strictly increasing and satisfies $\alpha(0) = 0$. The symbol Ω_r is used to denote the set $\Omega_r := \{x \in R^{n_x} : V(x) \leq r\}$ where V is a scalar continuous differentiable positive definite function, and the operator '/' denotes set subtraction, that is, $A/B := \{x \in R^{n_x} : x \in A, x \notin B\}$. The symbol $diag(v)$ denotes a matrix whose diagonal elements are the elements of vector v and all the other elements are zeros.

4.2.2 Class of nonlinear systems

We consider nonlinear systems described by the following state-space model:

$$\dot{x}(t) = f(x(t), u_1(t), \ldots, u_m(t), w(t)) \tag{4.1}$$

where $x(t) \in R^{n_x}$ denotes the vector of state variables of the system, $u_i(t) \in R^{m_{u_i}}$ $(i = 1, \ldots, m)$ and $w(t) \in R^{n_w}$ are the i^{th} set of control (manipulated) inputs and disturbances, respectively. The m sets of inputs are restricted to be in m nonempty convex sets $U_i \subseteq R^{m_{u_i}}$, $i = 1, \ldots, m$, which are defined as $U_i := \{u_i \in R^{m_{u_i}} : |u_i| \leq u_i^{\max}\}$ where $u_i^{\max}$, $i = 1, \ldots, m$, are the magnitudes of the input constraints. We will design m controllers to compute the m sets of control inputs u_i, $i = 1, \ldots, m$, respectively. We will refer to the controller computing u_i as controller i. $w(t)$ is assumed to be bounded, that is, $w(t) \in W$ with $W := \{w \in R^{n_w} : |w| \leq \theta, \theta > 0\}$. We assume that f is a locally Lipschitz vector function and that the origin is an equilibrium point of the unforced nominal system (i.e., system of Eq. 4.1 with $u_i(t) = 0$, $i = 1, \ldots, m$, $w(t) = 0$ for all t) which implies that $f(0, \ldots, 0) = 0$.

4.2.3 Stabilizability assumptions

We assume that there exists a feedback controller $h(x) = [h_1(x) \ \cdots \ h_m(x)]^T$ which renders the origin of the nominal closed-loop system asymptotically stable with $u_i = h_i(x)$, $i = 1, \ldots, m$, while satisfying the input constraints for all the states x inside a given stability region. Using converse Lyapunov theorems [60, 18], this assumption implies that there exist class $\mathcal{K}$ functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$ and a continuously differentiable Lyapunov function $V(x)$ for the nominal closed-loop system, that satisfy the following inequalities:

$$\begin{aligned} \alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \\ \frac{\partial V(x)}{\partial x} f(x, h_1(x), \ldots, h_m(x), 0) \leq -\alpha_3(|x|) \\ \left| \frac{\partial V(x)}{\partial x} \right| \leq \alpha_4(|x|), \ h_i(x) \in U_i, \ i = 1, \ldots, m \end{aligned} \tag{4.2}$$

for all $x \in O$. We denote the region $\Omega_\rho \subseteq O$ (Ω_ρ is a level set of $V(x)$) as the stability region of the closed-loop system under the Lyapunov-based controller $h(x)$. Note that explicit stabilizing control laws that provide explicitly defined regions of attraction for the closed-loop system have been developed using Lyapunov techniques for specific classes of nonlinear systems, particularly input-affine nonlinear systems; the reader may refer to [53, 18, 59] for results in this area including results on the design of bounded Lyapunov-based controllers by taking explicitly into account constraints for broad classes of nonlinear systems.

By continuity, the local Lipschitz property assumed for the vector field f and taking into account that the manipulated inputs u_i, $i = 1, \ldots, m$ are bounded, there exists a positive constant M such that:

$$|f(x, u_1, \ldots, u_m, w)| \leq M \tag{4.3}$$

for all $x \in \Omega_\rho$ and $u_i \in U_i$, $i = 1, \ldots, m$. By the continuous differentiable property of the Lyapunov function $V(x)$ and the Lipschitz property assumed for the vector field f, there exist positive constants L_x, L_w, L'_x and L'_w such that:

$$\begin{aligned} &|f(x, u_1, \ldots, u_m, w) - f(x', u_1, \ldots, u_m, 0)| \leq L_x |x - x'| + L_w |w| \\ &\left| \tfrac{\partial V(x)}{\partial x} f(x, u_1, \ldots, u_m, w) - \tfrac{\partial V(x')}{\partial x} f(x', u_1, \ldots, u_m, 0) \right| \leq L'_x |x - x'| + L'_w |w| \end{aligned} \tag{4.4}$$

for all $x, x' \in \Omega_\rho$, $u_i \in U_i$, $i = 1, \ldots, m$ and $w \in W$.

4.3 Nonlinear chemical process network

In this section, we consider the alkylation of benzene example as being described in Chapter 2.4. Subsequently, we introduce the economic cost measure.

4.3.1 Description of the alkylation of benzene process

The process of alkylation of benzene with ethylene to produce ethylbenzene is widely used in the petrochemical industry. Dehydration of the product produces styrene, which is the precursor to polystyrene and many copolymers. Over the last two decades, several modeling and simulation results of alkylation of benzene with catalysts have been reported in the literature. The process model developed in this section is based on the references [35, 58, 84, 105, 110]. More specifically, the process considered in this work consists of four continuously stirred tank reactors (CSTRs) and a flash tank separator, as shown in Fig. 4.1. The CSTR-1, CSTR-2 and CSTR-3 are in series and involve the alkylation of benzene with ethylene. Pure benzene is fed through stream F_1 and pure ethylene is fed through streams F_2, F_4 and F_6. Two catalytic reactions take place in CSTR-1, CSTR-2 and CSTR-3. Benzene (A) reacts with ethylene (B) and produces the required product ethylbenzene (C) (reaction 1); ethylbenzene can further react with ethylene to form 1,3-diethylbenzene (D) (reaction 2) which is the byproduct. The effluent of CSTR-3, including the products and leftover reactants, is fed to a flash tank separator, in which most of benzene is separated overhead by vaporization and condensation techniques and recycled back to the plant and the bottom product stream is removed. A portion of the recycle stream F_{r2} is fed back to CSTR-1 and another portion of the recycle stream F_{r1} is fed to CSTR-4 together with an additional feed stream F_{10} which contains 1,3-diethylbenzene from further distillation process that we do not consider in this example. In CSTR-4, both reaction 2 and the catalyzed transalkylation reaction in which 1,3-diethylbenzene reacts with benzene to produce ethylbenzene (reaction 3) take place. All chemicals left from CSTR-4 eventually pass into the separator. All the materials in the reactions are in liquid phase due to high pressure. The dynamic equations describing the be-

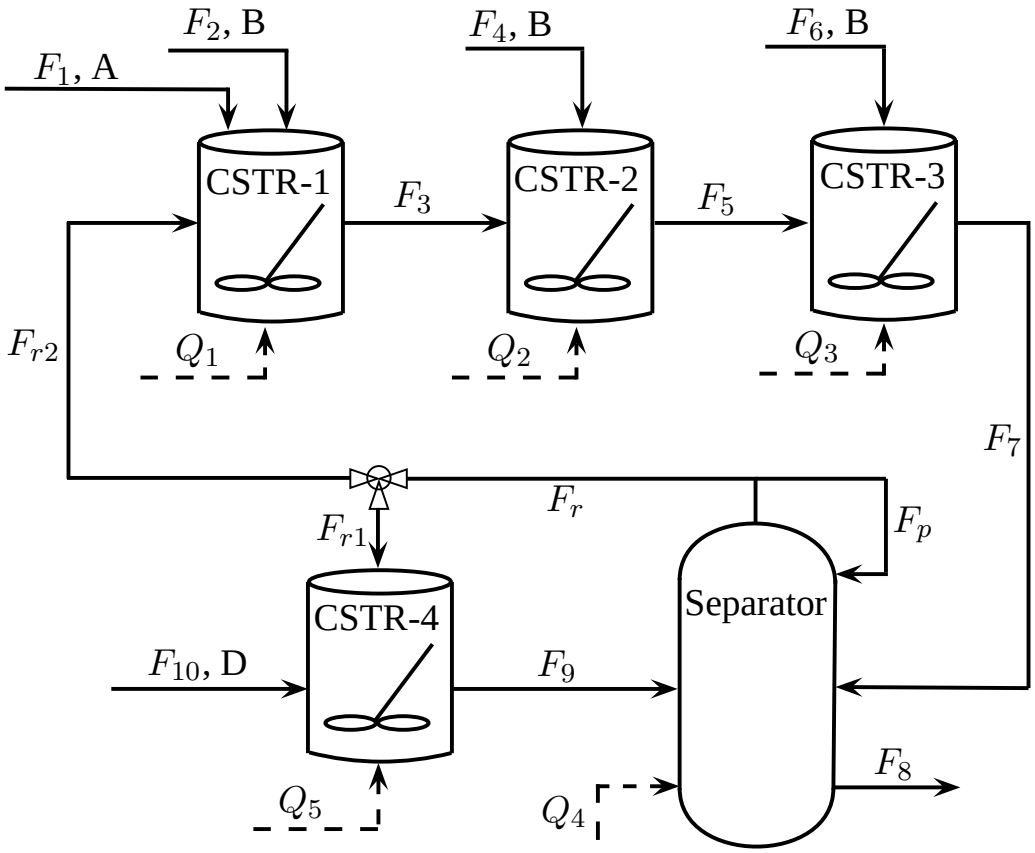


Figure 4.1: Process flow diagram of alkylation of benzene.

havior of the process, obtained through material and energy balances under standard modeling assumptions, can be found in Eq. 2.23. The process model consists of 25 coupled nonlinear ordinary differential equations.

Each of the tanks has an external heat/coolant input. The manipulated inputs to the process are the heat injected to or removed from the five vessels, Q_1, Q_2, Q_3, Q_4 and Q_5, and the feed streams F_2, F_4 and F_6 to CSTR-1, CSTR-2 and CSTR-3, respectively.

The states of the process consist of the concentrations of A, B, C, D in each of the five vessels and the temperatures of the vessels. The measurement of the process state is assumed to be available continuously to the controllers; i.e., state feedback

control is considered.

4.3.2 Economic cost measure

In this example, we consider the economic measure shown below accounting for three aspects: reaction conversion, separation quality, and energy efficiency:

$$\begin{aligned} L(x, u_1, ..., u_m) &= A_1 \frac{r_1}{r_2} + A_2 r_3 - A_4 Q_4 - A_5 Q_5 \\ &\quad + A_3 \frac{F_8 C_{C4}}{F_8 (C_{A4} + C_{B4} + C_{C4} + C_{D4})} \end{aligned} \tag{4.5}$$

where $L(x, u_1, ..., u_m)$ is the economic measure, x is the state of the system, $u_1, ..., u_m$ are the manipulated inputs with $U := [u_1 \ ... \ u_m] = [Q_1 \ Q_2 \ Q_3 \ Q_4 \ Q_5 F_2 \ F_4 \ F_6]$ and $A_1, \ldots, A_5$ are constant weighting coefficients, $r_1,\ r_2$ and r_3 are the reaction rates of reactions 1, 2 and 3, respectively, and C_{A4}, C_{B4}, C_{C4} and C_{D4} are concentrations of species A, B, C, D in the product outlet flow F_8. Note that the reaction rates are related to the concentrations of the reactants and the temperature in each reactor.

The first two terms of the measure describe the reaction conversion and the goal is to increase the rate of reactions one and three but suppress the rate of reaction two since it produces a by-product. The third and forth terms of the measure focus on energy efficiency. The fifth term of the measure takes the separation step into account, and the separation quality is measured in terms of the mole fraction of species C in the outlet stream F_8. We first solve a steady-state optimization problem using the economic measure of Eq. 4.5 as the cost function to be maximized to compute an economically optimal operating steady-state. The detailed formulation is shown

below:

$$\max_{x,u_1...u_m} L(x, u_1, ..., u_m) \tag{4.6a}$$

$$\text{s.t. } f(x, u_1, ..., u_m, 0) = 0 \tag{4.6b}$$

$$u_i \in U_i \tag{4.6c}$$

$$x \in X \tag{4.6d}$$

$$\sum_{i}^{2,4,6} F_i = F^{\max} \tag{4.6e}$$

$$V(x) \leq \tilde{\rho} \tag{4.6f}$$

where $L(x, u_1, ..., u_m)$ is the economic measure in Eq. 4.5, $f(x, u_1, ..., u_m, 0)$ is the nominal steady-state process model that described in [63], F^{max} is the maximum amount of reactant B that is allowed to enter the process per second, and V is the Lyapunov function and is taken to be $V(x) = (x - x^s)^T P(x - x^s)$ with P being a diagonal matrix of the form $diag([1\ 10\ 1\ 1\ 10^2\ 1\ 10\ 1\ 1\ 10^2\ 1\ 10\ 1\ 1\ 10^2\ 1\ 10\ 1\ 1\ 10^2\ 1\ 10\ 1\ 1\ 10^2])$. The optimal solution to this optimization problem is denoted as x^s and u_i^s, $i = 1, ..., m$. We note that the choice of P is done via trial-and-error and testing via simulations (there is no other way particularly given the type of nonlinearities involved); further, we note that the computation of $\dot{V}$ is done on the basis of the nonlinear vector fields of the process dynamic model and not on the basis of any type of linearization. We do scale the terms in P based on the different magnitude of the state variables. The value of $\tilde{\rho}$ is computed in a similar fashion and it is a function of V, process dynamic model and input constraints.

In the problem of Eq. 4.6, the constraint of Eq. 4.6b guarantees that the opti-

Table 4.1: Steady-state input values.

u_1	-4.4×10^6 $[J/s]$	u_2	-4.6×10^6 $[J/s]$
u_3	-4.7×10^6 $[J/s]$	u_4	9.2×10^6 $[J/s]$
u_5	5.9×10^6 $[J/s]$	u_6	8.697×10^{-4} $[m^3/s]$
u_7	8.697×10^{-4} $[m^3/s]$	u_8	8.697×10^{-4} $[m^3/s]$

mal solution satisfies the steady-state process model; the constraints of Eq. 4.6c and Eq. 4.6d define the state and input constraints; the constraint of Eq. 4.6e implies that the total amount of feed input B distributed through the stream F_2, F_4, and F_6 has to be equal to the maximum feed input F^{max}; and, the constraint of Eq. 4.6f imposes a Lyapunov constraint so that the solution has to lie inside the level set $\tilde{\rho}$.

We consider the system starting at $t = 0$ from a stable steady-state (x_0) that is defined by the steady-state inputs shown in Table 4.1. The state constraints that are imposed in this example require that the upper and lower bounds of the optimal temperature states are $\pm 6\%$ from the steady-state values of Table 4.1. The values of the rest of the state variables (concentrations) are required to be positive. The constrains that the manipulated inputs are subjected to are shown in Table 4.2. The values of F^{max} and $\tilde{\rho}$ are taken to be 26.091×10^{-4} mol/s and 2.4×10^6, respectively, and the coefficients A_1, A_2, A_3, A_4, A_5 are chosen to be 3, 1, 45, 27×10^{-7}, and 21×10^{-7}, respectively.

The steady-state optimization problem of Eq. 4.6 was solved by the open source interior point optimizer Ipopt under default settings in a JAVA programming environment. Simulation results indicate that there is only one optimal solution, and the optimal input values are given in Table 4.3. We also note that the optimal steady-state is unstable, determined by computing the Jacobian eigenvalues, and some tempera-

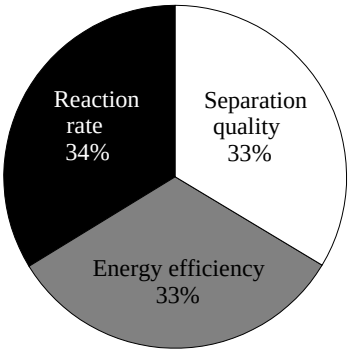


Figure 4.2: Weight percentage on the terms of the economic measure used by the steady-state optimization problem.

ture states of the final solution are at the boundary of the set X. The value of the Lyapunov function of the optimal solution of Eq. 4.6 is 1.45×10^6, which lies inside the set $\tilde{\rho}$. The optimal value of the economic measure is 33.41, which is a 5.7% increase from the initial steady-state of Table 4.1, and the weight placed on the various terms of the economic measure is shown in Fig. 4.2. We note that we nearly-equally weigh reaction conversion (first two terms), energy efficiency (third and fourth terms) and separation (fifth term). This was done because we consider, from a cost point of view, that these three terms equally contribute into the cost but if this is not the case, then the weights can be readily modified to accommodate different cost contributions of each one of these terms.

4.4 Distributed LEMPC

As the number of manipulated inputs increases as it is the case in the context of control of large-scale chemical plants, the evaluation time of a centralized MPC may increase

Table 4.2: Manipulated input constraints.

$\lvert u_1^s - u_1\rvert \leq 4.0 \times 10^6\ [J/s]$	$\lvert u_5^s - u_5\rvert \leq 2.0 \times 10^6\ [J/s]$
$\lvert u_2^s - u_2\rvert \leq 4.0 \times 10^6\ [J/s]$	$\lvert u_6^s - u_6\rvert \leq 8.679 \times 10^{-4}\ [m^3/s]$
$\lvert u_3^s - u_3\rvert \leq 4.0 \times 10^6\ [J/s]$	$\lvert u_7^s - u_7\rvert \leq 8.679 \times 10^{-4}\ [m^3/s]$
$\lvert u_4^s - u_4\rvert \leq 4.0 \times 10^6\ [J/s]$	$\lvert u_8^s - u_8\rvert \leq 8.679 \times 10^{-4}\ [m^3/s]$

Table 4.3: Optimal steady-state input values.

u_1^s	-5.773×10^6 $[J/s]$	u_2^s	-4.281×10^6 $[J/s]$
u_3^s	-1.481×10^6 $[J/s]$	u_4^s	6.238×10^6 $[J/s]$
u_5^s	7.010×10^6 $[J/s]$	u_6^s	1.296×10^{-3} $[m^3/s]$
u_7^s	7.355×10^{-4} $[m^3/s]$	u_8^s	5.773×10^{-4} $[m^3/s]$

significantly. This may impede the ability of centralized MPC to carry out real-time calculations within the limits imposed by process dynamics and operating conditions. Moreover, a centralized control system for large-scale systems may be difficult to organize and maintain and is vulnerable to potential process faults. To overcome these issues, in this work, we propose to utilize a sequential distributed economic model predictive control (EMPC) architecture as shown in Fig. 4.3. In this architecture, each set of the m sets of control inputs is calculated using an LEMPC. The distributed controllers are connected using one-directional communication network, evaluated in sequence. We will refer to the controller computing u_i associated with subsystem i as LEMPC i. In this section, we propose two different implementation strategies for the sequential distributed EMPC architecture and we assume that the state x of the system is sampled synchronously and the time instants at which we have state measurements are indicated by the time sequence $\{t_{k\geq 0}\}$ with $t_k = t_0 + k\Delta,\ k =$

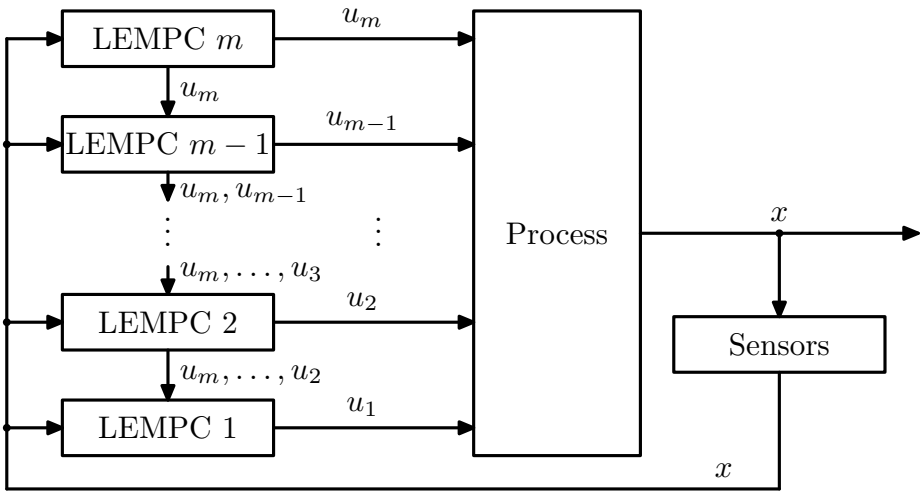


Figure 4.3: Distributed LEMPC architecture.

$0, 1, \ldots$ where t_0 is the initial time and Δ is the sampling time.

4.4.1 Implementation strategy I

In this implementation strategy for the distributed LEMPC architecture, all the distributed controllers are evaluated in sequence and once at each sampling time. Specifically, at a sampling time, t_k, when a measurement is received, the distributed controllers evaluate their future input trajectories in sequence starting from LEMPC m to LEMPC 1. Once a controller finishes evaluating its own future input trajectory, it sends its own future input trajectory and the future input trajectories it received to the next controller (i.e., LEMPC j sends input trajectories of u_i, $i = m, \ldots, j$, to LEMPC $j-1$).

This implementation strategy implies that LEMPC j, $j = m, \ldots, 2$, does not have any information about the values that u_i, $i = j-1, \ldots, 1$ will take when the optimization problem of LEMPC j is solved. In order to make a decision, LEMPC j, $j = m, \ldots, 2$ must assume trajectories for u_i, $i = j-1, \ldots, 1$, along the prediction horizon. To this end, the Lyapunov-based controller $h(x)$ is used. In order for the

distributed EMPC to inherit the stability properties of the controller $h(x)$, each control input u_i, $i = 1, \ldots, m$ must satisfy a constraint that guarantees a given minimum contribution to the decrease rate of the Lyapunov function $V(x)$. Specifically, the proposed design of the LEMPC j, $j = 1, \ldots, m$, is based on the following optimization problem:

$$\max_{u_j \in S(\Delta)} \int_{t_k}^{t_{k+N}} L(\tilde{x}^j(\tau), u_1(\tau), \ldots, u_m(\tau)) d\tau \tag{4.7a}$$

$$\text{s.t. } \dot{\tilde{x}}^j(t) = f(\tilde{x}^j(t), u_1(t), \ldots, u_m(t), 0) \tag{4.7b}$$

$$u_i(t) = h_i(\tilde{x}^j(t_{k+l})),\ i = 1, \ldots, j-1,$$

$$\forall t \in [t_{k+l}, t_{k+l+1}),\ l = 0, ..., N-1 \tag{4.7c}$$

$$u_i(t) = u_i^*(t|t_k), i = j+1, \ldots, m \tag{4.7d}$$

$$u_j(t) \in U_j,\ i = 1, \ldots, m \tag{4.7e}$$

$$\tilde{x}^j(t_k) = x(t_k) \tag{4.7f}$$

$$V(\tilde{x}^j(t)) \leq \tilde{\rho},\ \forall t \in [t_k, t_{k+N}),\ \text{if } t_k \leq t' \text{ and } V(x(t_k)) \leq \tilde{\rho} \tag{4.7g}$$

$$\frac{\partial V(x(t_k))}{\partial x} f(x(t_k), u_1^n(t_k), \ldots, u_{j-1}^n(t_k), u_j(t_k), \ldots, u_m(t_k))$$

$$\leq \frac{\partial V(x(t_k))}{\partial x} f(x(t_k), u_1^n(t_k), \ldots, u_j^n(t_k), u_{j+1}(t_k), \ldots, u_m(t_k)),$$

$$\text{if } t_k > t' \text{ or } \tilde{\rho} < V(x(t_k)) \leq \rho \tag{4.7h}$$

where $\tilde{x}^j$ is the predicted trajectory of the nominal system with u_i, $i = j+1, \ldots, m$, the input trajectory computed by the LEMPC controllers of Eq. 4.7 evaluated before

LEMPC j, u_i, $i = 1, \ldots, j-1$, the corresponding elements of $h(x)$ applied in a sample-and-hold fashion, $u_i^*(t|t_k)$ denotes the future input trajectory of u_i obtained by LEMPC i of the form of Eq. 4.7, and $u_i^n(t_k)$, $i = 1, \ldots, m$, are inputs determined by $h_i(x(t_k))$ (i.e., $u_i^n(t_k) = h_i(x(t_k))$). The optimal solution to the optimization problem of Eq. 4.7 is denoted $u_j^*(t|t_k)$ which is defined for $t \in [t_k, t_{k+N})$. The relation between $\tilde{\rho}$ and ρ is characterized in Theorem 4.1 below.

In the optimization problem of Eq. 4.7, the constraint of Eq. 4.7g is only active when $x(t_k) \in \Omega_{\tilde{\rho}}$ in the first operation mode and is incorporated to ensure that the predicted state evolution of the closed-loop system is maintained in the region $\Omega_{\tilde{\rho}}$ (thus, the actual state of the closed-loop system is in the stability region Ω_ρ; this point will be proved in Theorem 4.1 below). Due to the fact that all of the controllers receive state feedback $x(t_k)$ at sampling time t_k, all of the distributed controller operate in the same operation mode by verifying whether $V(x(t_k)) \leq \tilde{\rho}$; the constraint of Eq. 4.7h is only active in the second operation mode or when $\tilde{\rho} < V(x(t_k)) \leq \rho$ in the first operation mode. This constraint guarantees that the contribution of input u_j to the decrease rate of the time derivative of the Lyapunov function $V(x)$ at the initial time (i.e., t_k), if $u_j = u_j^*(t_k|t_k)$ is applied, is bigger than or equal to the value obtained when $u_j = h_j(x(t_k))$ is applied.

The manipulated inputs of the proposed distributed control design from time t_k to t_{k+1} ($k = 0, 1, 2, \ldots$) are applied in a receding horizon scheme as follows:

$$u_i(t) = u_i^*(t|t_k),\ i = 1, \ldots, m,\ \forall t \in [t_k, t_{k+1}). \tag{4.8}$$

To proceed for the closed-loop stability analysis, we need the following propositions.

Proposition 4.1 (c.f. [63]) *Consider the systems:*

$$\begin{aligned} \dot{x}_a(t) &= f(x_a(t), u_1(t), \ldots, u_m(t), w(t)) \\ \dot{x}_b(t) &= f(x_a(t), u_1(t), \ldots, u_m(t), 0) \end{aligned} \tag{4.9}$$

with initial states $x_a(t_0) = x_b(t_0) \in \Omega_\rho$. There exists a $\mathcal{K}$ function $f_W(\cdot)$ such that:

$$|x_a(t) - x_b(t)| \leq f_W(t - t_0), \tag{4.10}$$

for all $x_a(t), x_b(t) \in \Omega_\rho$ and all $w(t) \in W$ with $f_W(\tau) = L_w\theta(e^{L_x\tau} - 1)/L_x$.

Proposition 4.1 provides an upper bound on the deviation of the state trajectory obtained using the nominal model, from the actual system state trajectory when the same control input trajectories are applied. Proposition 4.2 below bounds the difference between the magnitudes of the Lyapunov function of two different states in Ω_ρ.

Proposition 4.2 (c.f. [63]) *Consider the Lyapunov function $V(\cdot)$ of the system of Eq. 4.1. There exists a quadratic function $f_V(\cdot)$ such that:*

$$V(x) \leq V(\hat{x}) + f_V(|x - \hat{x}|) \tag{4.11}$$

for all $x, \hat{x} \in \Omega_\rho$ with $f_V(s) = \alpha_4(\alpha_1^{-1}(\rho))s + M_v s^2$ where M_v is a positive constant.

Proposition 4.3 below ensures that if the nominal system controlled by $h(x)$ implemented in a sample-and-hold fashion and with open-loop state estimation starts in Ω_ρ, then it is ultimately bounded in $\Omega_{\rho_{\min}}$.

Proposition 4.3 (c.f. [63]) *Consider the nominal sampled trajectory $\hat{x}(t)$ of the system of Eq. 4.1 in closed-loop for a controller $h(x)$, which satisfies the condition of*

Eq. 4.2, obtained by solving recursively:

$$\dot{\hat{x}}(t) = f(\hat{x}(t), h_1(\hat{x}(t_k)), \ldots, h_m(\hat{x}(t_k)), 0) \tag{4.12}$$

where $t \in [t_k, t_{k+1})$ *with* $t_k = t_0 + k\Delta,\ k = 0, 1, \ldots.$ *Let* $\Delta, \epsilon_s > 0$ *and* $\rho > \rho_s > 0$ *satisfy:*

$$-\alpha_3\left(\alpha_2^{-1}(\rho_s)\right) + L'_x M\Delta \leq -\epsilon_s/\Delta. \tag{4.13}$$

Then, if $\hat{x}(t_0) \in \Omega_\rho$ *and* $\rho_{\min} < \rho$ *where* $\rho_{\min} = \max\{V(x(t+\Delta)) : V(x(t)) \leq \rho_s\}$, *the following inequality holds:* $V(\hat{x}(t)) \leq V(\hat{x}(t_k)),\ \forall t \in [t_k, t_{k+1})$ *and* $V(\hat{x}(t_k)) \leq \max\{V(\hat{x}(t_0)) - k\epsilon_s, \rho_{\min}\}$.

Theorem 4.1 below provides sufficient conditions under which the LEMPC of Eq. 4.7 guarantees that the state of the closed-loop system is always bounded in Ω_ρ and is ultimately bounded in a small region containing the origin.

Theorem 4.1 *Consider the system of Eq. 4.1 in closed-loop under the distributed LEMPC design of Eq. 4.7 based on a controller* $h(x)$ *that satisfies the conditions of Eq. 4.2. Let* $\epsilon_w > 0$, $\Delta > 0$, $\rho > \tilde{\rho} > 0$ *and* $\rho > \rho_s > 0$ *satisfy:*

$$\tilde{\rho} \leq \rho - f_V(f_W(\Delta)) \tag{4.14}$$

and

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + L'_x M\Delta + L'_w\theta \leq -\epsilon_w/\Delta. \tag{4.15}$$

If $x(t_0) \in \Omega_\rho$, $\rho_s \leq \tilde{\rho}$, $\rho_{\min} \leq \rho$ *and* $N \geq 1$, *then the state* $x(t)$ *of the closed-loop system is always bounded in* Ω_ρ *and is ultimately bounded in* $\Omega_{\rho_{\min}}$ *with* $\rho_{\min}$ *defined in Proposition 4.3 .*

Proof: The proof consists of three parts. We first prove that the optimization

problem of Eq. 4.7 is feasible for all states $x \in \Omega_\rho$. Subsequently, we prove that, in the first operation mode, under the LEMPC design of Eq. 4.7, the closed-loop state of the system of Eq. 4.1 is always bounded in Ω_ρ. Finally, we prove that, in the second operation mode, under the LEMPC of Eq. 4.7, the closed-loop state of the system of Eq. 4.1 is ultimately bounded in $\rho_{\min}$.

Part 1: When $x(t)$ is maintained in Ω_ρ (which will be proved in Part 2), the feasibility of the distributed EMPC (DEMPC) of Eq. 4.7 follows because input trajectory $u_j(t)$, $j = 1, \ldots, m$, such that $u_j(t) = h_j(x(t_{k+q}))$, $\forall t \in [t_{k+q}, t_{k+q+1})$ with $q = 0, \ldots, N-1$ is a feasible solution to the optimization problem of Eq. 4.7 since such trajectory satisfy the input constraint of Eq. 4.7e and the Lyapunov-based constraints of Eqs. 4.7g and 4.7h. This is guaranteed by the closed-loop stability property of the Lyapunov-based controller $h(x)$; the reader may refer to [80] for more detailed discussion on the stability property of the Lyapunov-based controller $h(x)$.

Part 2: We assume that the LEMPC of Eq. 4.7 operates in the first operation mode. We prove that if $x(t_k) \in \Omega_{\tilde{\rho}}$, then $x(t_{k+1}) \in \Omega_\rho$; and if $x(t_k) \in \Omega_\rho / \Omega_{\tilde{\rho}}$, then $V(x(t_{k+1})) < V(x(t_k))$ and in finite steps, the state converges to $\Omega_{\tilde{\rho}}$ (i.e., $x(t_{k+j}) \in \Omega_{\tilde{\rho}}$ where j is a finite positive integer).

When $x(t_k) \in \Omega_{\tilde{\rho}}$, from the constraint of Eq. 4.7g, we obtain that $\tilde{x}^1(t_{k+1}) \in \Omega_{\tilde{\rho}}$. By Propositions 4.1 and 4.2, we obtain the following inequality:

$$V(x(t_{k+1})) \leq V(\tilde{x}^1(t_{k+1})) + f_V(f_W(\Delta)). \tag{4.16}$$

Note that LEMPC 1 has access to all of the optimal input trajectories of the other distributed controllers evaluated before it. Since $V(\tilde{x}^1(t_{k+1})) \leq \tilde{\rho}$, if the condition of

Eq. 4.14 is satisfied, we can conclude that:

$$x(t_{k+1}) \in \Omega_\rho.$$

When $x(t_k) \in \Omega_\rho / \Omega_{\tilde{\rho}}$, from the constraint of Eq. 4.7h and the condition of Eq. 4.2, we can obtain:

$$\begin{aligned}
&\frac{\partial V(x(t_k))}{\partial x} f(x(t_k), u_1^*(t_k|t_k), \ldots, u_m^*(t_k|t_k), 0) \\
\leq\ &\frac{\partial V(x(t_k))}{\partial x} f(x(t_k), h_1(x(t_k)), u_2^*(t_k|t_k), \ldots, u_m^*(t_k|t_k), 0) \\
\leq\ &\ldots \\
\leq\ &\frac{\partial V(x(t_k))}{\partial x} f(x(t_k), h_1(x(t_k)), \ldots, h_m(x(t_k)), 0) \\
\leq\ &-\alpha_3(|x(t_k)|).
\end{aligned} \tag{4.17}$$

The time derivative of the Lyapunov function along the actual system state $x(t)$ for $t \in [t_k, t_{k+1})$ can be written as follows:

$$\dot{V}(x(t)) = \frac{\partial V(x(t))}{\partial x} f(x(t), u_1^*(t_k|t_k), \ldots, u_m^*(t_k|t_k), w(t)) \tag{4.18}$$

Adding and subtracting $\frac{\partial V(x(t_k))}{\partial x} f(x(t), u_1^*(t_k|t_k), \ldots,$
$u_m^*(t_k|t_k), 0)$ to/from the above equation and accounting for Eq. 4.17, we have:

$$\begin{aligned}
\dot{V}(x(t)) \ \leq\ & -\alpha_3(|x(t_k)|) + \frac{\partial V(x(t))}{\partial x} f(x(t), u_1^*(t_k|t_k), \ldots, u_m^*(t_k|t_k), w(t)) \\
& -\frac{\partial V(x(t_k))}{\partial x} f(x(t), u_1^*(t_k|t_k), \ldots, u_m^*(t_k|t_k), 0)
\end{aligned} \tag{4.19}$$

Due to the fact that the disturbance is bounded (i.e., $|w| \leq \theta$) and the Lipschitz

properties of Eq. 4.4, we can write:

$$\dot{V}(x(t)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + L'_x|x(t) - x(t_k)| + L_w\theta. \tag{4.20}$$

Taking into account Eq. 4.3 and the continuity of $x(t)$, the following bound can be written for all $\tau \in [t_k, t_{k+1})$

$$|x(\tau) - x(t_k)| \leq M\Delta. \tag{4.21}$$

Since $x(t_k) \in \Omega_\rho/\Omega_{\tilde{\rho}}$, it can be concluded that $x(t_k) \in \Omega_\rho/\Omega_{\rho_s}$. Thus, we can write for $t \in [t_k, t_{k+1})$

$$\dot{V}(x(t)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + L'_x M\Delta + L_w\theta. \tag{4.22}$$

If the condition of Eq. 4.15 is satisfied, then there exists $\epsilon_w > 0$ such that the following inequality holds for $x(t_k) \in \Omega_\rho/\Omega_{\tilde{\rho}}$:

$$\dot{V}(x(t)) \leq -\epsilon_w/\Delta, \ \forall t = [t_k, t_{k+1}).$$

Integrating this bound on $t \in [t_k, t_{k+1})$, we obtain that:

$$\begin{aligned} &V(x(t_{k+1})) \leq V(x(t_k)) - \epsilon_w \\ &V(x(t)) \leq V(x(t_k)), \ \forall t \in [t_k, t_{k+1}) \end{aligned} \tag{4.23}$$

for all $x(t_k) \in \Omega_\rho/\Omega_{\tilde{\rho}}$. Using Eq. 4.23 recursively, it is proved that, if $x(t_k) \in \Omega_\rho/\Omega_{\tilde{\rho}}$, the state converges to $\Omega_{\tilde{\rho}}$ in a finite number of sampling times without leaving Ω_ρ.

Part 3: We assume that the DEMPC of Eq. 4.7 operates in the second operation mode. We prove that if $x(t_k) \in \Omega_\rho$, then $V(x(t_{k+1})) \leq V(x(t_k))$ and the system state is ultimately bounded in an invariant set $\Omega_{\rho_{\min}}$. Following the similar steps as in Part

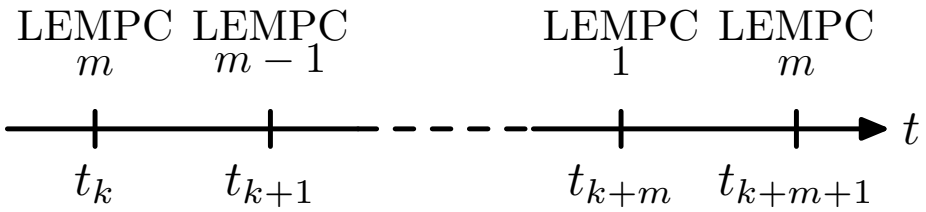


Figure 4.4: Distributed controller evaluation sequence.

2, we can derive that the inequality of Eq. 4.23 hold for all $x(t_k) \in \Omega_\rho / \Omega_{\rho_s}$. Using this result recursively, it is proved that, if $x(t_k) \in \Omega_\rho / \Omega_{\rho_s}$, the state converges to Ω_{ρ_s} in a finite number of sampling times without leaving Ω_ρ. Once the state converges to $\Omega_{\rho_s} \subseteq \Omega_{\rho_{\min}}$, it remains inside $\Omega_{\rho_{\min}}$ for all times. This statement holds because of the definition of $\rho_{\min}$. This proves that the closed-loop system under the LEMPC of Eq. 4.7 is ultimately bounded in $\Omega_{\rho_{\min}}$.

4.4.2 Implementation strategy II

In the implementation strategy introduced in the previous subsection, the evaluation time of the distributed LEMPC at a sampling time is the summation of the evaluation times of all the distributed controllers; this is because at each sampling time all distributed controllers are evaluated in a sequential fashion. However, for applications in which a small sampling time needs to be used and fast controller evaluation is required, we may distribute the evaluation of the distributed controllers into multiple sampling periods. In this implementation strategy, the distributed controllers are evaluated in sequence but over several sampling times and only one controller is evaluated at each sampling time. Figure 4.4 shows a possible evaluation sequence of the distributed controllers in this implementation strategy. In Fig. 4.4, at t_k, LEMPC m is evaluated and it sends the input trajectories of u_m to LEMPC $m-1$; at t_{k+1}, LEMPC $m-1$ is evaluated and it sends u_m and u_{m-1} to LEMPC $m-2$;

from time t_{k+2} to t_{k+m}, LEMPC $m-2$ to LEMPC 1 are evaluated in sequence and one complete distributed control system evaluation cycle is carried out. Another controller evaluation cycle starts at t_{k+m+1} with the evaluation of LEMPC m again. In order to guarantee the closed-loop stability of this implementation strategy, the design of the distributed LEMPC of Eq. 4.7 needs to be modified to account for the multiple sampling time evaluation cycle. We note that both implementation strategy I and implementation strategy II can be executed using parallel computing.

Remark 4.1 *Referring to the choice of the Lyapunov function in the context of a specific chemical process application, we note the following: First, an economically-optimal equilibrium point is computed as the solution of the steady-state optimization problem of Eq. 4.6. This equilibrium point is then used to construct a Lyapunov function for the process expressed in terms of state variable deviations from this equilibrium point (in most applications, quadratic Lyapunov functions can be used; please see application example in section 4.5). Subsequently, this Lyapunov function is used for the design of a state feedback controller $h(x)$ and the computation of the set of initial conditions starting from where closed-loop stability (i.e., convergence to a small neighborhood of the economically-optimal equilibrium point) is guaranteed. This set is typically a level set, Ω_ρ, of the Lyapunov function V embedded within the set where the time derivative of V along the trajectories of the nonlinear closed-loop system with $h(x)$ is negative. Therefore, the construction of Ω_ρ accounts explicitly for the process nonlinearity and it is not a local (i.e., based on the linearization) stability region. Referring to the economic MPC, we note that no assumption is made that the optimization problem at sampling time t_k with $x(t_k) \in \Omega_\rho$ has a unique solution. Due to the incorporation of the Lyapunov-based constraint of Eq. 4.7h, for any $x(t_k) \in \Omega_\rho$, the economic MPC problem has a solution; the one defined by $h(x)$. Therefore, the*

purpose of the optimization problem is to compute control actions over the prediction horizon that optimize the cost of Eq. 4.7 further, yet they satisfy the Lyapunov-based constraint of Eq. 4.7h. Given the constraint that $x(t)$, $t \in [t_k, t_{k+N}]$, *stays in* Ω_ρ, *the economic MPC optimization problem can be solved either locally or globally (with respect to its optimum) within* Ω_ρ, *depending on the type of optimal solution that is required to be found. Note that during mode* 1 *operation under the economic MPC of Eq. 4.7, the Lyapunov constraint is not used to steer the closed-loop system state to the economically-optimal equilibrium point used in the construction of the Lyapunov function but it is simply used to constrain the closed-loop state within a certain operating set (typically* Ω_ρ*) where feasibility of the economic MPC optimization problem is guaranteed. As a consequence, there is no need to impose explicit constraints that limit the discrepancy between* $h(x)$ *and the economic MPC-based control action in the centralized LEMPC case. Finally, due to the use of a finite sampling time, asymptotic stability of the final equilibrium point can not be studied instead practical stability (i.e., ultimate boundedness of the state in a small ball containing the desired steady-state) is studied.*

4.5 Application to nonlinear chemical process network

In this section, we apply the two economic MPC architectures to the process; that is: the centralized Lyapunov-based economic MPC and the sequential distributed Lyapunov-based economic MPC. The objective of all controllers is to drive the system from the stable steady-state defined in Table 4.1 to the economically optimal steady-state. We will also compare the performance of the economic MPC and DMPC with

the performance of a conventional centralized MPC utilizing a quadratic cost function.

4.5.1 Preliminaries

We begin with some preliminaries that will be used in the formulations of the various MPC designs. All MPCs utilize the following Lyapunov function $V(x) = (x - x^s)^T P(x - x^s)$ with P being the same matrix as in Section 4.3.2. We assume that the state x of the system is sampled synchronously and the time instants at which we have state measurements are indicated by the time sequence $t_{k\geq 0}$ with $t_k = t_0 + k\Delta$, $k = 0, 1, \ldots$ where t_0 is the initial time and Δ is the sampling time. The manipulated input in this control problem is defined below with respect to the optimal steady-state input values:

$$\tilde{U} = [\tilde{u}_1 \ ... \ \tilde{u}_8] = [u_1 - u_1^s \ ... \ u_8 - u_8^s]$$

The constraints that all MPC controllers have to satisfy are listed in Table 4.4. It is important to note that even though the input constraints have been modified accordingly, the set U still satisfies the constrains in Table 4.2.

The process model in [63] belongs to the following class of nonlinear systems (which is included in the broad class of nonlinear systems of Eq. 4.1):

$$\dot{x}(t) = f(x) + \sum_{i=1}^{8} g_i(x)\tilde{u}_i$$

where the state x is the deviation of the states variables from the economically-optimal steady-state. For the control of the process, the input $\tilde{u}_1$, $\tilde{u}_2$,$\tilde{u}_3$, $\tilde{u}_4$ and $\tilde{u}_5$ are necessary to keep the stability of the closed-loop system, while $\tilde{u}_6$, $\tilde{u}_7$ and $\tilde{u}_8$ can be used as extra inputs to improve the closed-loop performance. The design of the

Table 4.4: Manipulated input constraints for all controllers.

$\lvert\tilde{u}_1\rvert \leq 3.5 \times 10^5\ [J/s]$	$\lvert\tilde{u}_2\rvert \leq 3.5 \times 10^5\ [J/s]$
$\lvert\tilde{u}_3\rvert \leq 7 \times 10^5\ [J/s]$	$\lvert\tilde{u}_4\rvert \leq 10 \times 10^5\ [J/s]$
$\lvert\tilde{u}_5\rvert \leq 3 \times 10^5\ [J/s]$	$\lvert\tilde{u}_i\rvert \leq 1.7394 \times 10^{-4}\ [m^3/s] (i = 6, 7, 8)$

Lyapunov-based controller $h_i(x)$, $i = 1, ..., 5$ is based on Sontag's formula [93]:

$$h_i(x) = \begin{cases} -\frac{L_f V + \sqrt{(L_f V)^2 + (L_{g_i} V)^4}}{(L_{g_i} V)^2} L_{g_i} V & \text{if } L_{g_i} V \neq 0 \\ 0 & \text{if } L_{g_i} V = 0 \end{cases}$$

where $i = 1, ..., 5$, $L_f V = \frac{\partial V}{\partial x} f(x)$ and $L_{g_i} V = \frac{\partial V}{\partial x} g_i(x)$ denote the Lie derivatives of the scalar function V with respect to the vector fields f and g_i, respectively. The controllers $h_6(x)$, $h_7(x)$ and $h_8(x)$ are chosen to be 0.

For comparison purposes, we consider the following objective function of the conventional centralized MPC:

$$J = \sum_{i=0}^{N\Delta} [x(t_i)^T Q_c x(t_i) + \sum_{j=1}^{8} u_j(t_i)^T R_{cj} u_j(t_i)] \tag{4.24}$$

the weighting matrices are chosen to be $Q_c = diag([1\ 1\ 1\ 1\ 10^3\ 1\ 1\ 1\ 1\ 10^3\ 10\ 10\ 10\ 10\ 10^4\ 1\ 1\ 1\ 1\ 10^3\ 1\ 1\ 1\ 1\ 10^3])$, and $R_{cj} = diag([10^{-8}\ 10^{-8}\ 10^{-8}\ 10^{-8}\ 10^{-8}\ 1\ 1\ 1])$.

4.5.2 Centralized LEMPC

The centralized Lyapunov-based economic MPC design follows the formulation of our previous work [40] with minor modifications (appropriate for the chemical process

example in this work) as follows:

$$\tilde{u}_j^*(t_k) = \max_{u_j \in S(\Delta)} \int_{t_k}^{t_{k+N}} L(\tilde{x}(\tau), \tilde{u}_j(\tau)) d\tau \tag{4.25a}$$

$$\text{s.t. } \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + \sum_{i=1}^{8} g_i(\tilde{x}(t))\tilde{u}_i(t) \tag{4.25b}$$

$$\forall\, t \in [t_k, t_{k+1}),\ k = 0, ..., N-1$$

$$\tilde{u}_j(t) \in U_j \tag{4.25c}$$

$$\tilde{x}(t_k) = x(t_k) \tag{4.25d}$$

$$\sum_{i=2,4,6} F_i = F^{\max} \tag{4.25e}$$

Mode one:

$$\tilde{x}(t_k) \in X \tag{4.25f}$$

$$V(\tilde{x}(t_k)) \leq \tilde{\rho}_e \tag{4.25g}$$

Mode two:

$$\frac{\partial V(\tilde{x}(t_k))}{\partial x} g_j(\tilde{x}(t_k)) u_j(t_k) \leq \inf_{t \in [t_{k-1}, t_k]} \frac{\partial V(\tilde{x}(t))}{\partial x} g_j(\tilde{x}(t)) h_j(\tilde{x}(t)) \tag{4.25h}$$

where $\tilde{x}$ is the predicted closed-loop system state, $S(\Delta)$ is the family of piecewise constant function with period Δ and $t_{k+N} = t_k + N\Delta$. The economic measure L of Eq. 4.25 has the same set up as in Section 4.3.2 Eq. 4.5 and Eq. 4.25e imposes

the quantity constraint of reactant B, from which F_{max} has the same value as in Section 4.3.2.

At mode one operation, Eq. 4.25f of the formulation ensures that the state variable $\tilde{x}(t_k)$ that has been obtained by applying the solution $\tilde{u}_j^*(t_k)$ is bounded. It is important to distinguish the difference between the constraints of Eq. 4.6d and Eq. 4.25f. Since steady-state optimization focuses on a steady-state solution, Eq. 4.6d merely states that the solution has to be bounded; however, the economic MPC, which is a finite-horizon dynamic optimization problem, has to enforce a more aggressive constraint on the closed-loop state trajectory, where the state variables at the end of each sampling time have to be bounded. With respect to the constraint $x(t_k) \in X$, we require that the state variables $x(t_k)$ remain within $\pm 6\%$ of their initial steady-state values for all times; note that the economically optimal steady-state is within X. The level set ρ_e is chosen to be 1.45×10^6. At mode two operation, the constraint of Eq. 4.25h is a tighter version of the Lyapunov-based constraint in [40], and it is used to ensure that the closed-loop system state converges sufficiently close to the economically optimal steady-state.

4.5.3 Sequential distributed LEMPC

In this section, we design a sequential distributed LEMPC architecture for the benzene process. Specifically, the first distributed model predictive controller (DMPC 1) obtains the optimal values of $\tilde{u}_4$ and $\tilde{u}_5$, the second distributed controller (DMPC 2) is designed to obtain the optimal values of $\tilde{u}_1$, $\tilde{u}_2$ and $\tilde{u}_3$, while the third distributed controller (DMPC 3) is designed to obtain the optimal values of $\tilde{u}_6$, $\tilde{u}_7$ and

$\tilde{u}_8$. Specifically, DMPC j ($j = 1, 2, 3$) can be formulated as follows:

$$u^*_{s,j}(t_k) = \max_{u_{s,j} \in S(\Delta)} \int_{t_k}^{t_{k+N}} L(\tilde{x}(\tau), \tilde{u}_{s,j}(\tau)) d\tau \tag{4.26a}$$

$$\text{s.t. } \dot{\tilde{x}}(t) = f(\tilde{x}(t)) + \sum_{i=1}^{8} g_i(\tilde{x}(t)) \tilde{u}_{s,i}(t) \tag{4.26b}$$

$$\forall\, t \in [t_k, t_{k+1}),\ k = 0, ..., N-1$$

$$\tilde{u}_{s,i}(t_k) = u^*_{s,i}(t_k),\ i = 1, \ldots, j-1 \tag{4.26c}$$

$$\tilde{u}_{s,i}(t_k) = h_i(\tilde{x}(t_k)),\ i = j, \ldots, m \tag{4.26d}$$

$$\tilde{u}_{s,j}(t) \in U_j \tag{4.26e}$$

$$\tilde{x}(0) = x(t_k) \tag{4.26f}$$

$$\sum_{i=\{2,4,6\}} F_i = F^{\max} \tag{4.26g}$$

Mode one:

$$\tilde{x}(t_k) \in X_s \tag{4.26h}$$

$$V(\tilde{x}(t_k)) \leq \tilde{\rho}_e \tag{4.26i}$$

Mode two:

$$\frac{\partial V(\tilde{x}(t_k))}{\partial x} g_j(\tilde{x}(t_k)) u_j(t_k) \leq \frac{\partial V(x(t_k))}{\partial x} g_j(\tilde{x}(t_k)) h_j(\tilde{x}(t_k)) \tag{4.26j}$$

where the economic measure L has the same form as in Section 4.3.2 Eq. 4.5 and F^{max} and $\tilde{\rho}_e$ have the same values as in Section 4.5.2.

Since each DMPC controller is designed to obtain a subset of the manipulated inputs, the state constraint enforced in the centralized Lyapunov-based economic MPC design may not be satisfied in each DMPC calculation, and thus, a relaxed version of this constraint is used where we require that the state variables $x(t_k)$ remain within $\pm 7\%$ of their initial steady-state values for all times in all DMPC calculations. Thus, appropriate bounds on the initial state condition needs to be enforced (close enough to the desired steady-state).

4.5.4 Closed-loop simulation results

The simulations were performed in a JAVA platform by a Core2 Quad Q6600 computer. The simulation time for each run is 3000 seconds. Three different simulation cases are studied here in order to evaluate the properties of the proposed controller designs. The first case studies the closed-loop system performance by the centralized LEMPC and by the DLEMPC both operating at mode two. The second case studies the closed-loop system performance by the same controllers but operating at mode one. In the last case, we study the closed-loop system performance by the same controllers operating at mode one first and then at mode two. We will compare the closed-loop system performance of the economic MPC to the performance of the conventional centralized Lyapunov-based MPC (LMPC) which uses the conventional quadratic cost function of Eq. 4.24.

All simulation studies apply the same prediction horizons, which is $N = 3$. Only the first piece from the computed optimal input trajectory of the optimization problems is implemented in each sampling time following a receding horizon scheme. The

sampling time of the optimization problems is $\Delta = 30s$, and as a result, the total number of sampling times alone one simulation is one hundred. All state measurements are available to the MPC controllers at each sampling time.

The numerical method that is used to integrate the process model is explicit Euler with a fixed time step equal to 0.5 seconds. Again, as in the case of steady-state optimization, the optimization problems of each MPC scheme are solved using the open source interior point optimizer Ipopt. The Hessian is approximated by Quasi-Newton's method. Regarding to the termination criteria and the maximum number of iterations, the values used by all simulations are 10^{-3} and 200, respectively.

The results of case one are shown in Fig. 4.5 and Fig. 4.6. It is important to note that even though the conventional centralized MPC does not use the economic measure as its objective function, we recalculate its performance from an economic perspective based on Eq. 4.25a and include it in the figures. The Lyapunov function in Fig. 4.5 clearly indicates that both economic MPC schemes stabilize the process asymptotically to the optimal steady state. Specifically, the centralized LEMPC drives the system to the optimal steady-state, and in terms of accumulated economic measure, it performs better than the conventional MPC controller by 1.5% up to 1000 seconds of simulation time. In contrast, there exists an offset of the economic measure of the DLEMPC, and that is in coincidence with the fact that the value of its Lyapunov function converges to a non-zero positive number at the end. We note here that even though all simulations reported in Figs. 4.5, 4.7 and 4.9 have been carrying out using a total simulation time of 3000 seconds, in order to better show the initial transient behavior in Figs. 4.5 and 4.9, we report the results up to 1500 seconds where the trajectories of all simulations have reached steady-state.

With respect to the centralized LEMPC, we notice that the offset is caused pri-

marily by the structure of the DLEMPC. While each DMPC controller accomplishes each mission successfully according to its formulation, the overall closed-loop cost does not converge to the one of the centralized LEMPC.

With respect to the evaluation time of the different controllers at each sampling time, the DLEMPC outperforms the centralized LEMPC by more than 50% in average and even the centralized MPC at the beginning. It is important to note that the total evaluation time required for the DLEMPC in one sampling time is the sum of the evaluation times of all the DMPC controllers. We also observe that the evaluation time of centralized LMPC overshoots at few sampling times after $t = 1800\ s$. This increase in the controller evaluation time is due to a significant sampling time which allows for deviation of the closed-loop trajectories from the steady-state, as well as the effort of the controller to satisfy the Lyapunov constraint in the first move and the non-convexity of the optimization problem.

The results of case two are shown in Fig. 4.7 and in Fig. 4.8. The Lyapunov function of Fig. 4.7 indicates that starting from $\tilde{\rho}_e = 1.45 \times 10^6$, both control schemes are not able to stabilize the closed-loop system to the economically optimal steady-state but converge to a region with their V values approximately equal to $\tilde{\rho} = 1 \times 10^5$. This result is expected since as we have mentioned in Section 4.3.2 the economically optimal operating steady-state is an unstable steady state. Looking at the economic measure, even though both control schemes have similar value of the Lyapunov function at the end, the DLEMPC has a higher economic measure overall compared with the one of the centralized economic MPC. The reason is due to the different state constraints imposed in each of their problem formulations. Finally, comparing the evaluation time of the different MPC schemes at mode one, we see that, as expected, the DLEMPC outperforms the other schemes at the first 500 seconds. The average

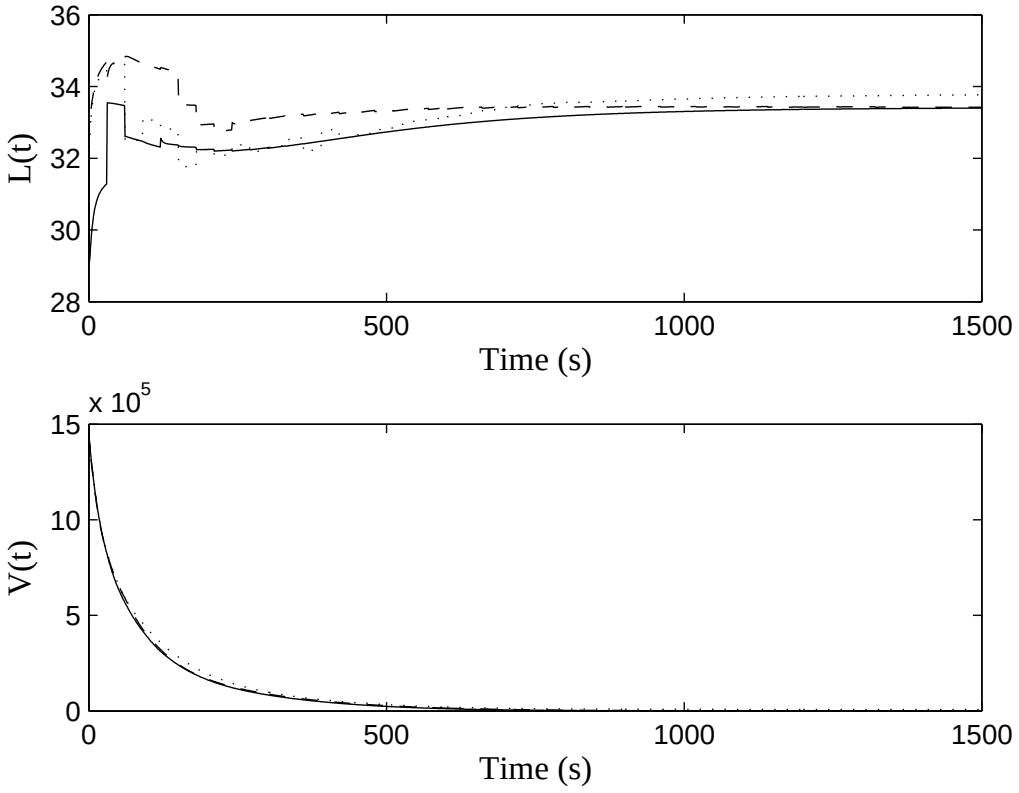


Figure 4.5: Trajectories of the economic measure and of the Lyapunov function using the centralized LMPC with conventional quadratic cost of Eq. 4.24 (solid line), the centralized LEMPC (dashed line) at mode two, and the DLEMPC (dotted line) at mode two. The prediction horizon $N = 3$.

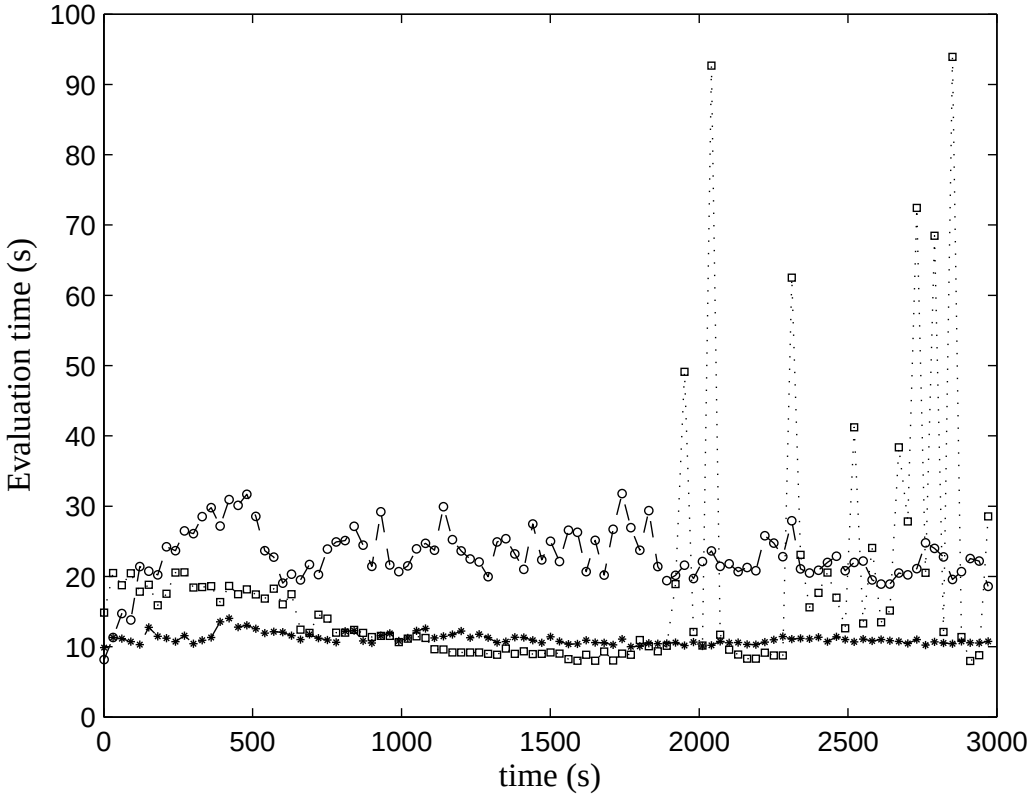


Figure 4.6: The total evaluation time needed for evaluation of each MPC method. Centralized LMPC with conventional quadratic cost of Eq. 4.24 (dotted line with squares), the centralized LEMPC (dashed line with circles) at mode two, and the DLEMPC (solid line with asterisks) at mode two. The prediction horizon $N = 3$.

Table 4.5: Average Control Action Evaluation Time

		Average Control Action Evaluation Time
LEMPC	Mode 1	5.58 seconds
	Mode 2	22.91 seconds
DLEMPC	Mode 1	5.16 seconds
	Mode 2	11.10 seconds
Centralized LMPC		17.49 seconds

control action evaluation times for the different cases are summarized in Table 4.5.

The last two figures (Fig. 4.9 and Fig. 4.10) belong to the study of case three, for which we want to demonstrate that the controllers can switch their operating mode between mode one and mode two under either the centralized LEMPC or the DLEMPC; the switching choice, depends on the control objective, e.g., fast computational time or offset elimination. Finally, it is worthwhile to discuss few benefits of using economic MPC instead of conventional MPC based on the results of this study. First, the centralized LEMPC is characterized by an improved coupling between the different layers of a plant-wide process control system, in particular, the economic optimization and process control layer. In terms of performance, the economic MPC is able to drive the system to the set-point for higher profit return with comparable computational time. Another benefit of applying economic MPC is the ease of tuning. It is a difficult task to assign a reasonable value to the parameters of conventional MPC with quadratic cost since very often they do not have any physical meaning. On the other hand, all parameters of economic MPC have specific economic meaning, and thus, their tuning is more intuitive.

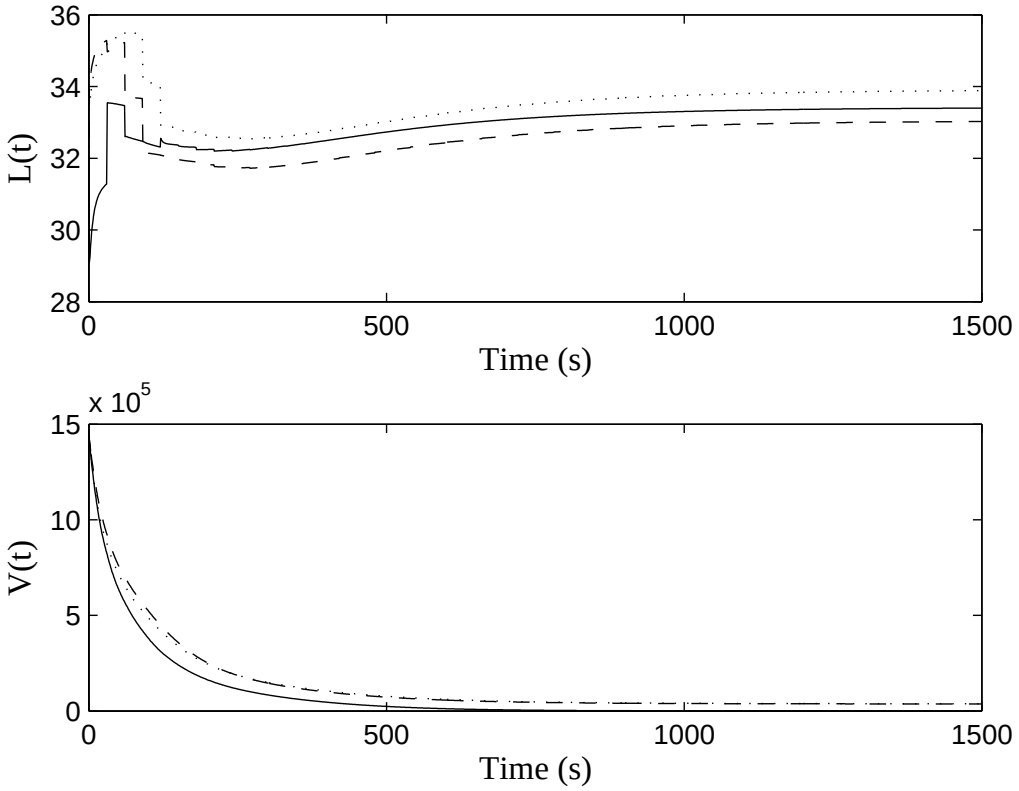


Figure 4.7: Trajectories of the economic measure and of the Lyapunov function using the centralized LMPC with conventional quadratic cost of Eq. 4.24 (solid line), the centralized LEMPC (dashed line) at mode one, and the DLEMPC (dotted line) at mode one. The prediction horizon $N = 3$, and the level set $\tilde{\rho}_e = 1.45 \times 10^6$.

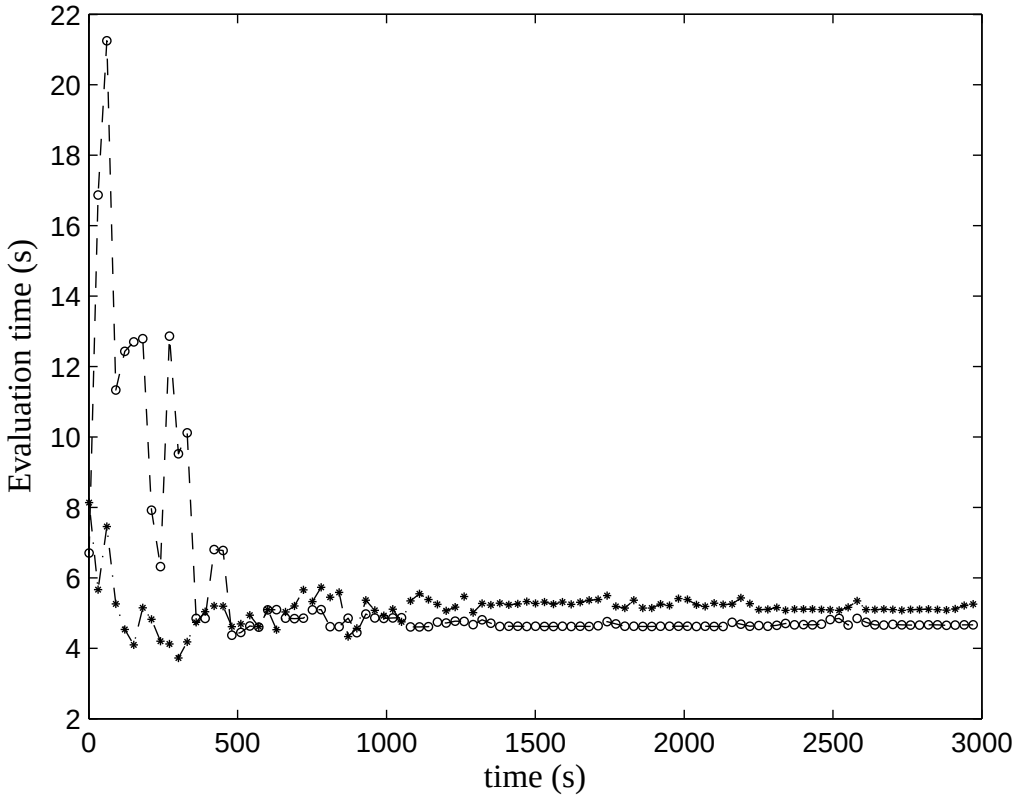


Figure 4.8: The total evaluation time needed for each evaluation of centralized LEMPC (dashed line with circles) at mode one and DLEMPC (solid line with asterisks) at mode one. The prediction horizon $N = 3$.

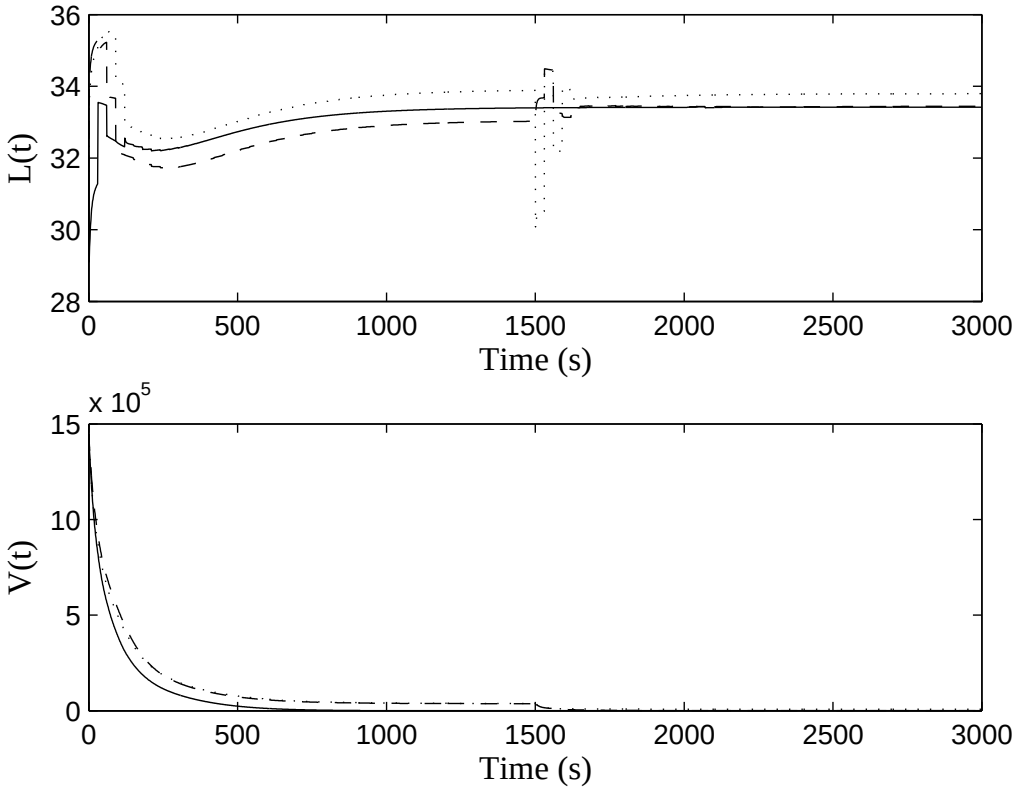


Figure 4.9: Trajectories of the economic measure and of the Lyapunov function by centralized LMPC with conventional quadratic cost of Eq. 4.24 (solid line), the centralized LEMPC (dashed line), and the DLEMPC (dot line). The last two operate at mode one up to $t = 1500$ s ans subsequently at mode two. The prediction horizon $N = 3$, and the level set $\tilde{\rho}_e = 1.45 \times 10^6$.

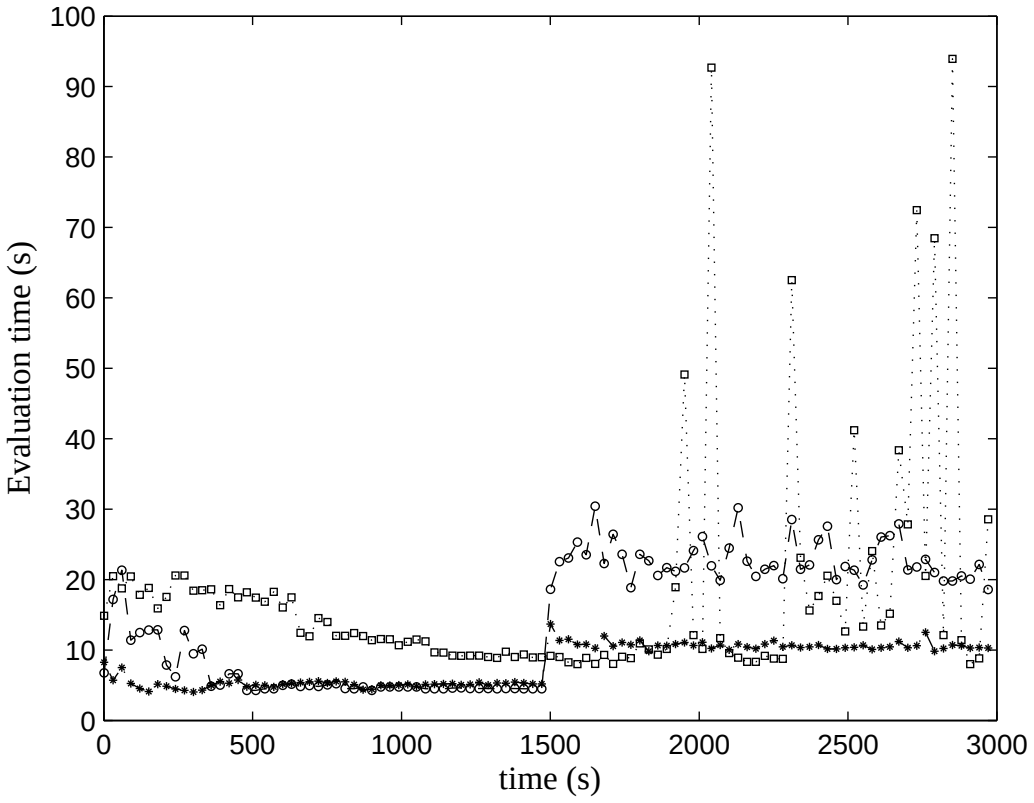


Figure 4.10: The total evaluation time needed for each evaluation of MPC method corresponding to Fig. 4.9. Centralized LMPC with conventional quadratic cost of Eq. 4.24 (dotted line with squares), the centralized LEMPC (dashed line with circles), and DLEMPC (solid line with asterisks). The prediction horizon $N = 3$.

4.6 Conclusions

In this chapter, we carried out an application of centralized LEMPC and sequential distributed LEMPC architectures to a catalytic alkylation of benzene process network which consists of four continuous stirred tank reactors and a flash separator. In the sequential distributed LEMPC design, three separate Lyapunov-based model predictive controllers were designed to control the process in a sequential coordinated fashion. The closed-loop stability properties of the sequential distributed LEMPC design were rigorously analyzed and sufficient conditions for closed-loop stability were established. Simulations were carried out to compare the proposed economic MPC architectures with a centralized LMPC which uses a quadratic cost function that includes penalty on the deviation of the states and inputs from their economically optimal steady-state values, from computational time and closed-loop performance points of view.

Chapter 5

Model Predictive Control of Nonlinear Singularly Perturbed Systems: Application to a Large-Scale Process Network

5.1 Introduction

Chemical processes and plants are characterized by nonlinear behavior and strong coupling of physico-chemical phenomena occurring at disparate time-scales. Examples include fluidized catalytic crackers, distillation columns, biochemical reactors as well as chemical process networks in which the individual processes evolve in a fast time-scale and the network dynamics evolve in a slow time-scale. Singular perturbation theory provides a natural framework for modeling, analysis, order reduction and controller design for nonlinear two-time-scale processes (e.g., [16, 57]).

This chapter focuses on model predictive control of nonlinear singularly perturbed systems in standard form where the separation between the fast and slow state variables is explicit [12]. A composite control system using multirate sampling (i.e., fast sampling of the fast state variables and slow sampling of the slow state variables) and consisting of a "fast" feedback controller that stabilizes the fast dynamics and a model predictive controller that stabilizes the slow dynamics and enforces desired performance objectives in the slow subsystem is designed. Using stability results for nonlinear singularly perturbed systems, the closed-loop system is analyzed and sufficient conditions for stability are derived. A large-scale nonlinear reactor-separator process network is used to demonstrate the application of the method including a distributed implementation of the predictive controller.

5.2 Preliminaries

5.2.1 Notation

The operator $|\cdot|$ is used to denote Euclidean norm of a vector and the symbol Ω_r is used to denote the set $\Omega_r := \{x \in R^{n_x} : V(x) \leq r\}$ where V is a positive definite scalar function. For any measurable (with respect to the Lebesgue measure) function $w : R_{\geq 0} \to R^l$, $||w||$ denotes ess.sup.$|w(t)|$, $t \geq 0$. A function $\gamma : R_{\geq 0} \to R_{\geq 0}$ is said to be of class K if it is continuous, nondecreasing, and is zero at zero. A function $\beta : R_{\geq 0} \times R_{\geq 0} \to R_{\geq 0}$ is said to be of class KL if, for each fixed t, the function $\beta(\cdot, t)$ is of class K and, for each fixed s, the function $\beta(s, \cdot)$ is nonincreasing and tends to zero at infinity. The symbol $diag(v)$ denotes a matrix whose diagonal elements are the elements of vector v and all the other elements are zeros.

5.2.2 Class of nonlinear singularly perturbed systems

In this work, we focus on nonlinear singularly perturbed systems in standard form with the following state-space description:

$$\begin{aligned} \dot{x} &= f(x, z, \epsilon, u_s, w), \quad x(0) = x_0 \\ \epsilon \dot{z} &= g(x, z, \epsilon, u_f, w), \quad z(0) = z_0 \end{aligned} \tag{5.1}$$

where $x \in R^n$ and $z \in R^m$ denote the vector of state variables, ϵ is a small positive parameter, $w \in R^l$ denotes the vector of disturbances and $u_s \in U \subset R^p$ and $u_f \in V \subset R^q$ are two sets of manipulated inputs. The sets U and V are nonempty convex sets which are defined as follows:

$$\begin{aligned} U &:= \{u_{s,i}(t) : |u_{s,i}(t)| \leq u_{s,i}^{\max},\ i \in [1, p]\} \\ V &:= \{u_{f,j}(t) : |u_{f,j}(t)| \leq u_{f,j}^{\max},\ j \in [1, q]\} \end{aligned} \tag{5.2}$$

where $u_{s,i}^{\max}$ and $u_{f,j}^{\max}$ are positive real numbers, specifying the input constraints. The disturbance vector is assumed to be absolutely continuous and bounded, i.e., $W := \{w(t) \in R^l : |w(t)| \leq \theta\}$ where θ is a positive real number. Since the small parameter ϵ multiplies the time derivative of the vector z in the system of Eq. 5.1, the separation of the slow and fast variables in Eq. 5.1 is explicit, and thus, we will refer to the vector x as the slow states and to the vector z as the fast states. We assume that the vector fields f and g are locally Lipschitz in $R^n \times R^m \times [0, \bar{\epsilon}) \times R^p \times R^q \times R^l$ for some $\bar{\epsilon} > 0$ and that the origin is an equilibrium point of the unforced nominal system (i.e., system of Eq. 5.1 with $u_s = 0$, $u_f = 0$ and $w = 0$).

With respect to the control problem formulation, we assume that the fast states z are sampled continuously and their measurements are available for all time t (for example, variables for which fast sampling is possible usually include temperature,

pressure and hold-ups) while the slow states x are sampled synchronously and are available at time instants indicated by the time sequence $\{t_{k\geq 0}\}$ with $t_k = t_0 + k\Delta,\ k = 0, 1, \ldots$ where t_0 is the initial time and Δ is the sampling time (for example, slowly sampled variables usually involve species concentrations). The set of manipulated inputs u_f is responsible for stabilizing the fast dynamics of Eq. 5.1 and for this set the control action is assumed to be computed continuously, while the set of manipulated inputs u_s is evaluated at each sampling time t_k and is responsible for stabilizing the slow dynamics and enforcing a desired level of optimal closed-loop performance.

5.2.3 Two-time-scale system decomposition

The explicit separation of the slow and fast variables in the system of Eq. 5.1 allows decomposing it into two separate reduced-order systems evolving in different time-scales. To proceed with such a two-time-scale decomposition and in order to simplify the notation of the subsequent development, we will first address the issue of stability of the fast dynamics. Since there is no assumption that the fast dynamics of Eq. 5.1 are asymptotically stable, we assume the existence of a "fast" feedback control law $u_f = p(x, z)$ that renders the fast dynamics asymptotically stable in a sense to be made precise in Assumption 5.2 below. Substituting $u_f = p(x, z)$ in Eq. 5.1 and setting $\epsilon = 0$ in the resulting system, we obtain:

$$\frac{dx}{dt} = f(x, z, 0, u_s, w) \tag{5.3a}$$

$$0 = g(x, z, 0, p(x, z), w) \tag{5.3b}$$

Assumption 5.1 *The equation* $g(x, z, 0, p(x, z), w) = 0$ *possesses a unique root*

$$z = \hat{g}(x, w) \tag{5.4}$$

with the properties that $\hat{g} : R^n \times R^l \rightarrow R^m$ *and its partial derivatives* $\dfrac{\partial \hat{g}}{\partial x}$, $\dfrac{\partial \hat{g}}{\partial w}$ *are locally Lipschitz.*

Assumption 5.1 is a standard requirement in singularly perturbation theory (please see, for example, [54]) and it is made to ensure that the system has an isolated equilibrium manifold for the fast dynamics. On this manifold, z can be expressed in terms of x and w using an algebraic expression. This assumption does not pose any practical limitation in the example but it is a necessary one in the singular perturbation framework to construct a well-defined slow subsystem.

Using $z = \hat{g}(x, w)$, we can re-write Eq. 5.3 as follows:

$$\frac{dx}{dt} = f(x, \hat{g}(x, w), 0, u_s, w) =: f_s(x, u_s, w) \tag{5.5}$$

We will refer to the subsystem of Eq. 5.5 as the slow subsystem.

Introducing the fast time scale $\tau = \dfrac{t}{\epsilon}$ and the deviation variable $y = z - \hat{g}(x, w)$, we can rewrite the nonlinear singularly perturbed system of Eq. 5.1 as follows:

$$\begin{aligned} \frac{dx}{d\tau} &= \epsilon f(x, y + \hat{g}(x, w), \epsilon, u_s, w) \\ \frac{dy}{d\tau} &= g(x, y + \hat{g}(x, w), \epsilon, u_f, w) - \epsilon \frac{\partial \hat{g}}{\partial w} \dot{w} \\ &\quad -\epsilon \frac{\partial \hat{g}}{\partial x} f(x, y + \hat{g}(x, w), \epsilon, u_s, w) \end{aligned} \tag{5.6}$$

Setting $\epsilon = 0$, we obtain the following fast subsystem:

$$\frac{dy}{d\tau} = g(x, y + \hat{g}(x, w), 0, u_f, w) \tag{5.7}$$

where x and w can be considered as "frozen" to their initial values. Below we state our assumption on the stabilization of the fast subsystem:

Assumption 5.2 *There exists a feedback control law* $u_f = p(x, z) = p(x, y+\hat{g}(x, w)) \in V$ *where* $p(x, z)$ *is a locally Lipschitz vector function of its arguments, such that the origin of the closed-loop fast subsystem:*

$$\frac{dy}{d\tau} = g(x, y + \hat{g}(x, w), 0, p(x, y + \hat{g}(x, w)), w) \tag{5.8}$$

is globally asymptotically stable, uniformly in $x \in R^n$ *and* $w \in R^l$*, in the sense that there exists a class* KL *function* β_y *such that for any* $y(0) \in R^m$*:*

$$|y(t)| \leq \beta_y(|y(0)|, \frac{t}{\epsilon}) \tag{5.9}$$

for $t \geq 0$*.*

5.2.4 Lyapunov-based controller

We assume that there exists a Lyapunov-based locally Lipschitz control law $h(x) = [h_1(x) \ \ldots \ h_p(x)]^T$ with $u_{s,i} = h_i(x)$, $i = 1, \ldots, p$, which renders the origin of the nominal closed-loop slow subsystem asymptotically stable while satisfying the input constraints for all the states x inside a given stability region. Using converse Lyapunov theorems [75, 60, 18], this assumption implies that there exist functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$ of class K and a continuously differentiable Lyapunov function $V(x)$ for the

nominal closed-loop slow subsystem that satisfy the following inequalities:

$$\begin{gathered}\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \\ \frac{\partial V(x)}{\partial x}(f_s(x, h(x), 0)) \leq -\alpha_3(|x|) \\ h(x) \in U\end{gathered} \tag{5.10}$$

for all $x \in D \subseteq R^n$ where D is an open neighborhood of the origin. We denote the region $\Omega_\rho \subseteq D$ as the stability region of the closed-loop slow subsystem under the Lyapunov-based controller $h(x)$. By continuity, the local Lipschitz property assumed for the vector fields $f_s(x, u_s, w)$ and taking into account that the manipulated inputs u_i, $i = 1, \ldots, p$, and the disturbance w are bounded in convex sets, there exists a positive constant M such that

$$|f_s(x, u_s, w)| \leq M \tag{5.11}$$

for all $x \in \Omega_\rho$, $u_s \in U$, and $w \in W$. In addition, by the continuous differentiable property of the Lyapunov function $V(x)$ and the Lipschitz property assumed for the vector field $f_s(x, u_s, w)$, there exist positive constants L_x and L_w such that

$$\begin{gathered}|\frac{\partial V}{\partial x} f_s(x, u_s, w) - \frac{\partial V}{\partial x} f_s(x', u_s, w)| \leq L_x |x - x'| \\ |\frac{\partial V}{\partial x} f_s(x, u_s, w) - \frac{\partial V}{\partial x} f_s(x, u_s, w')| \leq L_w |w - w'|\end{gathered} \tag{5.12}$$

for all $x, x' \in \Omega_\rho$, $u_s \in U$, and $w, w' \in W$.

5.2.5 Lyapunov-based MPC formulation

The longer sampling time of the slow state variables allows utilizing MPC to compute the control action u_s. A schematic of the proposed control system structure is shown

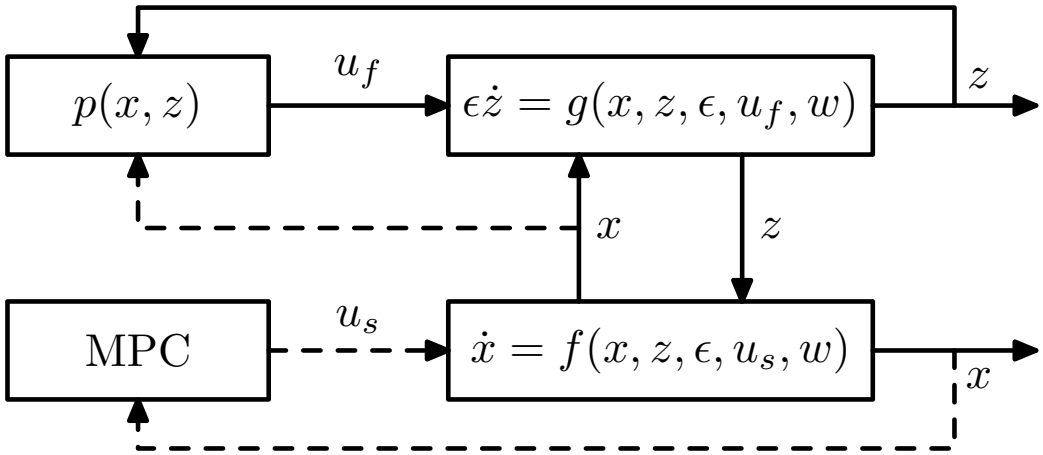


Figure 5.1: A schematic of the proposed control system structure.

in Fig. 5.1. Specifically, we use the LMPC proposed in [79] which guarantees practical stability of the closed-loop system and allows for an explicit characterization of the stability region to compute u_s. The LMPC is based on the Lyapunov-based controller $h(x)$. The controller $h(x)$ is used to define a stability constraint for the LMPC controller which guarantees that the LMPC controller inherits the stability and robustness properties of the Lyapunov-based controller $h(x)$. The LMPC controller is based on the following optimization problem:

$$\min_{u_s \in S(\Delta)} \int_0^{N_c\Delta} [\tilde{x}^T(\tau) Q_c \tilde{x}(\tau) + u_s^T(\tau) R_c u_s(\tau)] d\tau \tag{5.13a}$$

$$\text{s.t. } \dot{\tilde{x}}(\tau) = f_s(\tilde{x}(\tau), u_s, 0) \tag{5.13b}$$

$$u_s(\tau) \in U_s \tag{5.13c}$$

$$\tilde{x}(0) = x(t_k) \tag{5.13d}$$

$$\frac{\partial V(x)}{\partial x} f_s(x(t_k), u_s(0), 0) \leq \frac{\partial V(x)}{\partial x} f_s(x(t_k), h(x(t_k)), 0) \tag{5.13e}$$

where $S(\Delta)$ is the family of piece-wise constant functions with sampling period Δ, N_c is the prediction horizon, Q_c and R_c are positive definite weight matrices that define the cost, $x(t_k)$ is the state measurement obtained at t_k, $\tilde{x}$ is the predicted trajectory of the nominal system with u_s, the input trajectory computed by the LMPC of Eq. 5.13. The optimal solution to this optimization problem is denoted by $u_s^*(\tau|t_k)$, and is defined for $\tau \in [0, N_c\Delta)$.

The optimization problem of Eq. 5.13 does not depend on the uncertainty and guarantees that the system in closed-loop with the LMPC controller of Eq. 5.13 maintains the stability properties of the Lyapunov-based controller. The constraint of Eq. 5.13e guarantees that the value of the time derivative of the Lyapunov function at the initial evaluation time of the LMPC is lower or equal to the value obtained if only the Lyapunov-based controller $h(x)$ is implemented in the closed-loop system in a sample-and-hold fashion. This is the constraint that allows proving that the LMPC inherits the stability and robustness properties of the Lyapunov-based controller. The manipulated inputs of the closed-loop slow subsystem under the LMPC controller are defined as follows

$$u_s(t) = u_s^*(t - t_k|t_k), \ \forall t \in [t_k, t_{k+1}). \tag{5.14}$$

The main property of the LMPC controller is that the origin of the closed-loop system is practically stable for all initial states inside the stability region Ω_ρ for a sufficient small sampling time Δ and disturbance upper bound θ. The main advantage of LMPC approaches with respect to the Lyapunov-based controller is that optimality considerations can be taken explicitly into account (as well as constraints on the inputs and the states [79]) in the computation of the controller within an online optimization framework improving closed-loop performance.

Proposition 5.1 (c.f. [79, 80]) *Consider the slow subsystem of Eq. 5.5 in closed-*

loop under the LMPC design of Eq. 5.14 based on a Lyapunov-based controller $h(x)$ that satisfies the conditions of Eq. 5.10. Let $\epsilon_w > 0$, $\Delta > 0$ and $\rho > \rho_s > 0$, $\theta > 0$ satisfy the following constraint:

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + L_x M\Delta + L_w\theta \leq -\epsilon_w/\Delta. \tag{5.15}$$

There exists a class KL function β_x and a class K function γ such that if $x(0) \in \Omega_\rho$, then $x(t) \in \Omega_\rho$ for all $t \geq 0$ and

$$|x(t)| \leq \beta_x(|x(0)|, t) + \gamma(\rho^*) \tag{5.16}$$

with $\rho^* = \max\{V(x(t+\Delta)) : V(x(t)) \leq \rho_s\}$.

5.3 Stability analysis

The closed-loop stability of the system of Eq. 5.1 under the control of the controller $p(x, z)$ and the LMPC of Eq. 5.13 is established in the following theorem under appropriate conditions.

Theorem 5.1 *Consider the system of Eq. 5.1 in closed-loop with $u_f = p(x, z)$ and u_s determined by the LMPC of Eq. 5.13 based on a controller $h(\cdot)$ that satisfies the conditions of Eq. 5.10. Let also assumptions 5.1 and 5.2 and the condition of Eq. 5.15 hold. Then there exist functions β_x and β_y of class KL, a pair of positive real numbers (δ, d) and $\epsilon^* > 0$ such that if* $\max\{|x(0)|, |y(0)|, ||w||, ||\dot{w}||\}$

$\leq \delta$ *and* $\epsilon \in (0, \epsilon^*]$, *then,*

$$\begin{aligned} |x(t)| &\leq \beta_x(|x(0)|, t) + \gamma(\rho^*) + d \\ |y(t)| &\leq \beta_y(|y(0)|, \frac{t}{\epsilon}) + d \end{aligned} \tag{5.17}$$

for all $t \geq 0$.

Proof: When $u_f = p(x, z)$ and $u_s = u_s^*$ is determined by the LMPC of Eq. 5.14, the closed-loop system takes the following form:

$$\begin{aligned} \dot{x} &= f(x, z, \epsilon, u_s^*, w), \;\; x(0) = x_0 \\ \epsilon \dot{z} &= g(x, z, \epsilon, p(x, z), w), \;\; z(0) = z_0. \end{aligned} \tag{5.18}$$

We will first compute the slow and fast closed-loop subsystems. Setting $\epsilon = 0$ in Eq. 5.18, we obtain:

$$\begin{aligned} \frac{dx}{dt} &= f(x, z, 0, u_s^*, w) \\ 0 &= g(x, z, 0, p(x, z), w). \end{aligned} \tag{5.19}$$

Using that the second equation has a unique, isolated solution $z = \hat{g}(x, w)$ (assumption 5.1), we can re-write 5.19 as follows:

$$\frac{dx}{dt} \;\; = \;\; f(x, \hat{g}(x, w), 0, u_s^*, w) = f_s(x, u_s^*, w) \tag{5.20}$$

According to Proposition 5.1, the state $x(t)$ of the closed-loop slow subsystem of Eq. 5.20 starting from $x(0) \in \Omega_\rho$ stays in Ω_ρ (i.e., $x(t) \in \Omega_\rho \; \forall t \geq 0$) and satisfies the bound of Eq.5.16.

We now turn to the fast subsystem. Using $\tau = \frac{t}{\epsilon}$ and $y = z - \hat{g}(x,w)$, the closed-loop system of Eq. 5.18 can be written as:

$$\begin{aligned} \frac{dx}{d\tau} &= \epsilon f(x, y + \hat{g}(x,w), \epsilon, u_s(x), w) \\ \frac{dy}{d\tau} &= g(x, y + \hat{g}(x,w), \epsilon, p(x,y), w) - \epsilon \frac{\partial \hat{g}}{\partial w}\dot{w} \\ &\quad -\epsilon \frac{\partial \hat{g}}{\partial x} f(x, y + \hat{g}(x,w), u_s(x), w) \end{aligned} \tag{5.21}$$

Setting $\epsilon = 0$, the closed-loop fast subsystem is obtained:

$$\frac{dy}{d\tau} = g(x, y + \hat{g}(x,w), 0, p(x,y), w) \tag{5.22}$$

According to Assumption 5.2, the origin of the system of Eq. 5.22 is globally asymptotically stable, uniformly in $x \in R^n$ and $w \in R^l$ in the sense that there exists a class KL function β_y such that for any $y(0) \in R^m$, the bound of Eq. 5.9 holds for $t \geq 0$. Therefore, the closed-loop system of Eq. 5.18 satisfies the assumptions 1, 2 and 3 of Theorem 1 in [20]. Thus, there exist functions β_x and β_y of class KL, positive real numbers (δ, d) (note that the existence of δ such that $|x(0)| \leq \delta$ implies that $x(0) \in \Omega_\rho$ follows from the smoothness of $V(x)$), and $\epsilon^* > 0$ such that if $\max\{|x(0)|, |y(0)|, ||w||, ||\dot{w}||\} \leq \delta$ and $\epsilon \in (0, \epsilon^*]$, then, the bounds of Eq.5.17 hold for all $t \geq 0$. ■

Remark 5.1 *We note that the class of nonlinear systems of Eq. 5.1 can be generalized to include: a) u_s in the g vector field (i.e., the manipulated inputs that are used to control the slow subsystem affect directly the fast dynamics), and b) u_f in the f vector field (i.e., the manipulated inputs that are used to control the fast subsystem affect directly the slow dynamics). Such a generalization would simply require that u_s, which is computed by the MPC and is piecewise continuous in time, is passed through an*

appropriate filter to become absolutely continuous (see also Theorem 1 and Remark 1 in [20]); this generalization is not pursued here in order to avoid complicating further the notation. We also note that instead of using the LMPC of Eq. 5.13 other MPC schemes including distributed MPC schemes can be used to control the slow subsystem and they will inherit the stability properties of Theorem 5.1 as long as these MPC schemes satisfy the conditions of Proposition 5.1.

5.4 Application to a nonlinear large-scale process network

5.4.1 Process description and control system design

The process considered in this study is a reactor-distillation process network, shown in Fig. 5.2 (see also [57]). It consists of a continuously stirred tank reactor (CSTR), a distillation tower including a reboiler and a condenser, and a recycle loop. A set of elementary exothermic reactions in series takes place in the reactor of the form $A \xrightarrow{k_1} B \xrightarrow{k_2} C$, in which A is the reactant, B is the desired product and C is the by-product. The reactor is fed with a fresh feed of pure species A at flowrate F_0. The outlet of the reactor is fed into the distillation tower, where most of the reactant A is separated overhead and recycled back to the CSTR, and most of the product and the by-product leave the system through stream B_t. There are three heat/coolant inputs, labeled as Q_1, Q_2, and Q_3, that are assigned to the CSTR, the condenser, and the reboiler, respectively. The flow rates of streams F, D and B_t are regulated by three valves, labeled as $V1$, $V2$, and $V3$, respectively. The dynamic equations describing the behavior of the process are obtained through material and energy balances under standard modeling assumptions. Specifically, the dynamic model of the CSTR is as

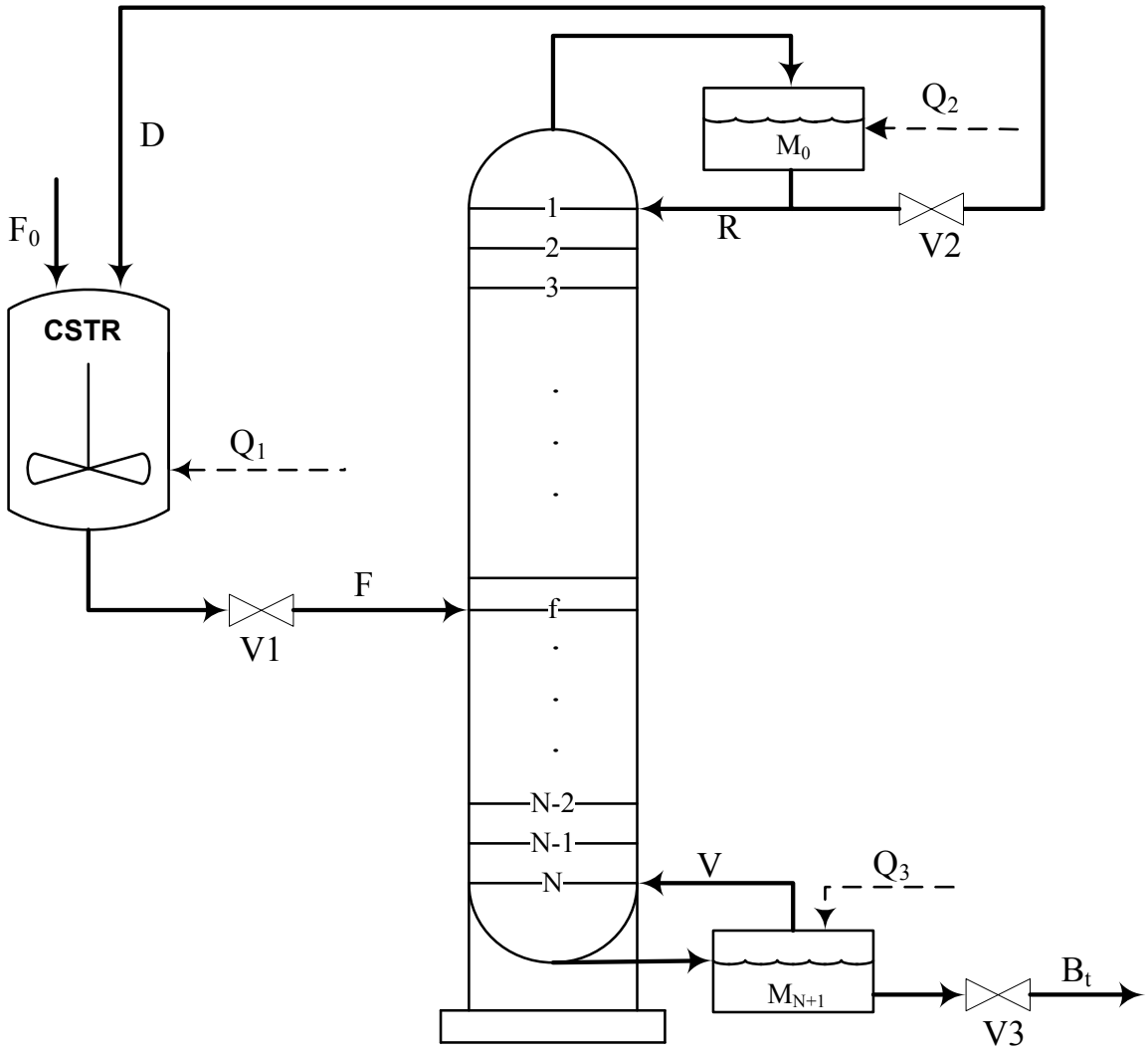


Figure 5.2: Chemical process network schematic.

follows:

$$\dot{M}_R = F_0 + D - F \tag{5.23a}$$

$$\dot{x}_{A,R} = \frac{F_0(1 - x_{A,R}) + D(x_{A,0} - x_{A,R})}{M_R} - k_1 e^{-E_1/RT} x_{A,R} \tag{5.23b}$$

$$\dot{x}_{B,R} = \frac{-F_0 x_{B,R} + D(x_{B,0} - x_{B,R})}{M_R} + k_1 e^{-E_1/RT} x_{A,R}$$

$$- k_2 e^{-E_2/RT} x_{B,R} \tag{5.23c}$$

$$\dot{H}_{L,R} = \frac{F_0(H_{L,F_0} - H_{L,R}) + D(H_{L,0} - H_{L,R})}{M_R} + \frac{Q_1}{M_R}$$

$$- k_1 e^{-E_1/RT} x_{A,R} \Delta H_{r1} - k_2 e^{-E_2/RT} x_{B,R} \Delta H_{r2} \tag{5.23d}$$

The dynamic model of the condenser is as follows:

$$\dot{M}_0 = \bar{V} - R - D \tag{5.24a}$$

$$\dot{x}_{i,0} = \frac{\bar{V}}{M_0}(y_{i,1} - x_{i,0}) \tag{5.24b}$$

$$\dot{H}_{L,0} = \frac{\bar{V}}{M_0}(H_{V,1} - H_{L,0}) + \frac{Q_2}{M_0} \tag{5.24c}$$

where $i = A, B, C$. The dynamic model of the distillation column is as follows:

$$\dot{x}_{i,j} = \frac{1}{M_j}[\bar{V}(y_{i,j+1} - y_{i,j}) + R(x_{i,j-1} - x_{i,j})], 1 \leq j < f \tag{5.25a}$$

$$\dot{H}_{L,j} = \frac{\bar{V}}{M_j}(H_{V,j+1} - H_{V,j}) + \frac{R}{M_j}(H_{L,j-1} - H_{L,j}), 1 \leq j < f \tag{5.25b}$$

$$\dot{x}_{i,f} = \frac{1}{M_f}[\bar{V}(y_{i,f+1} - y_{i,f}) + R(x_{i,f-1} - x_{i,f}) + F(x_{i,R} - x_{i,f})], \tag{5.25c}$$

$$j = f$$

$$\dot{H}_{L,f} = \frac{\bar{V}}{M_f}(H_{V,f+1} - H_{V,f}) + \frac{R}{M_f}(H_{L,f-1} - H_{L,f}) + \frac{F}{M_f}(H_{L,R} - H_{L,f}) \tag{5.25d}$$

$$j = f$$

$$\dot{x}_{i,j} = \frac{1}{M_j}[\bar{V}(y_{i,j+1} - y_{i,j}) + (R+F)(x_{i,j-1} - x_{i,j})], f < j \leq N \tag{5.25e}$$

$$\dot{H}_{L,j} = \frac{\bar{V}}{M_j}(H_{V,j+1} - H_{V,j}) + \frac{R+F}{M_j}(H_{L,j-1} - H_{L,j}), f < j \leq N \tag{5.25f}$$

where $i = A, B, C$ and N is the number of column stages. Finally, the dynamic model of the reboiler is as follows:

$$\dot{M}_{N+1} = R + F - \bar{V} - B_t \tag{5.26a}$$

$$\dot{x}_{i,N+1} = \frac{1}{M_{N+1}}[(R+F)(x_{i,N} - x_{i,N+1}) - \bar{V}(y_{i,N+1} - x_{i,N+1})] \tag{5.26b}$$

$$\dot{H}_{L,N+1} = \frac{R+F}{M_{N+1}}(H_{L,N} - H_{L,N+1}) - \frac{\bar{V}}{M_{N+1}}(H_{V,N+1} - H_{L,N+1}) + \frac{Q_3}{M_{N+1}} \tag{5.26c}$$

where $i = A, B, C$. The definitions of the process parameters and their nominal values are given in Table 5.1 and in Table 5.2, respectively.

The model of the CSTR assumes perfect mixing and spatially uniform heat conduction. Both reactions in the reactor are first-order elementary reactions. The composition of species C can be computed by the following relationship, $x_{A,R} + x_{B,R} + x_{C,R} = 1$. For the derivation of the dynamic model of the multicomponent distillation, we apply stage-by-stage methods and batch rectification. To apply this approach, we

Table 5.1: Process variables

F_0, D, F, R, $\bar{V}$, B	Effluent flow rates
$\tilde{F}_0$, $\tilde{D}$, $\tilde{F}$, $\tilde{R}$, $\tilde{V}$, $\tilde{B}$	Steady-state values of effluent flow rates
$x_{i,R}$	Species composition in the CSTR
$x_{i,j}$	Species composition in the distillation tower
M_R, M_0, M_{M+1}	Liquid hold-up in each vessel
$C_{P_V,i}$	Heat capacity of each species at vapor phase
α_i	Relative volatilities of each species
ΔH_{r1}, ΔH_{r2}	Heat of reactions 1 and 2
$H_{V,R}$	Enthalpy of mixture in the CSTR
$H_{V,j}$	Enthalpy of gas mixture
$H_{L,j}$	Enthalpy of liquid mixture
H_{L,F_0}	Enthalpy of feed input
k_1, k_2	Reaction coefficient
E_1,E_2	Activation energy
Q_1, Q_2, Q_3	External heat/coolant inputs to each vessel

Table 5.2: Parameter values

ΔH_{r1}	$2,500$ $[J/mol]$	ΔH_{r2}	$5,500$ $[J/mol]$
E_1	$9,500$ $[J/mol]$	E_2	$12,000$ $[J/mol]$
k_1	2.4 $[1/s]$	k_2	4.0 $[1/s]$
$F_0, \tilde{F}_0$	100 $[mol/s]$	H_{L,F_0}	61.06 $[J/mol]$

assume vapor-liquid equilibrium in each stage, perfect mixing of liquid and vapor in each stage, negligible vapor holdup, constant-molar-liquid holdup, M_j, on each stage, and adiabatic process for the entire distillation process. In this work, the thermodynamic properties of the mixtures are obtained by assuming ideal behavior in both liquid phase and vapor phase. Specifically, the enthalpy of each species in vapor state is described by the following expression:

$$h_{V,i} = h^o_{V,i} + C_{P_V,i}(T - T_0)$$

where T_0 is the reference temperature and its value is 373.15 K, $h^o_{V,i}$ is the enthalpy of a species at the reference temperature and $C_{P_V,i}$ is the heat capacity of a species and is assumed to be a constant. The derivation of the enthalpy of a vapor mixture and the enthalpy of a liquid mixture, based on above assumptions, is given by:

$$\begin{aligned} H_V &= \sum_i^{A,B,C} y_i h^0_{V,i} + (T - T_0) \sum_i^{A,B,C} y_i C_{p_V,i} \\ H_L &= \sum_i^{A,B,C} x_i (h^0_{V,i} - \Delta H_i^{Vap}) + (T - T_0) \sum_i^{A,B,C} x_i C_{p_V,i} \end{aligned} \tag{5.27}$$

If the enthalpy of a liquid mixture is known, we can obtain the temperature using the following expression:

$$T = \frac{H_L - \sum_i^{A,B,C} x_i (h^0_{V,i} - \Delta H_i^{Vap})}{\sum_i^{A,B,C} x_i C_{p_V,i}} + T_0$$

Furthermore, the enthalpy of the vapor mixture can be obtained by substituting the computed temperature value back into Eq. 5.27. For ideal liquid-vapor mixture, Raoult's law determines the relationship between the vapor phase molar composition

Table 5.3: Process parameters

	A	B	C
$C_{pV,i}$ $[J/mol \cdot K]$	1.86	2.01	2.00
ΔH_i^{Vap} $[J/mol]$	83.333	86.111	85.556
$h_{V,i}^o$ $[J/mol]$	283.889	369.844	394.444
α_i	5.5	1.2	1.0

Table 5.4: Final steady-state manipulated input values

$\tilde{Q}_1$ 2.85·10^5 $[J/s]$	$\tilde{Q}_2$ -1.93·10^5 $[J/s]$	$\tilde{Q}_3$ 2.31·10^5 $[J/s]$	$\tilde{F}$ 1880 $[mol/s]$
$\tilde{V}$ 2070 $[mol/s]$	$\tilde{B}_t$ 100 $[mol/s]$	$\tilde{D}$ 1780 $[mol/s]$	$\tilde{R}$ 290 $[mol/s]$

and the liquid phase molar composition of each species. In this model, we assume that the vapor pressure of each species, or the relative volatility of each species, is a constant. Hence, the following equation, based on Raoult's law, can be used to compute the vapor phase molar composition, once the liquid phase molar composition is known:

$$y_i = \frac{\alpha_i x_i}{\sum_{k}^{A,B,C} \alpha_k x_k}$$

For the other thermodynamic parameters, one can refer to Table 5.3 for their nominal values. The distillation tower has a total of 15 trays, and the reactor outlet is fed into tray 12. The entire process network has a total of 57 states which consist of the compositions of A, B, and C in the reactors, column stages, reboiler and condenser, as well as the enthalpy in each of the vessels. The desired (final) operating point of the process, corresponding to the seven steady-state manipulated input values, $\tilde{F}, \tilde{V}, \tilde{B}, \tilde{R}, \tilde{D}, \tilde{Q}_1, \tilde{Q}_2$, and $\tilde{Q}_3$ (Table 5.4), is given in Table 5.5.

Table 5.5: Final steady-state values of the states of CSTR, reboiler and condenser

$\tilde{M}_R$	1100 $[mol]$	$\tilde{M}_0$	1050 $[mol]$	$\tilde{M}_{N+1}$	1200 $[mol]$
$\tilde{x}_{A,R}$	0.897	$\tilde{x}_{A,0}$	0.948	$\tilde{x}_{A,N+1}$	0.00666
$\tilde{x}_{B,R}$	0.0965	$\tilde{x}_{B,0}$	0.0505	$\tilde{x}_{B,N+1}$	0.916
$\tilde{H}_{L,R}$	1.849 $\cdot 10^2$ $[J/mol]$	$\tilde{H}_{L,0}$	1.952 $\cdot 10^2$ $[J/mol]$	$\tilde{H}_{L,N+1}$	3.826 $\cdot 10^2$ $[J/mol]$

Table 5.6: Initial steady-state manipulated input values

Q_1	$3.58{\cdot}10^5$ [J/s]	Q_2	$-2.00{\cdot}10^5$ [J/s]	Q_3	$2.335{\cdot}10^5$ [J/s]	F	1880 $[mol/s]$
V	2070 $[mol/s]$	B_t	100 $[mol/s]$	D	1780 $[mol/s]$	R	290 $[mol/s]$

Table 5.7: Initial steady state values of the states of CSTR, reboiler and condenser

M_R	1300 $[mol]$	$x_{A,R}$	0.763
M_0	1125 $[mol]$	$x_{A,0}$	0.806
M_{N+1}	1425 $[mol]$	$x_{A,N+1}$	0.00159
$x_{B,R}$	0.210	$H_{L,R}$	1.966 $\times 10^2$ $[J/mol]$
$x_{B,0}$	0.176	$H_{L,0}$	2.047 $\times 10^2$ $[J/mol]$
$x_{B,N+1}$	0.800	$H_{L,N+1}$	3.880 $\times 10^2$ $[J/mol]$

The goal of the controller is to drive the system from the initial stable operating point to the desired operating point. The initial steady-state values for the manipulated inputs and the states of the CSTR, reboiler and condenser are given in Table 5.6 and in Table 5.7, respectively. Before proceeding with the control design, we note that via extensive simulation we have verified that the process exhibits two-time-scale behavior (see Figure 5.12) owing to the use of large recycle, D, relative to the feed input, F_0, which motivates defining $\epsilon = \tilde{F}_o/\tilde{D} = 0.056$. However, several of the process states exhibit dynamic behavior in both fast and slow time scales, and thus, the explicit separation of the process model states into fast and slow ones in a way that it is consistent with the standard singularly perturbed model form of Eq. 5.1 is not a feasible task in this particular application. For this reason, instead of separating the states into fast and slow ones, we divide the manipulated inputs into the ones that regulate critical fast states and the ones that regulate the process state in the slow time-scale. Specifically, we define the following dimensionless manipulated inputs, $u_1 = F/\tilde{F}$, $u_2 = \bar{V}/\tilde{V}$, $u_3 = B_t/\tilde{B}_t$, $u_4 = D/\tilde{D}$, $u_5 = Q_1/\tilde{Q}_1$, $u_6 = Q_2/\tilde{Q}_2$ and $u_7 = Q_3/\tilde{Q}_3$. Through extensive simulations, we found that the manipulated inputs, u_1, u_2, u_3 and u_4 can be used to control the liquid hold-ups (fast dynamics), and u_5, u_6 and u_7 can be used to control the process state in the slow time-scale; please see remark 5.2 below for a detailed discussion and simulations on this issue.

With respect to control design, we propose to design a control system that utilizes proportional control to compute the inputs associated with the fast dynamics and MPC to compute the inputs associated with the slow dynamics. Specifically, four different proportional controllers are used to regulate each of the flow rates, F, D, $\bar{V}$, and B with respect to the final steady-state input values in Table 5.4 and the

steady-state liquid holdups in Table 5.5:

$$u_1 = F/\tilde{F} = 1 - k_{c1}(\tilde{M}_R - M_R) \tag{5.28a}$$

$$u_2 = \bar{V}/\tilde{V} = 1 - k_{c2}(\tilde{M}_{N+1} - M_{N+1}) \tag{5.28b}$$

$$u_3 = B/\tilde{B} = 1 - k_{c3}(\tilde{M}_0 - M_0) \tag{5.28c}$$

$$u_4 = D/\tilde{D} = 1 - k_{c4}(\tilde{M}_0 - M_0) \tag{5.28d}$$

in which k_{c1}, k_{c2}, k_{c3} and k_{c4} are all equal to 0.0001. The controllers of Eq. 5.28 utilize feedback of the hold-ups that can be sampled fast and can stabilize the liquid hold-up levels of the CSTR, the reboiler and the condenser. Note that the effluent flow rate of the vapor mixture $\bar{V}$ is also regulated by a pressure valve. When the pressure inside the reboiler goes down, the rate of liquid evaporation rises and therefore, the flow rate $\bar{V}$ goes up. In this example, we do not consider the pressure effect in the process model and assume that $\bar{V}$ is controllable and is directly related to the liquid holdup of the reboiler.

The control of the slow dynamics involves the application of MPC. Three MPC strategies are applied and compared in this study. Specifically, a centralized LMPC which calculates all the inputs in one optimization problem, a sequential distributed MPC (DMPC) in which the control inputs are calculated by distributed optimization problems in sequence, and an iterative DMPC in which the control inputs are evaluated by parallel distributed optimization problems solved in an iterative fashion. For more discussion on the sequential and iterative DMPC, please refer to [63]. We define the term *evaluation number* to indicate the number of evaluations for the optimization problem solved in each controller at each sampling time. For instance,

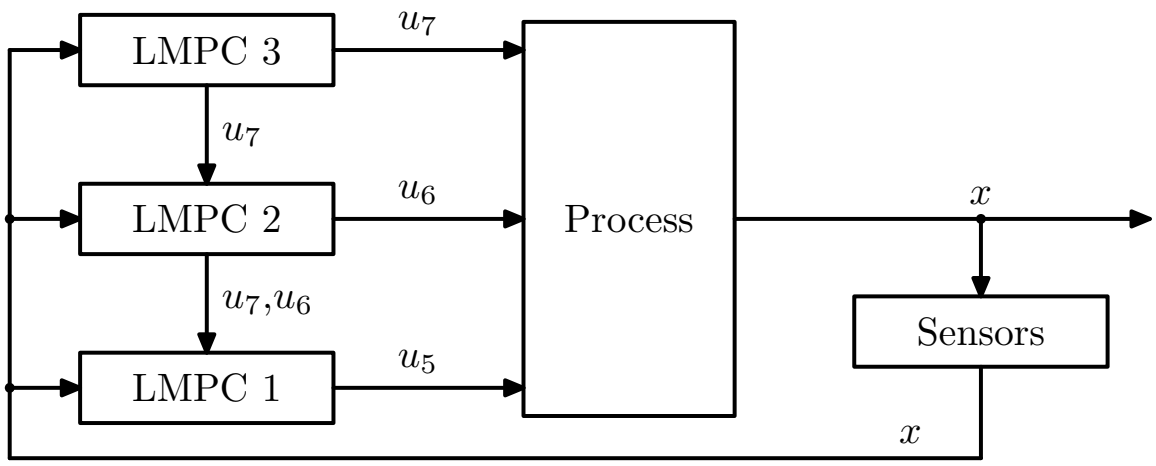


Figure 5.3: Sequential DMPC architecture manipulating u_5, u_6 and u_7.

an evaluation number of one implies that there is no information sharing between the controllers, and each one of them returns the manipulated input values after the end of one evaluation.

Three distributed LMPCs are designed for both DMPC control strategies. In both strategies, LMPC 1 determines the input Q_1, LMPC 2 determines the input Q_2, and LMPC 3 determines the input Q_3. Schematics of the sequential and iterative DMPC architectures for this process are shown in Figs. 5.3 and 5.4, respectively. In order to formulate each of the optimization problems of the DMPCs (see [63]), the following feedback laws are used as the reference control laws in the design of the three LMPCs:

$$u_5 = Q_1/\tilde{Q}_1 = 1 + k_{c5}(\tilde{T}_1 - T_1) \tag{5.29a}$$

$$u_6 = Q_2/\tilde{Q}_2 = 1 + k_{c6}(\tilde{T}_2 - T_2) \tag{5.29b}$$

$$u_7 = Q_3/\tilde{Q}_3 = 1 + k_{c7}(\tilde{T}_2 - T_3) \tag{5.29c}$$

where $k_{c5} = 0.008$, $k_{c6} = 0.0002$, $k_{c7} = 0.0002$, $\tilde{T}_1 = 360.25$, $\tilde{T}_2 = 367.97$ and $\tilde{T}_3 =$

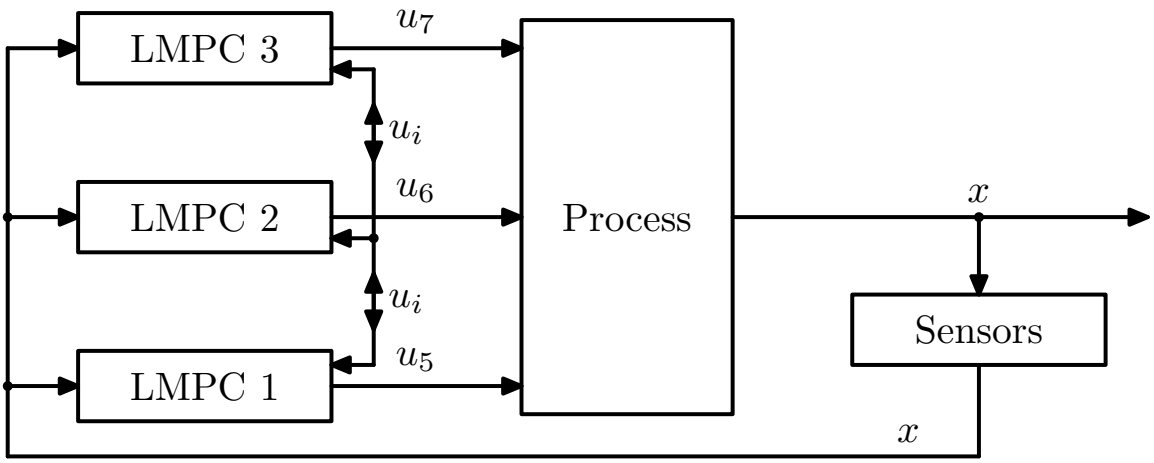


Figure 5.4: Iterative DMPC architecture manipulating u_5, u_6 and u_7.

421.72. In the design of the LMPCs, a quadratic Lyapunov function $V(x) = x^T Px$ where P is an identity matrix is used to put even weights on the different states. Through extensive simulations, we found that this Lyapunov function choice to be a good one in terms of control performance and ease of controller implementation. In the simulations, the inputs associated with the slow dynamics are subject to the following constraints:

$$0.9 \leq u_5 \leq 1.3, \quad 0.9 \leq u_6 \leq 1.2, \quad 0.9 \leq u_7 \leq 1.2.$$

5.4.2 Simulation results

The simulations were performed in Microsoft Visual Studio by a Core2 Quad Q6600 computer. The total process evaluation time for each run is 3000 seconds. Four different cases are studied here. The first one applies the centralized LMPC scheme. The second case is for the sequential DMPC approach. In the third and fourth case study,

the iterative DMPC scheme with one evaluation and two evaluations are used. Two different prediction horizons are used for each of the MPC methods, $N = 1$ and $N = 2$. Only the first input value from the output of the optimization problems is implemented following a receding horizon scheme. The sampling time of the optimization problems is $\Delta = 30\ s$, and as a result, the total number of sampling times along one simulation is 100. By assumption, all state measurements are available to the MPC controllers at each sampling time and are available continuously to the proportional controllers. The numerical method that is used to integrate the process is explicit Euler with a fixed time step of 0.1 s and the LMPC optimization problems are solved using the open source interior point optimizer Ipopt [101].

The cost function used in each MPC scheme is as follows:

$$J = \int_{t_k}^{t_{k+N}} \left[x^T(t)Q_c x(t) + U_2^T(t)R_{ci}U_2(t)\right] dt$$

where t_k is time when the controller is evaluated and $U_2^T = [u_5 - 1\ u_6 - 1\ u_7 - 1]$. The weighting matrix Q_c is a diagonal matrix with its diagonal element $Q_{c,i} = 1/x_{set,i}$, where $x_{set,i}$ is the steady state value of the corresponding state variable. The weighting matrix R_{c2} is also a diagonal matrix with $R_{c2} = diag([10000\,10000\,10000])$.

Figure 5.5 shows the trajectories of the Lyapunov function $V(x)$ under the different control schemes. Based on these trajectories, it can be seen that all MPC strategies stabilize the closed-loop system and give very close results in terms of trajectories of $V(x)$. The corresponding trajectories of the inputs Q_1, Q_2 and Q_3 (i.e., u_5, u_6 and u_7) are shown in Figs. 5.6, 5.7 and 5.8.

Next, we investigate the instantaneous closed-loop performance at each sampling time measured by $x^T(t_k)Q_c x(t_k) + \sum_{i=1}^{2} U_i^T(t_k)R_{ci}U_i(t_k)$, $k = 0, 1, \ldots$ under the centralized LMPC and the two DMPC schemes where $U_1^T = [u_1 - 1\ u_2 - 1\ u_3 - 1\ u_4 - 1]$ and

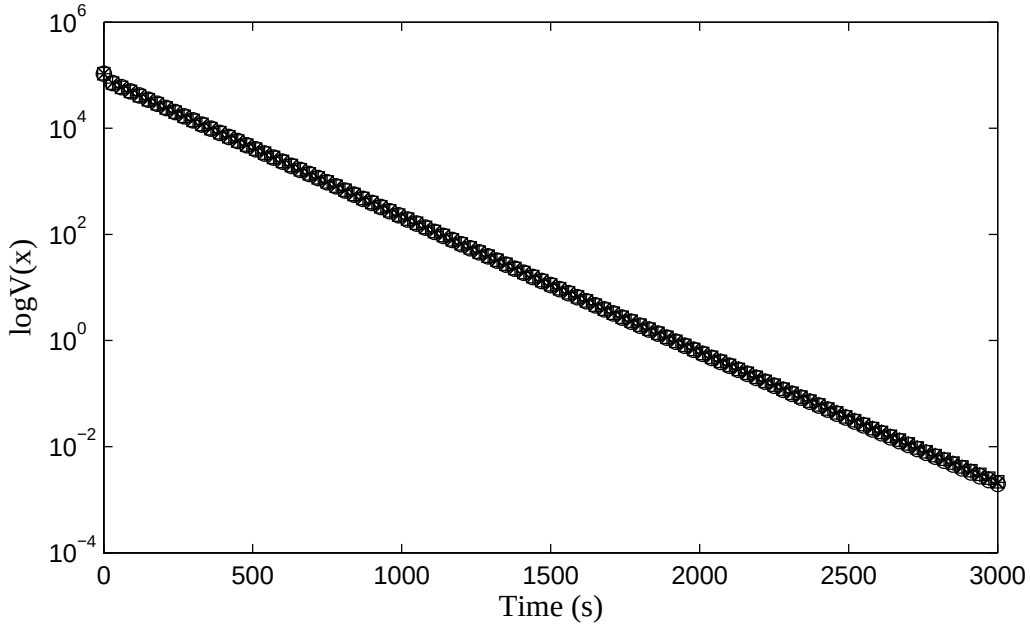


Figure 5.5: Trajectories of $V(x)$ under the centralized LMPC ($\circ$), the sequential DMPC ($*$), and the iterative DMPC with one evaluation ($\square$) and with two evaluations ($\times$).

$R_{c1} = diag([10000\,10000\,10000\,10000\,10000])$. The results are shown in Fig. 5.9. We note that in this cost we also include the control inputs used for the fast dynamics in order to have a comprehensive comparison. From Fig. 5.9, we see that as the simulation time approaches 1000 s, the instantaneous closed-loop performance given by the different control schemes is nearly the same. This is because under the different control schemes, the closed-loop system state is driven to the same desired steady-state. From Fig. 5.9 (especially from the first half of the simulation: $0 < t < 500$ s), we can also see that the centralized control scheme gives the best performance and as the iteration number increases, the performance given by the iterative DMPC converges to the one given by the centralized control scheme. This property of the iterative DMPC is not guaranteed for general nonlinear systems but it is found to hold for this specific simulation study.

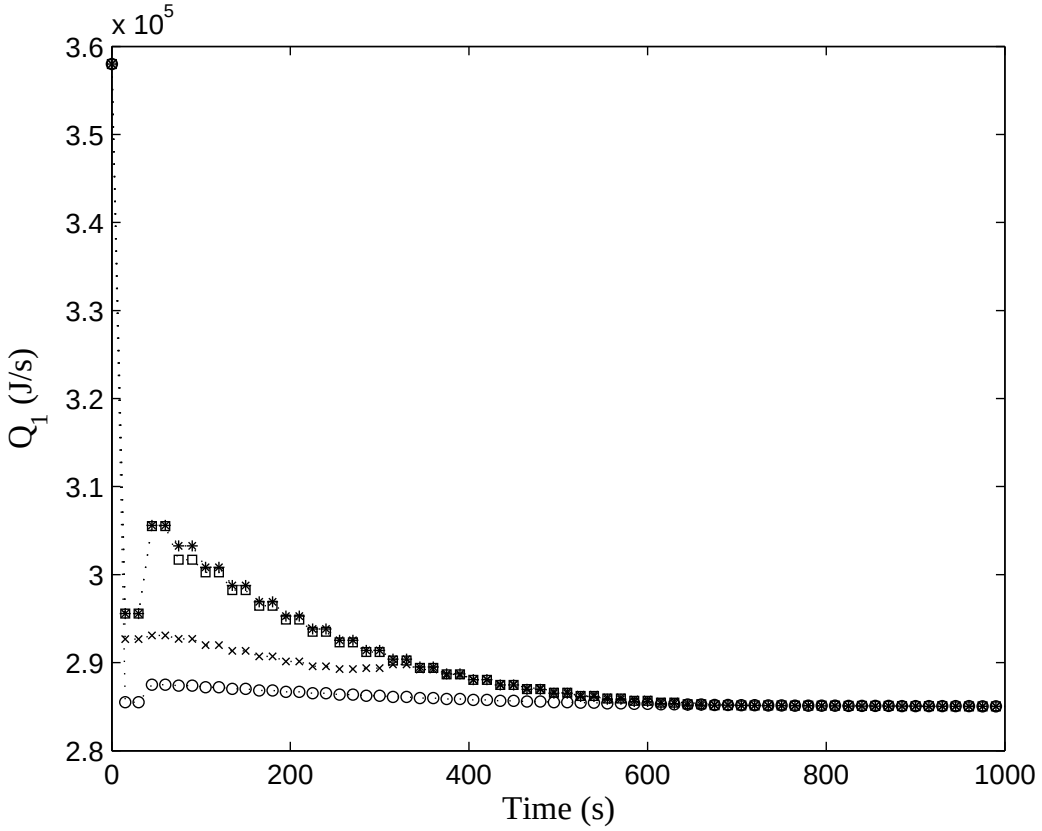


Figure 5.6: Trajectories of the input Q_1 (i.e., u_5) under the centralized LMPC ($\circ$), the sequential DMPC ($*$), and the iterative DMPC with one evaluation ($\square$) and with two evaluations ($\times$).

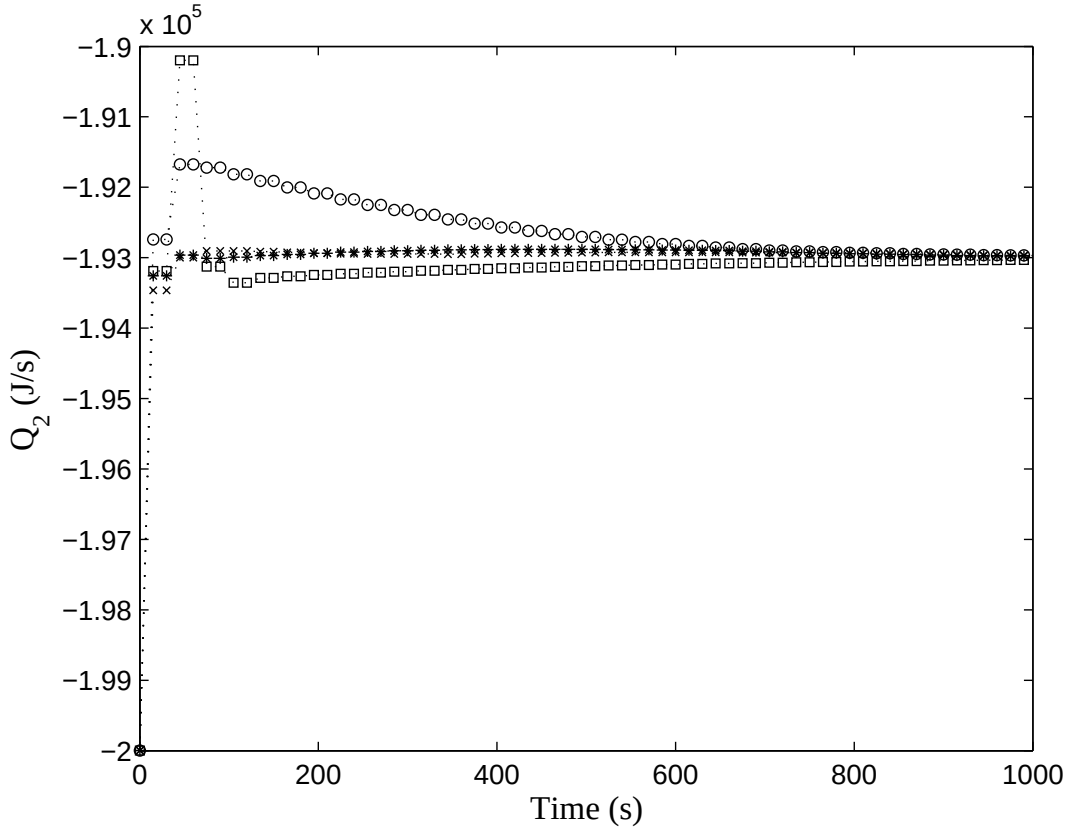


Figure 5.7: Trajectories of the input Q_2 (i.e., u_6) under the centralized LMPC ($\circ$), the sequential DMPC ($*$), and the iterative DMPC with one evaluation ($\square$) and with two evaluations ($\times$).

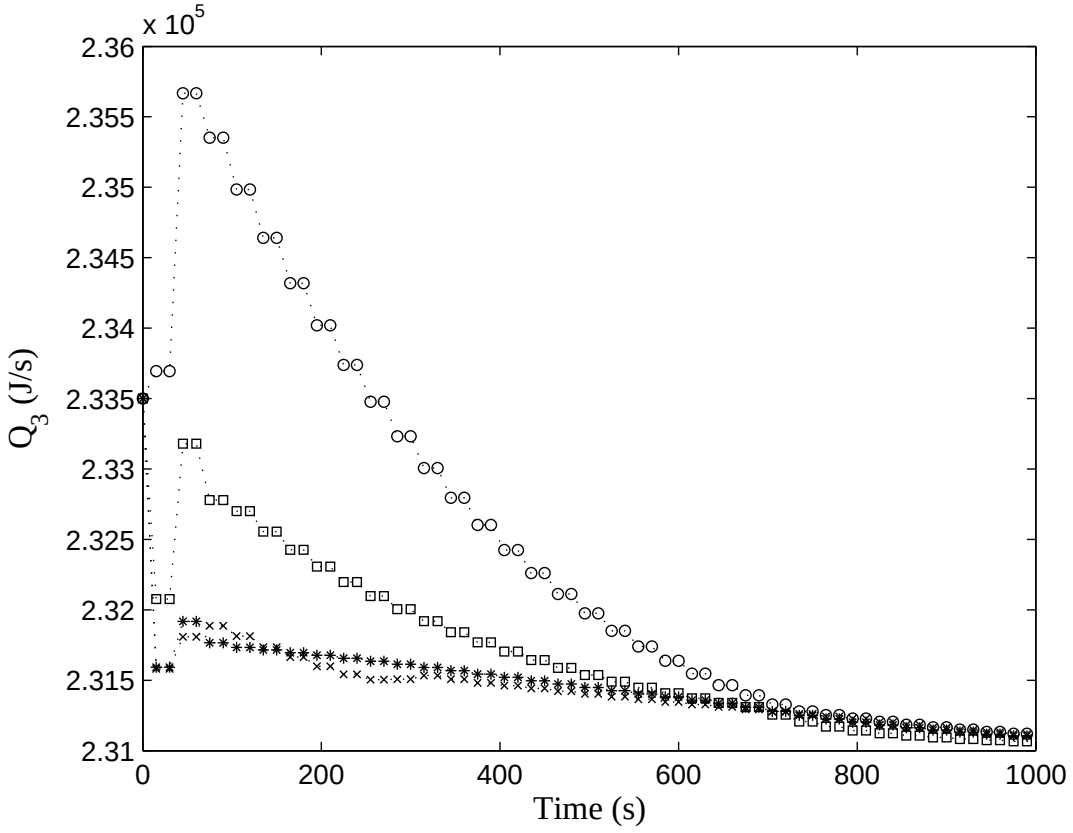


Figure 5.8: Trajectories of the input Q_3 (i.e., u_7) under the centralized LMPC ($\circ$), the sequential DMPC ($*$), and the iterative DMPC with one evaluation ($\square$) and with two evaluations ($\times$).

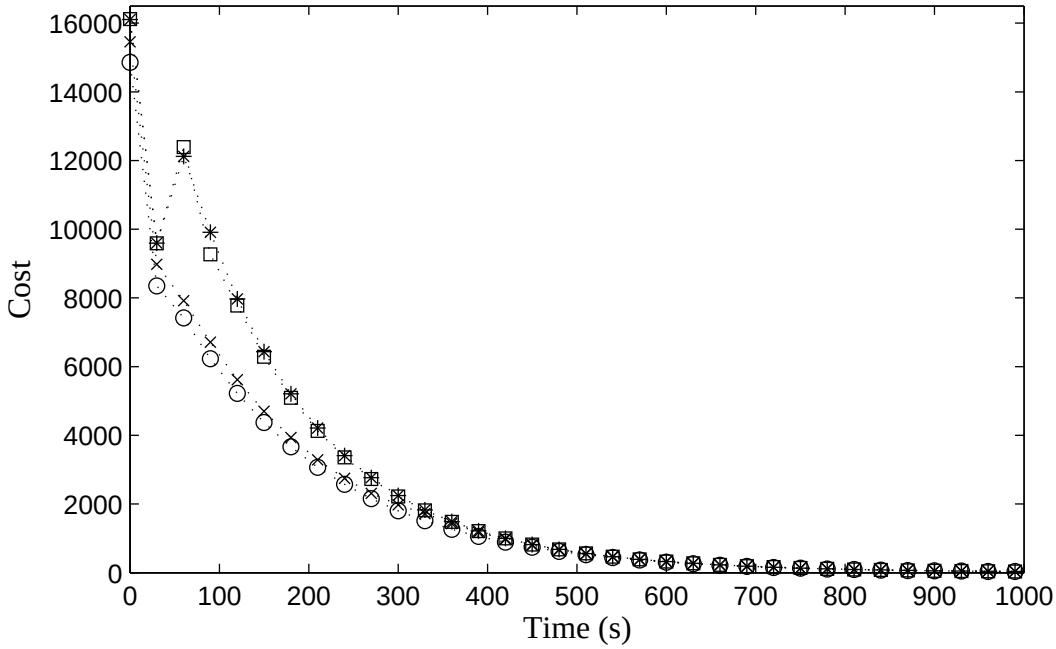


Figure 5.9: The costs of the closed-loop system under the centralized LMPC ($\circ$), the sequential DMPC ($*$), and the iterative DMPC with one evaluation ($\square$) and with two evaluations ($\times$).

In the last set of simulations, attention is given to the evaluation time of the three MPC schemes, as shown in Fig. 5.10 ($N = 1$) and in Fig. 5.11 ($N = 2$). Because of the different structure of the two DMPC architectures, it is important to note that the total evaluation time required for the sequential DMPC in one sampling time is the sum of the evaluation times of the three LMPCs; on the other hand, the total evaluation time required for the iterative DMPC with one evaluation in one sampling time is the maximum evaluation time among all the three LMPCs. Both Figures clearly demonstrate that the iterative DMPC with one evaluation has the smallest total evaluation time compared with the other MPC schemes, and the sequential DMPC requires more evaluation time than the centralized LMPC in this set of simulations. In Fig. 5.10, the average evaluation time of the iterative DMPC with one evaluation over the entire simulation is 1.70 s, which is about 70% of the average time needed for the centralized LMPC and 2.6 times faster than the average time needed for the sequential DMPC. Similarly, in Fig. 5.11, the average total evaluation time of the iterative DMPC with one evaluation along the simulation is 4.25 seconds, which is about 63% of the average time needed for the centralized LMPC and 2.3 times faster than the average time needed for the sequential DMPC.

Remark 5.2 *To justify the use of u_1, u_2, u_3 and u_4 to control the liquid hold-ups that exhibit fast dynamic behavior, we carried out a set of simulations of the closed-loop system under the fast proportional controls used to manipulate u_1, u_2, u_3 and u_4 and the centralized MPC used to manipulate u_5, u_6 and u_7. Figure 5.12 shows the evolution of a measure of the liquid hold-ups ($\rho_f(t)$) which exhibit fast dynamics initially (fast time-scale) and the evolution of a measure of the compositions ($\rho_s(t)$) which exhibit dynamics in a slow time-scale, thereby confirming our choice to use fast acting feedback to regulate the liquid hold-ups. In Figure 5.12, the measures $\rho_s(t)$ and*

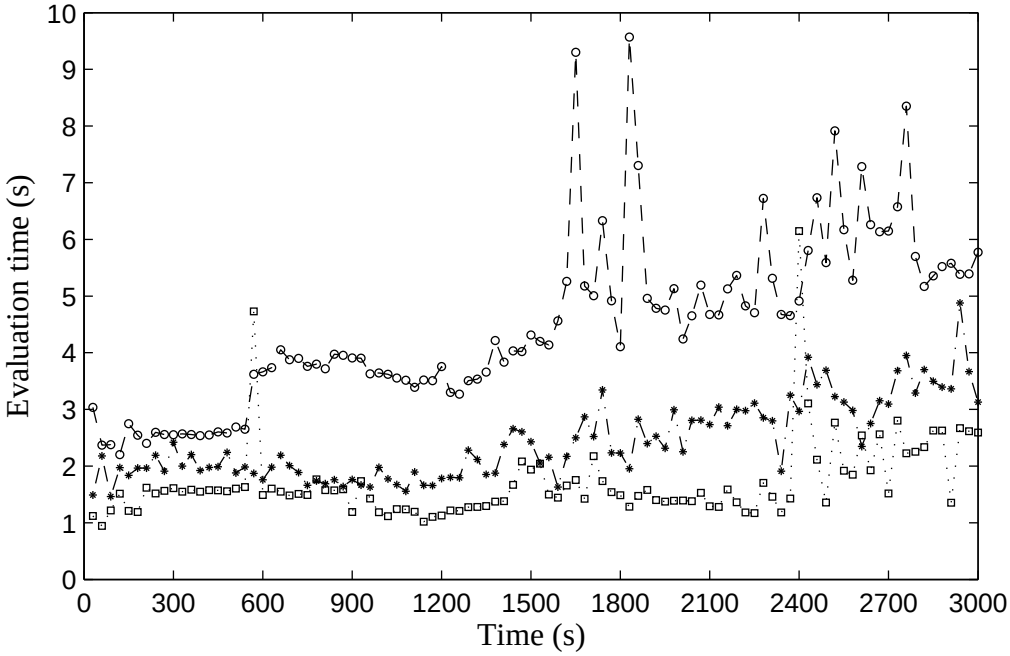


Figure 5.10: The total evaluation time needed for each evaluation of each MPC method. Centralized LMPC (solid line with $*$), sequential DMPC (dashed line with $\circ$), and iterative DMPC with one evaluation (dotted line with $\square$). The prediction horizon $N = 1$.

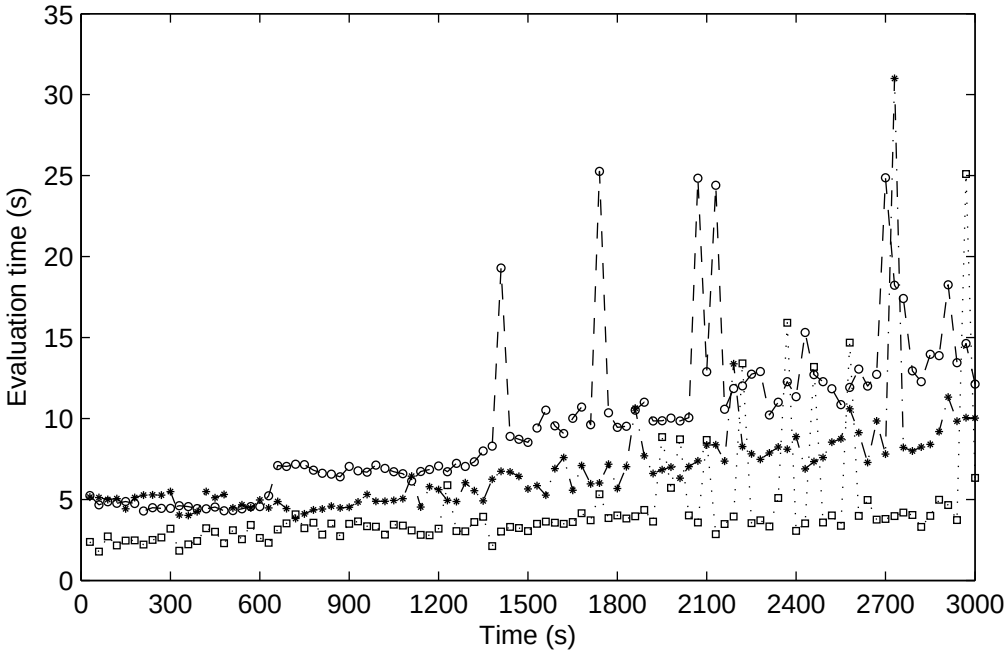


Figure 5.11: The total evaluation time needed for each evaluation of each MPC method. Centralized LMPC (solid line with $*$), sequential DMPC (dashed line with $\circ$), and iterative DMPC with one evaluation (dotted line with $\square$). The prediction horizon $N = 2$.

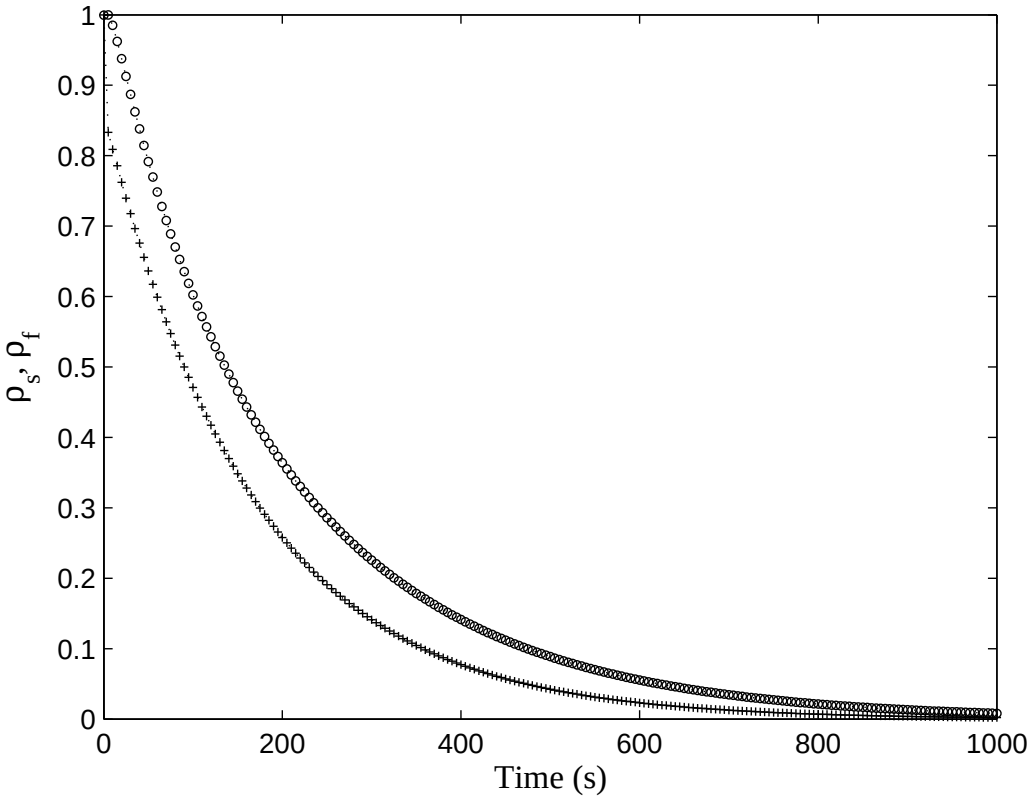


Figure 5.12: Evolution of a measure of the liquid hold-ups ($\rho_f(t)$; + symbol) and evolution of a measure of the compositions ($\rho_s(t)$; o symbol).

$\rho_f(t)$ are defined as follows:

$$\rho_s(t) = \frac{\sum_{i}^{A,B,C} \sum_{j}^{15} (x_{i,j} - \tilde{x}_{i,j})^2 + \sum_{i}^{A,B,C} (x_{i,R} - \tilde{x}_{i,R})^2}{\max((x_{i,j} - \tilde{x}_{i,j})^2, (x_{i,R} - \tilde{x}_{i,R})^2)} \tag{5.30a}$$

$$\rho_f(t) = \frac{(M_o - \tilde{M}_o)^2 + (M_{N+1} - \tilde{M}_{N+1})^2 + (M_R - \tilde{M}_R)^2}{\max((M_o - \tilde{M}_o)^2, (M_{N+1} - \tilde{M}_{N+1})^2, (M_R - \tilde{M}_R)^2)} \tag{5.30b}$$

5.5 Conclusions

This chapter focused on model predictive control of a class of nonlinear singularly perturbed systems. The motivation for this work is provided by broad classes of large-scale process networks that involve coupled variables that evolve in disparate (fast and slow) time scales. For such process networks, direct application of model predictive control to compute the control actions for all manipulated inputs leads to very high-order optimization problems that may not be solvable in real-time. Instead, we proposed a control system using multirate sampling (i.e., fast sampling of easy-to-measure fast-evolving variables and slow sampling of slow-evolving variables) and consisting of an explicit feedback controller that stabilizes the fast dynamics and a model predictive controller that stabilizes the slow dynamics and enforces desired performance objectives in the slow subsystem was proposed. In this way, the model predictive controller solves an optimization problem with a substantially smaller number of decision variables, and thus, it requires less computational time. Sufficient conditions under which the closed-loop system stability, accounting for multirate sampling and sample-and-hold implementation of the predictive controller, is guaranteed were provided. The applicability and effectiveness of the proposed control system was illustrated via a large-scale nonlinear reactor-separator process network which exhibits two-time-scale behavior and the computational effectiveness of distributed predictive control implementation was demonstrated.

Chapter 6

Composite Fast-Slow MPC Design for Nonlinear Singularly Perturbed Systems

6.1 Introduction

In Chapter 5, we studied MPC for nonlinear singularly perturbed systems where MPC is used only in the slow time-scale and the fast dynamics are assumed to be stabilizable by a "fast" explicit controller.

This chapter focuses on model predictive control of nonlinear singularly perturbed systems in standard form where the separation between the fast and slow state variables is explicit [11]. Specifically, a composite control system comprised of a "fast" MPC acting to regulate the fast dynamics and a "slow" MPC acting to regulate the slow dynamics is designed. The composite MPC system uses multirate sampling of the plant state measurements, i.e., fast sampling of the fast state variables is used

in the fast MPC and slow-sampling of the slow state variables is used in the slow MPC as well as in the fast MPC. Using singular perturbation theory, the stability and optimality of the closed-loop nonlinear singularly perturbed system are analyzed. The proposed fast-slow MPC design does not require communication between the two MPCs, and thus, it can be classified as decentralized in nature. A chemical process example which exhibits two-time-scale behavior is used to demonstrate the structure and implementation of the fast-slow MPC architecture in a practical setting. Extensive simulations are carried out to assess the performance and computational efficiency of the fast-slow MPC system.

6.2 Preliminaries

6.2.1 Notation

The operator $|\cdot|$ is used to denote Euclidean norm of a vector and the symbol Ω_r is used to denote the set $\Omega_r := \{x \in R^{n_x} : V(x) \leq r\}$ where V is a positive definite scalar function. For any measurable (with respect to the Lebesgue measure) function $w : R_{\geq 0} \to R^l$, $||w||$ denotes ess.sup.$|w(t)|$, $t \geq 0$. A function $\gamma : R_{\geq 0} \to R_{\geq 0}$ is said to be of class K if it is continuous, nondecreasing, and is zero at zero. A function $\beta : R_{\geq 0} \times R_{\geq 0} \to R_{\geq 0}$ is said to be of class KL if, for each fixed t, the function $\beta(\cdot, t)$ is of class K and, for each fixed s, the function $\beta(s, \cdot)$ is nonincreasing and tends to zero at infinity. The symbol $diag(v)$ denotes a matrix whose diagonal elements are the elements of vector v and all the other elements are zeros.

6.2.2 Class of nonlinear singularly perturbed systems

In this section, we focus on nonlinear singularly perturbed systems in standard form with the following state-space description:

$$\begin{aligned} \dot{x} &= f(x, z, \epsilon, u_s, w), \quad x(0) = x_0 \\ \epsilon \dot{z} &= g(x, z, \epsilon, u_f, w), \quad z(0) = z_0 \end{aligned} \tag{6.1}$$

where $x \in R^n$ and $z \in R^m$ denote the vector of state variables, ϵ is a small positive parameter, $w \in R^l$ denotes the vector of disturbances and $u_s \in U \subset R^p$ and $u_f \in V \subset R^q$ are two sets of manipulated inputs. The sets U and V are nonempty convex sets which are defined as follows:

$$\begin{aligned} U &:= \{u_{s,i}(t) : |u_{s,i}(t)| \leq u_{s,i}^{\max},\ i \in [1, p]\} \\ V &:= \{u_{f,j}(t) : |u_{f,j}(t)| \leq u_{f,j}^{\max},\ j \in [1, q]\} \end{aligned} \tag{6.2}$$

where $u_{s,i}^{\max}$ and $u_{f,j}^{\max}$ are positive real numbers, specifying the input constraints. The disturbance vector is assumed to be absolutely continuous and bounded, i.e., $W := \{w(t) \in R^l : |w(t)| \leq \theta\}$ where θ is a positive real number. Since the small parameter ϵ multiplies the time derivative of the vector z in the system of Eq. 6.1, the separation of the slow and fast variables in Eq. 6.1 is explicit, and thus, we will refer to the vector x as the slow states and to the vector z as the fast states. We assume that the vector fields f and g are sufficiently smooth in $R^n \times R^m \times [0, \bar{\epsilon}) \times R^p \times R^l$ and $R^n \times R^m \times [0, \bar{\epsilon}) \times R^q \times R^l$, respectively, for some $\bar{\epsilon} > 0$, and that the origin is an equilibrium point of the unforced nominal system (i.e., system of Eq. 6.1 with $u_s = 0$, $u_f = 0$ and $w = 0$).

With respect to the control problem formulation, we assume that the fast states z are sampled continuously and their measurements are available for all time t (for

example, variables for which fast sampling is possible usually include temperature, pressure and hold-ups) while the slow states x are sampled synchronously and are available at time instants indicated by the time sequence $\{t_{k\geq 0}\}$ with $t_k = t_0 + k\Delta,\ k = 0, 1, \ldots$ where t_0 is the initial time and Δ is the measurement sampling time of the slow states (for example, slowly sampled variables usually involve species concentrations). The set of manipulated inputs u_f is responsible for stabilizing the fast dynamics of Eq. 6.1 and for this set the control action is assumed to be computed every Δ_f, while the set of manipulated inputs u_s is evaluated every Δ_s and is responsible for stabilizing the slow dynamics and enforcing a desired level of optimal closed-loop performance. The relationship between Δ, Δ_s and Δ_f will be discussed below.

6.2.3 Two-time-scale decomposition

The explicit separation of slow and fast variables in the system of Eq. 6.1 allows decomposing it into two separate reduced-order systems evolving in different time-scales. To proceed with such a two-time-scale decomposition and in order to simplify the notation of the subsequent development, we will first address the issue of controlling the fast dynamics. Since there is no assumption that the fast dynamics of Eq. 6.1 are asymptotically stable, we assume the existence of a "fast" model predictive control law u_f that renders the fast dynamics asymptotically stable in a sense to be made precise in Assumption 6.2 below. In contrast to previous approaches to u_f controller design (e.g., [54]), we focus on the design of a feedback control law that does not modify the open-loop equilibrium manifold for the fast dynamics. This is in contrast to our previous work [12] (also Chapter 5) where the fast feedback u_f modifies the equilibrium manifold for the fast dynamics in the closed-loop system. This implies that when we set $\epsilon = 0$ in the singularly perturbed system of Eq. 6.1 to

derive the slow subsystem $u_f = 0$, and the resulting slow subsystem takes the form:

$$\frac{dx}{dt} = f(x, z, 0, u_s, w) \tag{6.3a}$$

$$0 = g(x, z, 0, 0, w) \tag{6.3b}$$

Assumption 6.1 below is a standard requirement in singularly perturbation theory (please see, for example, [54]) and it is made to ensure that the system of Eq. 6.1 has an isolated equilibrium manifold for the fast dynamics. On this manifold, z can be expressed in terms of x and w using an algebraic expression; note that $g(x, z, 0, 0, w)$ is in this case independent of the expression of the "fast" feedback control law u_f. This assumption does not pose any significant limitations in practical applications but it is a necessary one in the singular perturbation framework to construct a well-defined slow subsystem.

Assumption 6.1 *The equation* $g(x, z, 0, 0, w) = 0$ *possesses a unique root*

$$z = \tilde{g}(x, w) \tag{6.4}$$

with the properties that $\tilde{g} : R^n \times R^l \rightarrow R^m$ *and its partial derivatives* $\dfrac{\partial \tilde{g}}{\partial x}$, $\dfrac{\partial \tilde{g}}{\partial w}$ *are sufficiently smooth.*

Using $z = \tilde{g}(x, w)$, we can re-write Eq. 6.3 as follows:

$$\frac{dx}{dt} \;=\; f(x, \tilde{g}(x, w), 0, u_s, w) =: f_s(x, u_s, w) \tag{6.5}$$

We will refer to the subsystem of Eq. 6.5 as the slow subsystem.

Introducing the fast time scale $\tau = \dfrac{t}{\epsilon}$ and the deviation variable $y = z - \tilde{g}(x, w)$, we can rewrite the nonlinear singularly perturbed system of Eq. 6.1 as follows:

$$\begin{aligned} \frac{dx}{d\tau} &= \epsilon f(x, y + \tilde{g}(x, w), \epsilon, u_s, w) \\ \frac{dy}{d\tau} &= g(x, y + \tilde{g}(x, w), \epsilon, u_f, w) - \epsilon \frac{\partial \tilde{g}}{\partial w} \dot{w} \\ &\quad -\epsilon \frac{\partial \tilde{g}}{\partial x} f(x, y + \tilde{g}(x, w), \epsilon, u_s, w) \end{aligned} \tag{6.6}$$

Setting $\epsilon = 0$, we obtain the following fast subsystem:

$$\frac{dy}{d\tau} = g(x, y + \tilde{g}(x, w), 0, u_f, w) \tag{6.7}$$

where x and w can be considered as "frozen" to their initial values.

6.2.4 Slow-fast subsystem stabilizability assumptions

We assume that there exists a Lyapunov-based locally Lipschitz control law $h_s(x) = [h_{s1}(x) \ \ldots \ h_{sp}(x)]^T$ with $u_{s,i} = h_{si}(x)$, $i = 1, \ldots, p$, which renders the origin of the nominal closed-loop slow subsystem of Eq. 6.5 asymptotically stable while satisfying the input constraints for all the states x inside a given stability region. Such an explicit controller can be designed using Lyapunov-based control techniques [60, 18]. Using converse Lyapunov theorems [75, 60, 18], this assumption implies that there exist functions $\alpha_{s,i}(\cdot)$, $i = 1, 2, 3$ of class K and a continuously differentiable Lyapunov function $V_s(x)$ for the nominal closed-loop slow subsystem that satisfy the following inequalities:

$$\begin{gathered} \alpha_{s,1}(|x|) \leq V_s(x) \leq \alpha_{s,2}(|x|) \\ \frac{\partial V_s(x)}{\partial x}(f_s(x, h_s(x), 0)) \leq -\alpha_{s,3}(|x|) \\ h_s(x) \in U \end{gathered} \tag{6.8}$$

for all $x \in D_s \subseteq R^n$ where D_s is an open neighborhood of the origin. We denote the region $\Omega_{\rho^s} \subseteq D_s$ as the stability region of the closed-loop slow subsystem under the Lyapunov-based controller $h_s(x)$. By continuity, the smoothness property assumed for the vector fields $f_s(x, u_s, w)$ and taking into account that the manipulated inputs $u_{s,i}$, $i = 1, \ldots, p$, and the disturbance w are bounded in convex sets, there exists a positive constant M_s such that

$$|f_s(x, u_s, w)| \leq M_s \tag{6.9}$$

for all $x \in \Omega_{\rho^s}$, $u_s \in U$, and $w \in W$. In addition, by the continuous differentiable property of the Lyapunov function $V_s(x)$ and the smoothness property assumed for the vector field $f_s(x, u_s, w)$, there exist positive constants L_x and L_{w_s} such that

$$\begin{aligned} |\frac{\partial V_s}{\partial x} f_s(x, u_s, w) - \frac{\partial V_s}{\partial x} f_s(x', u_s, w)| &\leq L_x |x - x'| \\ |\frac{\partial V_s}{\partial x} f_s(x, u_s, w) - \frac{\partial V_s}{\partial x} f_s(x, u_s, w')| &\leq L_{w_s} |w - w'| \end{aligned} \tag{6.10}$$

for all $x, x' \in \Omega_{\rho^s}$, $u_s \in U$, and $w, w' \in W$.

Assumption 6.2 *There exists a feedback control law $u_f = p(x)y \in V$ where $p(x)$ is a sufficiently smooth vector function of its argument, such that the origin of the closed-loop fast subsystem:*

$$\frac{dy}{d\tau} = g(x, y + \tilde{g}(x, w), 0, p(x)y, w) \tag{6.11}$$

is globally asymptotically stable, uniformly in $x \in R^n$ and $w \in R^l$, in the sense that there exists a class KL function β_y such that for any $y(0) \in R^m$:

$$|y(t)| \leq \beta_y(|y(0)|, \frac{t}{\epsilon}) \tag{6.12}$$

for $t \geq 0$.

This assumption implies that there exist functions $\alpha_{f,i}(\cdot),\ i = 1, 2, 3$ of class K and a continuously differentiable Lyapunov function $V_f(y)$ for the nominal closed-loop fast subsystem that satisfy the following inequalities:

$$\begin{gathered} \alpha_{f,1}(|y|) \leq V_f(x) \leq \alpha_{f,2}(|y|) \\ \frac{\partial V_f(x)}{\partial y}(g(x, y + \tilde{g}(x, w), 0, p(x)y, w)) \leq -\alpha_{f,3}(|y|) \\ p(x)y \in V \end{gathered} \tag{6.13}$$

for all $y \in D_f \subseteq R^m$ where D_f is an open neighborhood of the origin. We denote the region $\Omega_{\rho^f} \subseteq D_f$ as the stability region of the closed-loop fast subsystem under the nonlinear controller $p(x)y$.

By continuity, the smoothness property assumed for the vector fields $g(x, y, 0, u_f, w)$ and taking into account that the manipulated inputs $u_{f,j},\ j = 1, \ldots, q$, and the disturbance w are bounded in convex sets, there exists a positive constant M_f such that

$$|g(x, y, 0, u_f, w)| \leq M_f \tag{6.14}$$

for all $y \in \Omega_{\rho^f},\ u_f \in V$, and $w \in W$. In addition, by the continuous differentiable property of the Lyapunov function $V_f(x)$ and the smoothness property assumed for the vector field $g(x, y, 0, u_f, w)$, there exist positive constants L_y and L_{w_f} such that

$$\begin{gathered} |\frac{\partial V_f}{\partial y} g(x, y, 0, u_f, w) - \frac{\partial V_f}{\partial y} g(x, y', 0, u_f, w)| \leq L_y |y - y'| \\ |\frac{\partial V_f}{\partial y} g(x, y, 0, u_f, w) - \frac{\partial V_f}{\partial y} g(x, y, 0, u_f, w')| \leq L_{w_f} |w - w'| \end{gathered} \tag{6.15}$$

for all $y, y' \in \Omega_{\rho^f},\ u_f \in V$, and $w, w' \in W$.

6.3 Fast-Slow MPC design

The singular perturbation framework of Eq. 6.1 can be used to develop composite control systems where an MPC is used in the fast time scale and another MPC is used in the slow time-scale. A schematic of the proposed composite fast-slow MPC architecture is shown in Fig. 6.1. In this case, a convenient way from a control problem formulation point of view is to design a fast-MPC that uses feedback of the deviation variable y in which case u_f is only active in the boundary layer (fast motion of the fast dynamics) and becomes nearly zero in the slow time-scale. In this case, there is no need for communication between the fast MPC and the slow MPC (please see Remark 6.2 below). Specifically, referring to the singularly perturbed system of Eq. 6.6, the cost can be defined as:

$$\begin{aligned} J &= J_s + J_f \\ &= \int_0^{t_s} \left[x^T(\tilde{\tau})Q_s x(\tilde{\tau}) + u_s^T(\tilde{\tau})R_s u_s(\tilde{\tau})\right] d\tilde{\tau} \\ &\qquad + \int_0^{t_f} \left[y^T(\tilde{\tau})Q_f y(\tilde{\tau}) + u_f^T(\tilde{\tau})R_f u_f(\tilde{\tau})\right] d\tilde{\tau} \end{aligned} \tag{6.16}$$

where Q_s, Q_f, R_s, R_f are positive definite weighting matrices, and t_s and t_f are the prediction horizons for the parts of the cost focusing on the slow and fast subsystems, respectively.

6.3.1 Lyapunov-based slow MPC formulation

Referring to the slow subsystem of Eq. 6.5, we use the LMPC proposed in [79] which guarantees practical stability of the closed-loop system and allows for an explicit characterization of the stability region to compute u_s. The LMPC is based on the Lyapunov-based controller $h_s(x)$. The controller $h_s(x)$ is used to define a stability

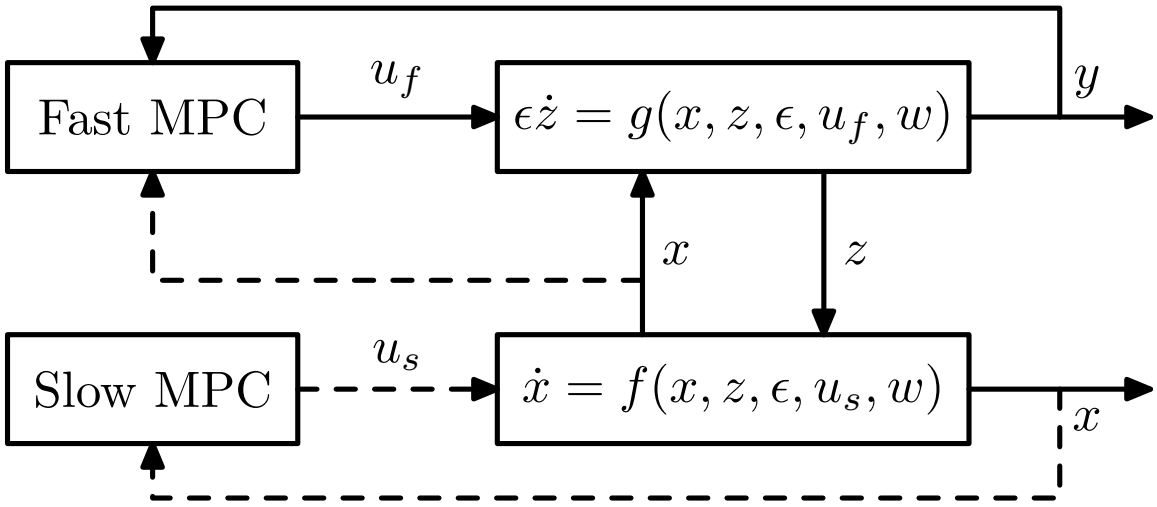


Figure 6.1: A schematic of the composite control system using MPC in both the fast and slow time-scales.

constraint for the LMPC controller which guarantees that the LMPC controller inherits the stability and robustness properties of the Lyapunov-based controller $h_s(x)$. The LMPC controller is based on the following optimization problem:

$$\min_{u_s \in S(\Delta_s)} \int_0^{N_s \Delta_s} [\tilde{x}^T(\tilde{\tau}) Q_s \tilde{x}(\tilde{\tau}) + u_s^T(\tilde{\tau}) R_s u_s(\tilde{\tau})] d\tilde{\tau} \tag{6.17a}$$

$$\text{s.t. } \dot{\tilde{x}}(\tilde{\tau}) = f_s(\tilde{x}(\tilde{\tau}), u_s, 0) \tag{6.17b}$$

$$u_s(\tilde{\tau}) \in U \tag{6.17c}$$

$$\tilde{x}(0) = x(t_k) \tag{6.17d}$$

$$\frac{\partial V_s(x)}{\partial x} f_s(x(t_k), u_s(0), 0) \leq \frac{\partial V_s(x)}{\partial x} f_s(x(t_k), h_s(x(t_k)), 0) \tag{6.17e}$$

where $S(\Delta_s)$ is the family of piece-wise constant functions with sampling period Δ_s, N_s is the prediction horizon, $x(t_k)$ is the state measurement obtained at t_k, $\tilde{x}$ is the predicted trajectory of the nominal system with u_s, the input trajectory computed by the LMPC of Eq. 6.17. The optimal solution to this optimization problem is denoted by $u_s^*(\tilde{\tau}|t_k)$, and is defined for $\tilde{\tau} \in [0, N_s\Delta_s)$. Note that in the MPC of Eq. 6.17, the control action is calculated every Δ_s which is the sampling interval of the slow states (i.e., $\Delta = \Delta_s$).

The optimization problem of Eq. 6.17 does not depend on the uncertainty and guarantees that the system in closed-loop with the LMPC controller of Eq. 6.17 maintains the stability properties of the Lyapunov-based controller. The constraint of Eq. 6.17e guarantees that the value of the time derivative of the Lyapunov function at the initial evaluation time of the LMPC is lower or equal to the value obtained if only the Lyapunov-based controller $h_s(x)$ is implemented in the closed-loop system in a sample-and-hold fashion. This is the constraint that allows proving that the LMPC inherits the stability and robustness properties of the Lyapunov-based controller. The manipulated inputs of the closed-loop slow subsystem under the LMPC controller are defined as follows

$$u_s(t) = u_s^*(t - t_k|t_k), \ \forall t \in [t_k, t_{k+1}). \tag{6.18}$$

The main property of the LMPC controller is that the origin of the closed-loop system is practically stable for all initial states inside the stability region Ω_{ρ^s} for a sufficient small sampling time Δ_s and disturbance upper bound θ. The main advantage of LMPC approaches with respect to Lyapunov-based control is that optimality considerations can be explicitly taken into account (as well as constraints on the inputs and the states [79]) in the computation of the control action within an online optimization framework.

Proposition 6.1 (c.f. [79, 80]) *Consider the slow subsystem of Eq. 6.5 in closed-loop under the LMPC of Eq. 6.18 based on a Lyapunov-based controller $h_s(x)$ that satisfies the conditions of Eq. 6.8. Let $\epsilon_{w_s} > 0$, $\Delta_s > 0$ and $\rho^s > \rho^s_s > 0$, $\theta > 0$ satisfy the following constraint:*

$$-\alpha_{s,3}(\alpha_{s,2}^{-1}(\rho^s_s)) + L_x M_s \Delta_s + L_{w_s}\theta \leq -\epsilon_{w_s}/\Delta_s. \tag{6.19}$$

There exists a class KL function β_x and a class K function γ_x such that if $x(0) \in \Omega_{\rho^s}$, then $x(t) \in \Omega_{\rho^s_s}$ for all $t \geq 0$ and

$$|x(t)| \leq \beta_x(|x(0)|, t) + \gamma_x(\rho^*_s) \tag{6.20}$$

with $\rho^*_s = \max\{V_s(x(t+\Delta_s)) : V_s(x(t)) \leq \rho^s_s\}$.

6.3.2 Lyapunov-based fast MPC formulation

Referring to the fast subsystem of Eq. 6.7, the fast MPC at time t_k is formulated as follows

$$\min_{u_f \in S(\Delta_f)} \int_0^{N_f\Delta_f} [\tilde{y}^T(\hat{\tau})Q_f\tilde{y}(\hat{\tau}) + u_f^T(\hat{\tau})R_f u_f(\hat{\tau})]d\hat{\tau} \tag{6.21a}$$

$$\text{s.t. } \frac{d\tilde{y}}{d\hat{\tau}} = g(x(t_k), \tilde{y} + \tilde{g}(x(t_k), 0), 0, u_f, 0) \tag{6.21b}$$

$$u_f \in V \tag{6.21c}$$

$$\tilde{y}(0) = y(t_k) \tag{6.21d}$$

$$\frac{\partial V_f(y)}{\partial y} g(x(t_k), y(t_k) + \tilde{g}(x(t_k), 0), 0, u_f, 0)$$

$$\leq \frac{\partial V_f(y)}{\partial y} g(x(t_k), y(t_k) + \tilde{g}(x(t_k), 0), 0, p(x(t_k))y(t_k), 0) \tag{6.21e}$$

where $S(\Delta_f)$ is the family of piece-wise constant functions with sampling period Δ_f, N_f is the prediction horizon of this MPC, $y(t_k)$ is the state measurement obtained at t_k, $\tilde{y}$ is the predicted trajectory of the nominal system with u_f, the input trajectory computed by the LMPC of Eq. 6.21. The optimal solution to this optimization problem is denoted by $u_f^*(\hat{\tau}|t_k)$, and is defined for $\hat{\tau} \in [0, N_f\Delta_f)$. Note that in the MPC of Eq. 6.21, the control action is calculated every Δ_f, the fast state z is available every Δ_f and the slow state x is available every Δ_s. Note also that we assume that Δ_s is an integer multiple of Δ_f.

The manipulated inputs of the closed-loop fast subsystem under the LMPC controller are defined as follows:

$$u_f(t) = u_f^*(t - t_k|t_k), \ \forall t \in [t_k, t_k + \Delta_f). \tag{6.22}$$

Proposition 6.2 (c.f. [79, 80]) *Consider the fast subsystem of Eq. 6.7 in closed-loop under the LMPC of Eq. 6.22 based on a nonlinear feedback control law $p(x)y$ that satisfies the conditions of Eq. 6.13. Let $\epsilon_{w_f} > 0$, $\Delta_f > 0$ and $\rho^f > \rho_s^f > 0$, $\theta > 0$ satisfy the following constraint:*

$$-\alpha_{f,3}(\alpha_{f,2}^{-1}(\rho_s^f)) + L_y M_f \Delta_f + L_{w_f}\theta \leq -\epsilon_{w_f}/\Delta_f. \tag{6.23}$$

Then, there exists a class KL function β_y and a class K function γ_y such that if $y(0) \in \Omega_{\rho^f}$, then $y(t) \in \Omega_{\rho_s^f}$ for all $t \geq 0$ and

$$|y(t)| \leq \beta_y(|y(0)|, \frac{t}{\epsilon}) + \gamma_y(\rho_f^*) \tag{6.24}$$

with $\rho_f^* = \max\{V_f(y(t+\Delta_f)) : V_f(y(t)) \leq \rho_s^f\}$, *uniformly in* $x \in \Omega_{\rho^s}$ *and* $w \in W$.

Remark 6.1 *The "fast" LMPC of Eq. 6.21 utilizes feedback of the "fast" state vector* z *which is obtained continuously (fast sampling of the fast states) as well as feedback of the slow states that is available at* $t_k, t_{k+1}, \ldots$ *where* $t_{k+1} = t_k + \Delta$. *Since the fast MPC has to compute its control action every* Δ_f, *the measurement* $x(t_k)$ *will be used in the controller of Eq. 6.21 for all fast sampling times,* Δ_f, *within* $[t_k, t_k + \Delta]$ *until a new measurement of the slow state vector is obtained. Since* x *is practically frozen in the boundary layer - time interval in which* z *changes a lot but* x *stays nearly fixed (depending on the value of* ϵ*), - the stability of the closed-loop fast subsystem uniformly in* x *and* w *can be proved; please see also [20] and next section.*

6.4 Stability analysis

The closed-loop stability of the system of Eq. 6.1 under the LMPCs of Eqs. 6.17 and 6.21 is established in the following theorem under appropriate conditions.

Theorem 6.1 *Consider the system of Eq. 6.1 in closed-loop with* u_f *and* u_s *computed by the LMPCs of Eqs. 6.22 and 6.18 based on controllers* $p(x)y$ *and* $h_s(\cdot)$ *that satisfies the conditions of Eqs. 6.13 and 6.8. Let also Assumptions 6.1 and 6.2 and the condition of Eqs. 6.19 and 6.23 hold. Then there exist functions* β_x *and* β_y *of class* KL, *a pair of positive real numbers* (δ, d) *and* $\epsilon^* > 0$ *such that if* $\max\{|x(0)|, |y(0)|, ||w||, ||\dot{w}||\} \leq \delta$ *and* $\epsilon \in (0, \epsilon^*]$, *then,*

$$\begin{aligned} |x(t)| &\leq \beta_x(|x(0)|, t) + \gamma_x(\rho_s^*) + d \\ |y(t)| &\leq \beta_y(|y(0)|, \frac{t}{\epsilon}) + \gamma_y(\rho_f^*) + d \end{aligned} \tag{6.25}$$

for all $t \geq 0$.

Proof: When $u_f = u_f^*$ and $u_s = u_s^*$ are determined by the LMPCs of Eqs. 6.22 and 6.18, respectively, the closed-loop system takes the following form:

$$\begin{aligned} \dot{x} &= f(x, z, \epsilon, u_s^*, w), \;\; x(0) = x_0 \\ \epsilon \dot{z} &= g(x, z, \epsilon, u_f^*, w), \;\; z(0) = z_0. \end{aligned} \tag{6.26}$$

We will first compute the slow and fast closed-loop subsystems. Setting $\epsilon = 0$ in Eq. 6.26 and taking advantage of the fact that $u_f^* = 0$ when $\epsilon = 0$, we obtain:

$$\begin{aligned} \frac{dx}{dt} &= f(x, z, 0, u_s^*, w) \\ 0 &= g(x, z, 0, 0, w). \end{aligned} \tag{6.27}$$

Using that the second equation has a unique, isolated solution $z = \tilde{g}(x, w)$ (Assumption 6.1), we can re-write 6.27 as follows:

$$\frac{dx}{dt} \;\; = \;\; f(x, \tilde{g}(x, w), 0, u_s^*, w) = f_s(x, u_s^*, w) \tag{6.28}$$

According to Proposition 6.1, the state $x(t)$ of the closed-loop slow subsystem of Eq. 6.28 starting from $x(0) \in \Omega_{\rho^s}$ stays in Ω_{ρ^s} (i.e., $x(t) \in \Omega_{\rho^s} \; \forall t \geq 0$) and satisfies the bound of Eq.6.20.

We now turn to the fast subsystem. Using $\tau = \dfrac{t}{\epsilon}$ and $y = z - \tilde{g}(x, w)$, the closed-loop system of Eq. 6.26 can be written as:

$$\begin{aligned} \frac{dx}{d\tau} &= \epsilon f(x, y + \tilde{g}(x, w), \epsilon, u_s^*, w) \\ \frac{dy}{d\tau} &= g(x, y + \tilde{g}(x, w), \epsilon, u_f^*, w) - \epsilon \frac{\partial \tilde{g}}{\partial w} \dot{w} - \epsilon \frac{\partial \tilde{g}}{\partial x} f(x, y + \tilde{g}(x, w), u_s^*, w) \end{aligned} \tag{6.29}$$

Setting $\epsilon = 0$, the following closed-loop fast subsystem is obtained:

$$\frac{dy}{d\tau} = g(x, y + \tilde{g}(x, w), 0, u_f^*, w) \tag{6.30}$$

According to Proposition 6.2, the state $y(t)$ of the closed-loop fast subsystem of Eq. 6.30 starting from $y(0) \in \Omega_{\rho^f}$ stays in Ω_{ρ^f} (i.e., $y(t) \in \Omega_{\rho^f}$ $\forall t \geq 0$) and satisfies the bound of Eq.6.24. Therefore, using similar arguments to the proof of Theorem 6.1 in [20], we have that there exist functions β_x and β_y of class KL, positive real numbers (δ, d) (note that the existence of δ such that $|x(0)| \leq \delta$ and $|y(0)| \leq \delta$ imply that $x(0) \in \Omega_{\rho^s}$ and $y(0) \in \Omega_{\rho^f}$ follows from the smoothness of $V_s(x)$ and $V_f(y)$), and $\epsilon^* > 0$ such that if $\max\{|x(0)|, |y(0)|, ||w||, ||\dot{w}||\} \leq \delta$ and $\epsilon \in (0, \epsilon^*]$, then, the bounds of Eq.6.25 hold for all $t \geq 0$. ■

Remark 6.2 *Referring to the composite fast-slow MPC architecture of Fig. 6.1, we note that it can find an interpretation in the context of distributed MPC architectures [65, 63, 19]. While conventional distributed MPC design where one MPC could manipulate u_f and another MPC could manipulate u_s would normally require the use of communication between the fast MPC and the slow MPC to coordinate their actions ([92, 95]), the fast-slow MPC architecture of Fig. 1 takes advantage of the two-time-scale system property to design a fast-MPC that uses feedback of the deviation variable y in which case u_f is only active in the boundary layer (fast motion of the fast dynamics) and becomes nearly zero in the slow time-scale. As a result, there is no need for communication between the fast MPC and the slow MPC; in this sense, the control structure of Fig. 6.1 can be classified as decentralized. This point demonstrates that accounting for time-scale multiplicity can lead to simplification in the communication strategy of distributed MPCs. Such a two-time-scale DMPC architecture takes advantage of the time-scale separation in the process model and yields*

near optimal performance in a sense to be precisely defined in the next section.

6.5 Near Optimality

In this section, we establish that the finite-time cost of the closed-loop singularly perturbed system of Eq. 6.26 under the fast-slow MPCs, converges to the corresponding cost computed on the basis of the fast and slow subsystems. It should be emphasized that the finite-time time analysis of the closed-loop system optimality relies on the practical closed-loop system stability established in the previous section. We note that after the fast and slow states enter their corresponding final invariant set, we can only guarantee boundedness of the closed-loop system states but not eventual convergence to the origin. Therefore, the integral of the closed-loop cost over the infinite-time interval is infinite. As a consequence, we focus on near-optimality over a finite-time interval.

We assume $w = 0$ and define the finite-time interval $[0, t_I]$, where t_I is the time needed for the state of the closed-loop system of Eq. 6.26 starting from the initial condition $(x(0), z(0))$ that satisfies the conditions of Theorem 6.1 to enter an invariant set containing the origin in which $x(t_I) \in \Omega_{\rho_s^s}$ and $y(t_I) \in \Omega_{\rho_s^f}$. Referring to the system of Eq. 6.26 with $w = 0$, the finite-time cost in the interval time $[0, t_I]$ is defined as follows:

$$\begin{aligned} J &= J_s + J_f \\ &= \int_0^{t_I} \left[x^T(\bar{\tau}) Q_s x(\bar{\tau}) + u_s^{*^T}(\bar{\tau}) R_s u_s^*(\bar{\tau}) \right] d\bar{\tau} \\ &\qquad + \int_0^{t_I} \left[y^T(\bar{\tau}) Q_f y(\bar{\tau}) + u_f^{*^T}(\bar{\tau}) R_f u_f^*(\bar{\tau}) \right] d\bar{\tau} \end{aligned} \tag{6.31}$$

where Q_s, R_s, Q_f, R_f are appropriate matrices defined in section 6.3. We now define

the trajectory of the closed-loop slow subsystem under the slow LMPC:

$$\dot{\hat{x}} = f_s(\hat{x}, u_s^*, 0),\ \hat{x}(0) = x_0 \tag{6.32}$$

for $t \in [0, t_I]$, and the corresponding cost is defined as follows:

$$J_s^* = \int_0^{t_I} \left[\hat{x}^T(\bar{\tau}) Q_s \hat{x}(\bar{\tau}) + u_s^{*^T}(\bar{\tau}) R_s u_s^*(\bar{\tau})\right] d\bar{\tau} \tag{6.33}$$

Similarly, we define the trajectory of the closed-loop fast subsystem under the fast LMPC:

$$\frac{d\hat{y}}{d\tau} = g(x, \hat{y} + \tilde{g}(x, 0), 0, u_f^*, 0),\ \hat{y}(0) = y_0, \tag{6.34}$$

and the corresponding cost is defined as follows:

$$J_f^* = \int_0^{t_b} \left[\hat{y}^T(\frac{\bar{\tau}}{\epsilon}) Q_f \hat{y}(\frac{\bar{\tau}}{\epsilon}) + u_f^{*^T}(\frac{\bar{\tau}}{\epsilon}) R_f u_f^*(\frac{\bar{\tau}}{\epsilon})\right] d\bar{\tau} \tag{6.35}$$

where $t_b \sim \mathcal{O}(\epsilon)$. We now state the main result of this section.

Theorem 6.2 *Consider the closed-loop system of Eq. 6.26 under the slow and fast LMPCs of Eqs. 6.17 and 6.21, respectively, and its corresponding slow and fast subsystems of Eqs. 6.32 and 6.34. Let t_I be the time needed for the state of the closed-loop system of Eq. 6.26 starting from the initial condition $(x(0), z(0))$ satisfying the conditions of Theorem 6.1 to enter an invariant set containing the origin in which $x(t_I) \in \Omega_{\rho_s^s}$ and $y(t_I) \in \Omega_{\rho_s^f}$. Then, $J \to J_s^* + J_f^*$ as $\epsilon \to 0$.*

Proof 6.1 We exploit closeness of solutions results and combine them with optimality results to prove that the two-time-scale LMPC is near-optimal in the sense that the cost function associated with the full closed-loop system approaches the sum of the optimal costs of the reduced subsystems when $\epsilon \to 0$. Using the closed-loop

stability results of Eq. 6.25, we can obtain time t_I which is the time needed for the state of the closed-loop system of Eq. 6.26 starting from $(x(0), z(0))$ satisfying the conditions of Theorem 6.1 to enter an invariant set containing the origin in which $x(t_I) \in \Omega_{\rho_s^s}$ and $y(t_I) \in \Omega_{\rho_s^f}$. Using the bound of Eq. 6.25 and similar arguments to the ones in the proof of Tikhonov's theorem (see Theorem 9.1 in [52]), there exists $\epsilon_0 \in (0, \epsilon^*]$ such that $\forall \epsilon \in (0, \epsilon_0]$:

$$x(t) = \hat{x}(t) + \mathcal{O}(\epsilon), \ \forall t \in [0, t_I] \tag{6.36}$$

$$y(t) = \hat{y}(\frac{t}{\epsilon}) + \mathcal{O}(\epsilon), \ \forall t \in [0, t_I] \tag{6.37}$$

and

$$\begin{aligned} u_s^*(x(t)) &= u_s^*(\hat{x}(t)) + \mathcal{O}(\epsilon), \ \forall t \in [0, t_I] \\ u_f^*(x(t), y(t)) &= u_f^*(\hat{x}(t), \hat{y}(\tfrac{t}{\epsilon})) + \mathcal{O}(\epsilon), \ \forall t \in [0, t_I] \end{aligned} \tag{6.38}$$

From the estimates of Eqs. 6.36, 6.37 and 6.38, it can be concluded that there exists a positive real number $\bar{N}$ such that Eq. 6.31 yields:

$$\begin{aligned} J &= \int_0^{t_I} \left[x^T(\bar{\tau}) Q_s x(\bar{\tau}) + u_s^{*^T}(x(\bar{\tau})) R_s u_s^*(x(\bar{\tau})) \right] d\bar{\tau} \\ &\quad + \int_0^{t_I} \left[y^T(\bar{\tau}) Q_f y(\bar{\tau}) + u_f^{*^T}(x(\bar{\tau}), y(\bar{\tau})) R_f u_f^*(x(\bar{\tau}), y(\bar{\tau})) \right] d\bar{\tau} \\ &= \int_0^{t_I} \left[\hat{x}^T(\bar{\tau}) Q_s \hat{x}(\bar{\tau}) + u_s^{*^T}(\hat{x}(\bar{\tau})) R_s u_s^*(\hat{x}(\bar{\tau})) \right] d\bar{\tau} \\ &\quad + \int_0^{t_b} \left[\hat{y}^T(\frac{\bar{\tau}}{\epsilon}) Q_f \hat{y}(\frac{\bar{\tau}}{\epsilon}) + u_f^{*^T}(\hat{x}(\bar{\tau}), \hat{y}(\frac{\bar{\tau}}{\epsilon})) R_f u_f^*(\hat{x}(\bar{\tau}), \hat{y}(\frac{\bar{\tau}}{\epsilon})) \right] d\bar{\tau} + \bar{N}\epsilon \\ &= J_s^* + J_f^* + \bar{N}\epsilon \end{aligned} \tag{6.39}$$

Thus, as $\epsilon \to 0$, we have that $J \to J_s^* + J_f^*$.

Remark 6.3 *Most of the literature on control of singularly perturbed systems (e.g., [10]) deals with systems in which the manipulated input (or input vector), u, is de-*

composed into two components, u_s and u_f, (i.e., $u = u_s + u_f$) and continuous (not sample-and-hold) implementation of the control action on the process (i.e., singularly perturbed system) is assumed. This formulation for the manipulated inputs and control action implementation, however, is not general enough to deal with model predictive control as the method used for feedback design. Specifically, MPC implementation should be done in a sample-and-hold fashion, and moreover, different sampling times should be utilized by the manipulated inputs used to control the fast and slow dynamics, respectively. In particular, the manipulated inputs used to control the fast dynamics, u_f, should use a small sampling time, while the manipulated inputs used to control the slow dynamics, u_s, could use a larger sampling time. Such a difference in sampling times would not have been possible in the standard singularly perturbed control problem formulation where continuous implementation of manipulated inputs is assumed.

6.6 Chemical Process Example

In this section, we consider a chemical process example to demonstrate the structure and implementation of the proposed fast-slow MPC architecture in a practical setting. The chemical process consists of a network of two continuously stirred tank reactors (CSTRs) and one flash tank separator with recycle. Specifically, fresh feed of species A goes into CSTR 1 through stream F_0. An elementary reaction (r_1) $A \rightarrow B$ takes place inside CSTR 1. The outlet of CSTR 1 is fed into CSTR 2, where a second reaction takes place (r_2) $B + C \rightarrow D$ and produces the desired product D; note that r_2 does not take place in CSTR 1 because catalyst is not added in CSTR 1. Another stream F_4 supplies reactant C into CSTR 2 continuously. All leftover materials from CSTR 2 enter a flash separator where most of the reactants are being recycled back

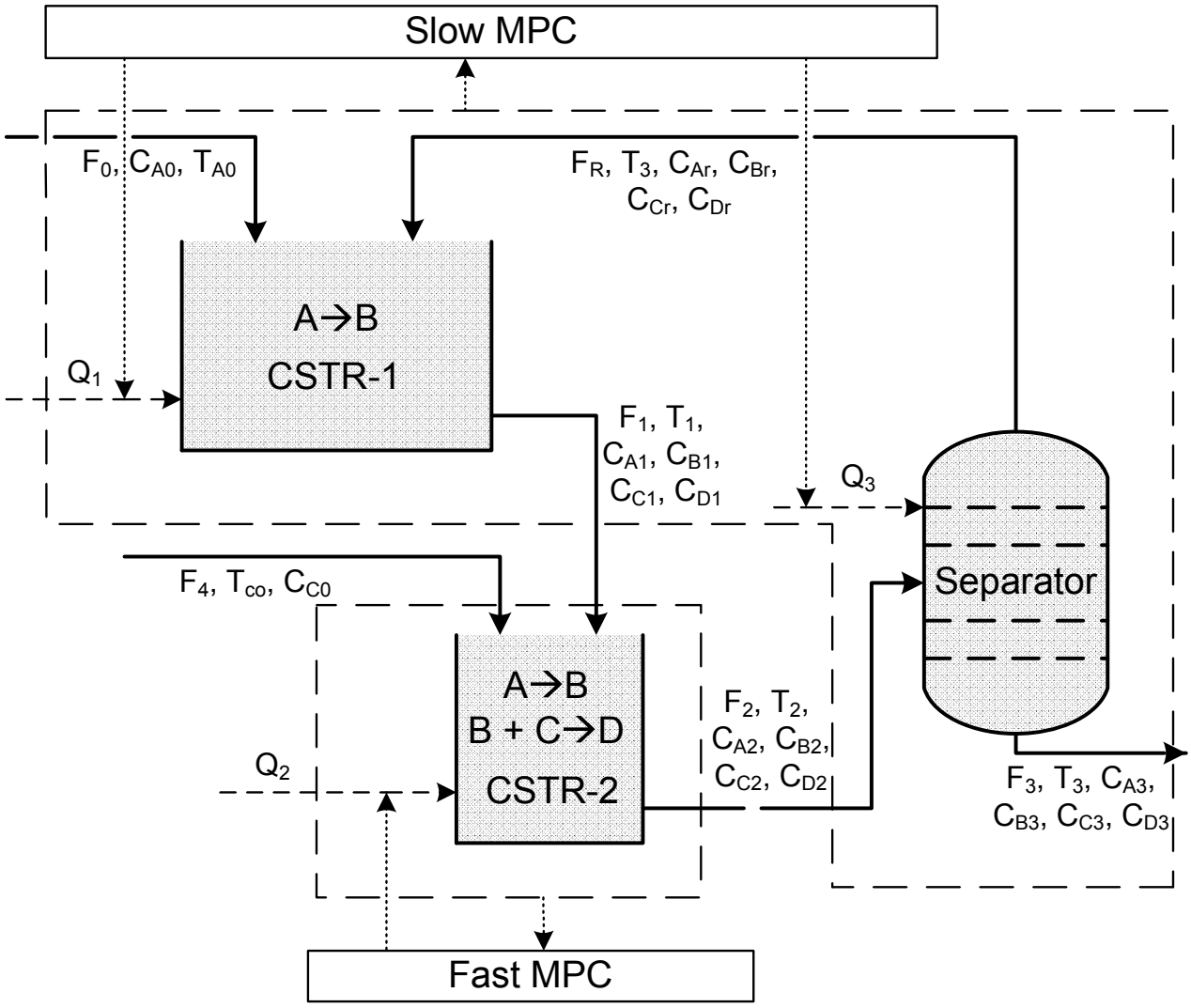


Figure 6.2: Diagram of chemical process example and fast-slow MPC architecture.

to CSTR 1. The dynamic equations describing the behavior of the process, obtained through material and energy balances under standard modeling assumptions, are given below:

$$V_1\frac{dC_{A1}}{dt} = F_0C_{A0} + F_rC_{Ar} - F_1C_{A1} - k_1e^{-E_1/RT_1}C_{A1}V_1 \tag{6.40a}$$

$$V_1\frac{dC_{B1}}{dt} = F_rC_{Br} - F_1C_{B1} + k_1e^{-E_1/RT_1}C_{A1}V_1 \tag{6.40b}$$

$$V_1\frac{dC_{C1}}{dt} = F_rC_{Cr} - F_1C_{C1} \tag{6.40c}$$

$$V_1\frac{dC_{D1}}{dt} = F_rC_{Dr} - F_1C_{D1} \tag{6.40d}$$

$$\rho_m c_{pm}V_1\frac{dT_1}{dt} = F_0\rho_{mA}c_{pmA}T_{A0} + F_r\rho_m c_{pm}T_3 - F_1\rho_m c_{pm}T_1 + (-\Delta H_{r_1})k_1e^{-E_1/RT_1}C_{A1}V_1 + Q_1 \tag{6.40e}$$

$$V_2\frac{dC_{A2}}{dt} = F_1C_{A1} - F_2C_{A2} - k_1e^{-E_1/RT_2}C_{A2}V_2 \tag{6.40f}$$

$$V_2\frac{dC_{B2}}{dt} = F_1C_{B1} - F_2C_{B2} + k_1e^{-E_1/RT_2}C_{A2}V_2 - k_2e^{-E_2/RT}C_{B2}C_{C2}V_2 \tag{6.40g}$$

$$V_2\frac{dC_{C2}}{dt} = F_1C_{C1} + F_4C_{C0} - F_2C_{C2} - k_2e^{-E_2/RT_2}C_{B2}C_{C2}V_2 \tag{6.40h}$$

$$V_2\frac{dC_{D2}}{dt} = F_1C_{D1} - F_2C_{D2} + k_2e^{-E_2/RT_2}C_{B2}C_{C2}V_2 \tag{6.40i}$$

$$\rho_m c_{pm}V_2\frac{dT_2}{dt} = F_1\rho_m c_{pm}T_1 + F_4\rho_{mc}c_{pmc}T_{C0} - F_2\rho_m c_{pm}T_2 + (-\Delta H_{r_1})k_1e^{-E_1/RT_2}C_{A2}V_2 + (-\Delta H_{r_2})k_2e^{-E_2/RT_2}C_{B2}C_{C2}V_2 + Q_2 \tag{6.40j}$$

$$V_3\frac{dC_{A3}}{dt} = F_2C_{A2} - F_3C_{A3} - F_rC_{Ar} \tag{6.40k}$$

$$V_3\frac{dC_{B3}}{dt} = F_2C_{B2} - F_3C_{B3} - F_rC_{Br} \tag{6.40l}$$

$$V_3\frac{dC_{C3}}{dt} = F_2C_{C2} - F_3C_{C3} - F_rC_{Cr} \tag{6.40m}$$

$$V_3\frac{dC_{D3}}{dt} = F_2C_{D2} - F_3C_{D3} - F_rC_{Dr} \tag{6.40n}$$

$$\rho_m c_{pm} V_3\frac{dT_3}{dt} = F_2\rho_m c_{pm}(T_2 - T_3) - \sum_{i}^{A,B,C,D} F_rC_{ir}H_i^{vap} + Q_3 \tag{6.40o}$$

where the definitions of the the process variables and their assigned values are shown in Table 6.1 and Table 6.2, respectively.

The models of the CSTRs and of the separator are developed under the assumptions that the liquid hold-up level of all tanks is fixed and the relative volatility of each species is constant. The following algebraic equations govern the molar composition of different species at the recycle stream:

$$\begin{aligned}
x_{Ar} &= \frac{\alpha_A C_{A3}}{\alpha_A C_{A3} + \alpha_B C_{B3} + \alpha_C C_{C3} + \alpha_D C_{D3}} \\
x_{Br} &= \frac{\alpha_B C_{B3}}{\alpha_A C_{A3} + \alpha_B C_{B3} + \alpha_C C_{C3} + \alpha_D C_{D3}} \\
x_{Cr} &= \frac{\alpha_C C_{C3}}{\alpha_A C_{A3} + \alpha_B C_{B3} + \alpha_C C_{C3} + \alpha_D C_{D3}} \\
x_{Dr} &= \frac{\alpha_D C_{D3}}{\alpha_A C_{A3} + \alpha_B C_{B3} + \alpha_C C_{C3} + \alpha_D C_{D3}}
\end{aligned}$$

where x_{ir}, $i = A, B, C, D$ represents the molar composition of species A, B, C and D, respectively. Each of the tanks has an external heat input that is used as manipulated input, labeled as Q_1, Q_2 and Q_3. In this example, the liquid holdup V_2 in the catalytic reactor CSTR 2 is significantly smaller than the liquid holdup in CSTR 1, V_1, and the flash separator V_3.

Taking this into account, the process model of Eq. 6.40 can be converted into

Table 6.1: Process variables

C_{i1}, i = A,B,C D	Concentration of different species at CSTR-1
C_{i2}, i = A,B,C D	Concentration of different species at CSTR-2
C_{i3}, i = A,B,C D	Concentration of different species at CSTR-3
C_{ir}, i = A,B,C D	Concentration of different species at F_r
T_j, i = 1,2,3,r	Temperatures of CSTR-1,2,3
T_{A0}, T_{C0}	Temperatures of stream F_0 and F_4
V_1, V_2 and $V3$	Vessel volume of CSTR-1,2 and separator
ρ_m, ρ_{mA} and ρ_{mC}	Density of the mixture, species A and C
c_m, c_{mA} and c_{mC}	Heat capacity of the mixture, species A and C
ΔH_{r_1} and ΔH_{r_2}	Heat of reaction r_1 and r_2
k_1 and k_2	Reaction coefficients of r_1 and r_2
E_1 and E_2	Activation energy of r_1 and r_2
H_i^{vap}, i =A,B,C,D	Enthalpy of vaporization of different species

Table 6.2: Process values

V_1	4.0 m^3	V_2	0.2 m^3
V_3	3.0 m^3	T_{A0}	298.15 K
T_{C0}	298.15 K	C_{A0}	2 $kmol\ m^{-3}$
C_{C0}	2 $kmol\ m^{-3}$	F_0	0.04 $m^3\ s^{-1}$
F_4	0.05 $m^3\ s^{-1}$	F_r	1.8 $m^3\ s^{-1}$
ρ_m	900.0 $kg\ m^{-3}$	ρ_{mA}	950.0 $kg\ m^{-3}$
ρ_{mC}	870.0 $kg\ m^{-3}$	c_{pm}	0.231 $kcal\ kg^{-1}\ K^{-1}$
c_{pmA}	0.214 $kcal\ kg^{-1}\ K^{-1}$	c_{pmC}	0.251 $kcal\ kg^{-1}\ K^{-1}$
ΔH_{r_1}	$5.4 \times 10^1\ kcal\ mol^{-1}$	ΔH_{r_2}	$9.98 \times 10^1\ kcal\ mol^{-1}$
k_1	$3.35 \times 10^3\ s^{-1}$	k_2	$5.25 \times 10^4\ m^3\ kmol^{-1}\ s^{-1}$
E_1	$1.04 \times 10^4\ kcal\ kmol$	E_2	$4.0 \times 10^3\ kcal\ kmol$
R	1.987 $kcal\ kmol^{-1}\ K^{-1}$	H_A^{vap}	100 $kcal\ kmol^{-1}$
H_B^{vap}	110 $kcal\ kmol^{-1}$	H_C^{vap}	120 $kcal\ kmol^{-1}$
H_D^{vap}	120 $kcal\ kmol^{-1}$		

the standard singularly perturbed form by dividing Eq.6.40f-6.40j by V_1 and defining $\varepsilon = \frac{V_2}{V_1}$ to derive the following state-space model:

$$\begin{aligned} \dot{x} &= f(x, z, \varepsilon, Q_1, Q_3, w_s), \ x(0) = x_0 \\ \varepsilon \dot{z} &= g(x, z, \varepsilon, Q_2, w_f), \ z(0) = z_0 \end{aligned} \tag{6.41}$$

where the fast state $z = [C_{A2} \ C_{B2} \ C_{C2} \ C_{D2} \ T_2]$ and the slow state x consists of the rest of the state variables. With respect to the manipulated input decomposition, Q_1 and Q_3 enter the slow process states, x, and will be regulated by the slow MPC, and Q_2 enters the fast process states, z, and will be regulated by the fast MPC. Finally, we define the following deviation variables $\bar{z} = z - z_{set}$ and $\bar{x} = x - x_{set}$, where z_{set} and x_{set} are the desired (final) steady-state values and are defined in Table. 6.3.

6.6.1 Controller synthesis

Table 6.3: Initial steady-state and final steady-state manipulated input values

Initial steady state (unit kcal)		
$Q_1^{int} = 3.2e3$	$Q_2^{int} = -3.7e3$	$Q_3^{int} = 3.5e3$
Final steady-state (unit kcal)		
$Q_1^{set} = 5.2e3$	$Q_2^{set} = -4.7e3$	$Q_3^{set} = 2.8e3$

In this section, we synthesize three control schemes: a) a centralized LMPC architecture, b) an MPC architecture that uses a "fast" explicit controller and a slow LMPC [12], and c) the composite fast-slow LMPC architecture introduced in this paper, shown in Figure 6.2. All controller designs have the same objective to drive the system from an initial steady-state to a final steady-state; both are stable steady-

states and are defined in Table 6.3. In the context of the centralized LMPC, the manipulated input vector is defined as $u = [u_1\ u_2\ u_3]^T = [Q_1 - Q_1^{set}\ Q_2 - Q_2^{set}\ Q_3 - Q_3^{set}]^T$ for all LMPC synthesis and their constraints are chosen to be:

$$3.2e3\ kcal \leq u_1 \leq 6.0e3\ kcal$$
$$-5.5e3\ kcal \leq u_2 \leq -2.8e3\ kcal$$
$$2.0e3\ kcal \leq u_3 \leq 4.0e3\ kcal$$

We consider the following objective function in the centralized LMPC design:

$$J_c = \int_0^{t_I} [\bar{x}^T(\bar{\tau}) Q_{c1} \bar{x}(\bar{\tau}) + \bar{z}^T(\bar{\tau}) Q_{c2} \bar{z}(\bar{\tau}) + u^T(\bar{\tau}) R_c u(\bar{\tau})] d\bar{\tau} \tag{6.42a}$$

where Q_{c1} and Q_{c2} are weighting matrices and their diagonal values are defined as the reciprocal of the average of the initial and final steady-state values of the states they are associated with, and $R_c = diag([1.0e-8\ 1.0e-8\ 1.0e-10])$ is also a weighting matrix. With respect to the Lyapunov constraint, three different proportional controllers are implemented as $h(\bar{x})$:

$$u_1 = k_1(T_1^{set} - T_1) \tag{6.43a}$$

$$u_2 = k_2(T_2^{set} - T_2) \tag{6.43b}$$

$$u_3 = k_3(T_3^{set} - T_3) \tag{6.43c}$$

where T_1^{set}, T_2^{set} and T_3^{set} are the final steady-state temperatures of each vessel and k_1, k_2 and k_3 are constant coefficients and are chosen to be -60, 50 and 30, respectively. All diagonal elements of the weighting matrices of the Lyapunov function $V = \bar{x}^T P_1 \bar{x} + \bar{z}^T P_2 \bar{z}$ are chosen to be unity.

To synthesize the slow LMPC and the composite fast-slow LMPC schemes, we first split the set of manipulated inputs u into $u_s = [u_1\ u_3]$ and $u_f = [u_2]$. Following from Eq. 6.17, the slow time-scale objective function J_s in this example is chosen to be:

$$J_s = \int_0^{t_I} \left[\bar{x}^T(\bar{\tau}) Q_{c1} \bar{x}(\bar{\tau}) + u_s^T(\bar{\tau}) R_s u_s(\bar{\tau})\right] d\bar{\tau} \tag{6.44}$$

where R_s has the same values as in R_c for the same input variables. In order to solve for Eq. 6.17c, the nonlinear algebraic equation $g(\bar{x}, w)$ has to be solved at each slow sampling time when the slow state measurements are available. The Lyapunov based controller $h_s(\bar{x})$ in the slow LMPC design implements the same proportional control law as in Eq.6.43a and Eq.6.43c. To complete the synthesis of the slow LMPC scheme, we assign the slow LMPC to regulate the heat inputs Q_1 and Q_3 and assign the proportional controller Eq.6.43b to regulate the heat input Q_2.

Finally, we use the following objective function J_f for the fast LMPC:

$$J_f = \int_0^{t_I} \left[y^T(\bar{\tau}) Q_{c2} y(\bar{\tau}) + u_f^T(\bar{\tau}) R_f u_f(\bar{\tau})\right] d\tau \tag{6.45}$$

where $R_f = 1.0e-8$ and y is defined as $y = \bar{z} - g(\bar{x}, w)$. The formulation of fast LMPC requires that the nonlinear function $g(\bar{x}, w)$ has to be solved at the end of each fast sampling time where the slow state measurements are available. The controller $p(\bar{x})y$ appeared on the Lyapunov constraint at Eq. 6.21e is designed to be a control law of the form $p(\bar{x})y = k_4(g(\bar{x}, w) - T_2)$ where $k_4 = 25$. Together with the slow LMPC design, this completes the synthesis of the composite fast-slow LMPC.

6.7 Closed-loop results

The simulations are performed in C++ programming environment by a Core2 Quad Q6600 computer. The simulation time for each run is 510 seconds. We study the closed-loop system performance as well as the response speed by each of the MPC schemes, and by comparison, we evaluate the characteristics of the composite fast-slow LMPC design in terms of stability, optimality and control action evaluation time.

For all simulation studies, we use the same prediction horizon $N = 1$. The fast and slow state measurements are assumed to be available at every three seconds ($\Delta = 3\ s$) to all LMPC controllers, and the temperature measurement (T_2) is assumed to be continuously available to the proportional controller in the composite LMPC design. The sampling time of the centralized LMPC, fast LMPC, and slow LMPC are $\Delta_c = 15\ s$, $\Delta_s = 15\ s$ and $\Delta_f = 3\ s$, respectively. For each LMPC scheme, only the first piece from the computed optimal input trajectory of the optimization problems is implemented in each sampling time following a receding horizon scheme. The manipulated input regulated by the proportional controller in the slow LMPC scheme is also implemented in a sample-and-hold fashion and its sampling time is denoted as Δ_f^P.

The numerical method that is used to integrate the process model is explicit Euler with a fixed time step equal to 0.001 seconds. However, in the cases of slow LMPC scheme and composite fast-slow LMPC scheme, the explicit separation of the fast and slow dynamics allows a larger time step to be used by the slow LMPC, and it is 0.1 seconds. The optimization problems of each LMPC scheme are solved using the open source interior point optimizer Ipopt. The Hessian is approximated by Quasi-Newton's method. Regarding the termination criteria and the maximum number of

iterations, the values used by all simulations are 10^{-3} and 150, respectively.

The closed-loop performance of all LMPC schemes is illustrated in Fig. 6.3 and in Fig. 6.4. The trajectories of the slow LMPC scheme with the "fast" proportional controller in Fig. 6.3 are evaluated based on Eq. 6.42a for the purpose of comparison. The performance costs in both figures clearly indicate that both the centralized LMPC and the fast-slow LMPC schemes stabilize the closed-loop process asymptotically to the final steady state. We also note that both the fast LMPC and the slow LMPC in the composite fast-slow MPC design are able to stabilize the fast and slow dynamics of the singularly perturbed system asymptotically (Fig. 6.4). On the other hand, under the setting of $\Delta_f^P = 1\ s$, the slow LMPC scheme using a "fast" explicit controller is not able to drive the closed-loop system to the final steady state. It causes oscillations of the closed-loop system around the final steady state, which contributes to the increase of the performance cost with increasing simulation time (Fig. 6.3). This problem can be resolved if the sampling time for the "fast" proportional controller is reduced to 0.001 seconds; specifically, Fig. 6.3 shows that in this case the oscillations diminish and the closed-loop system converges to the final steady state gradually with a performance cost that is higher than the one of the centralized LMPC. Finally, it is important to note that we can not directly compare the optimal performance of the centralized LMPC scheme and of the composite fast-slow LMPC scheme since they utilize different cost functions.

Fig. 6.5 and Fig. 6.6 show the simulation results comparing the computational speed between the centralized LMPC and the composite fast-slow LMPC. Since the explicit separation between the fast and slow dynamics reduces the dimension of the process dynamic model and the corresponding step of numerical integration as well as the number of manipulated inputs for the slow LMPC and the fast LMPC, the

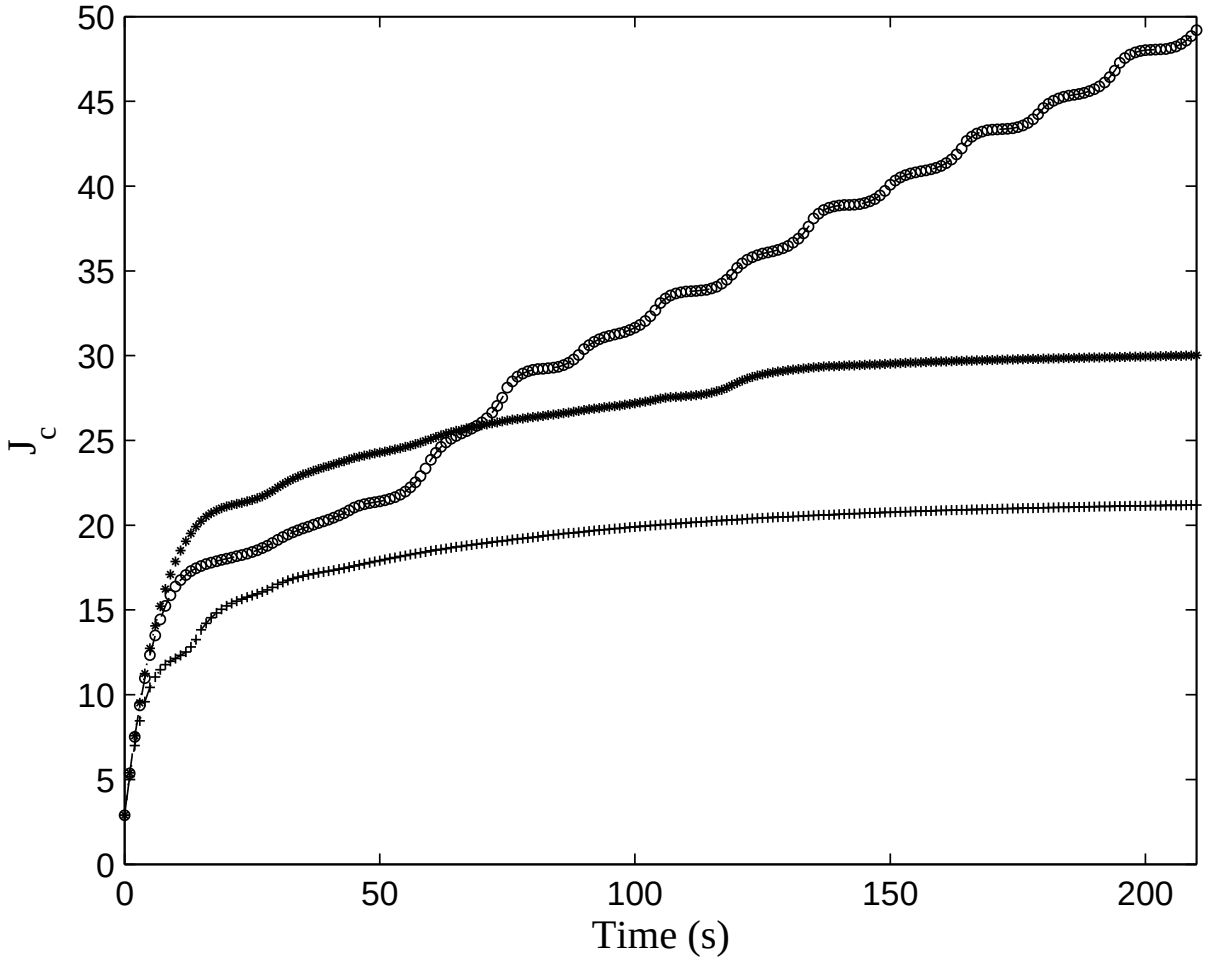


Figure 6.3: Performance cost of each controller design based on Eq. 6.42a: the centralized LMPC design (dotted-solid line with plus), the slow LMPC design (dashed line with circles) with the "fast" proportional controller using $\Delta_f^P = 1s$, and the slow LMPC design (solid line with asterisks) with the "fast" proportional controller using $\Delta_f^P = 0.001s$.

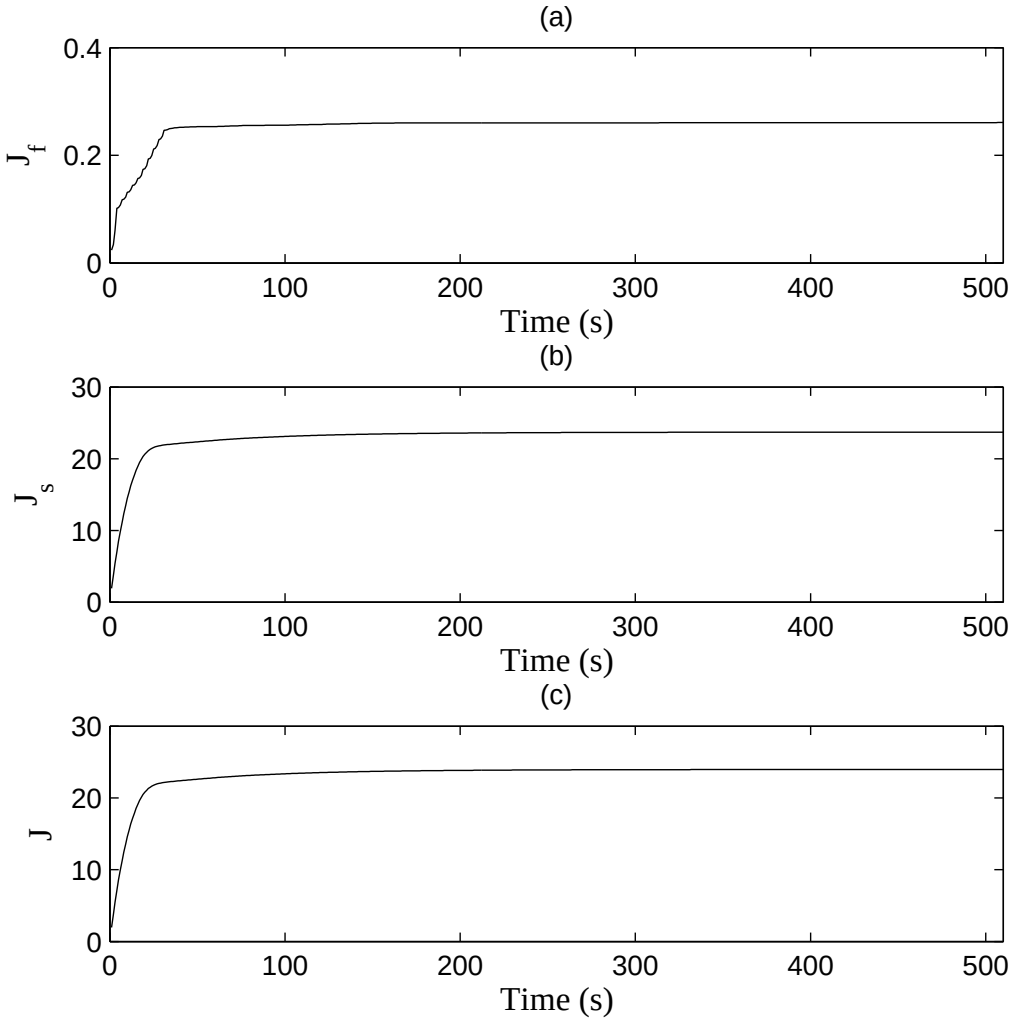


Figure 6.4: Fast-slow LMPC performance: (a) performance cost, J_f, of the fast LMPC, (b) performance cost, J_s, of the slow LMPC and (c) overall cost, J.

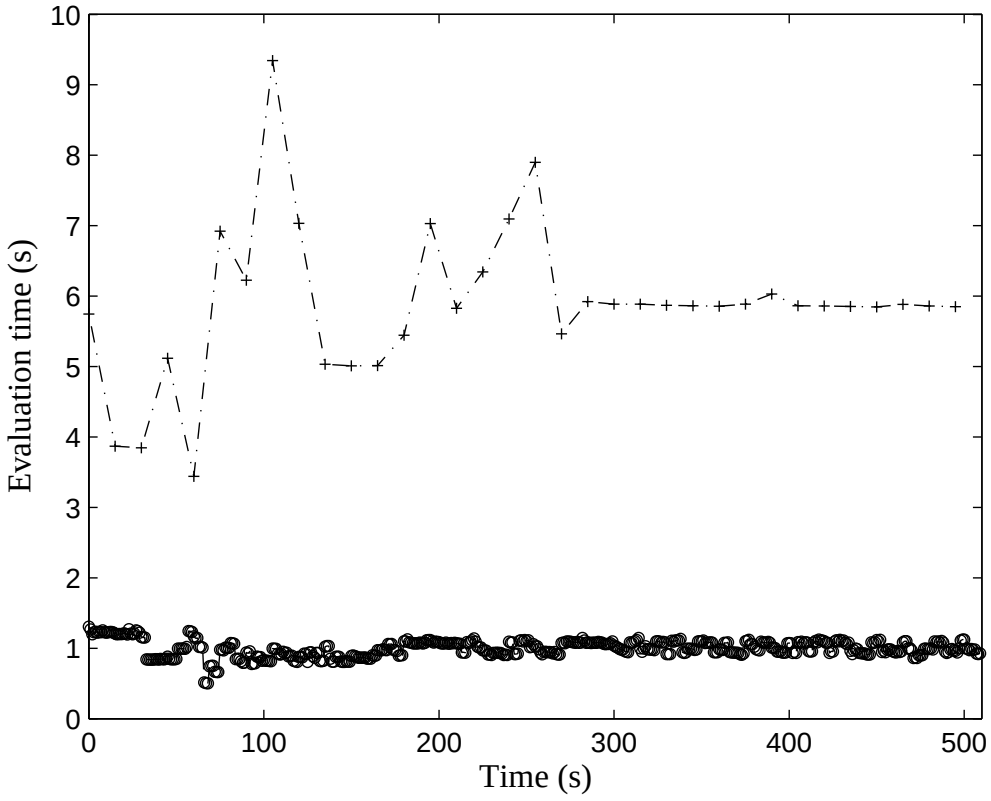


Figure 6.5: The total evaluation time needed for each evaluation of the control action by the centralized LMPC design (dashed-dotted line with plus) and by the fast LMPC of the fast-slow LMPC design (dashed line with circles).

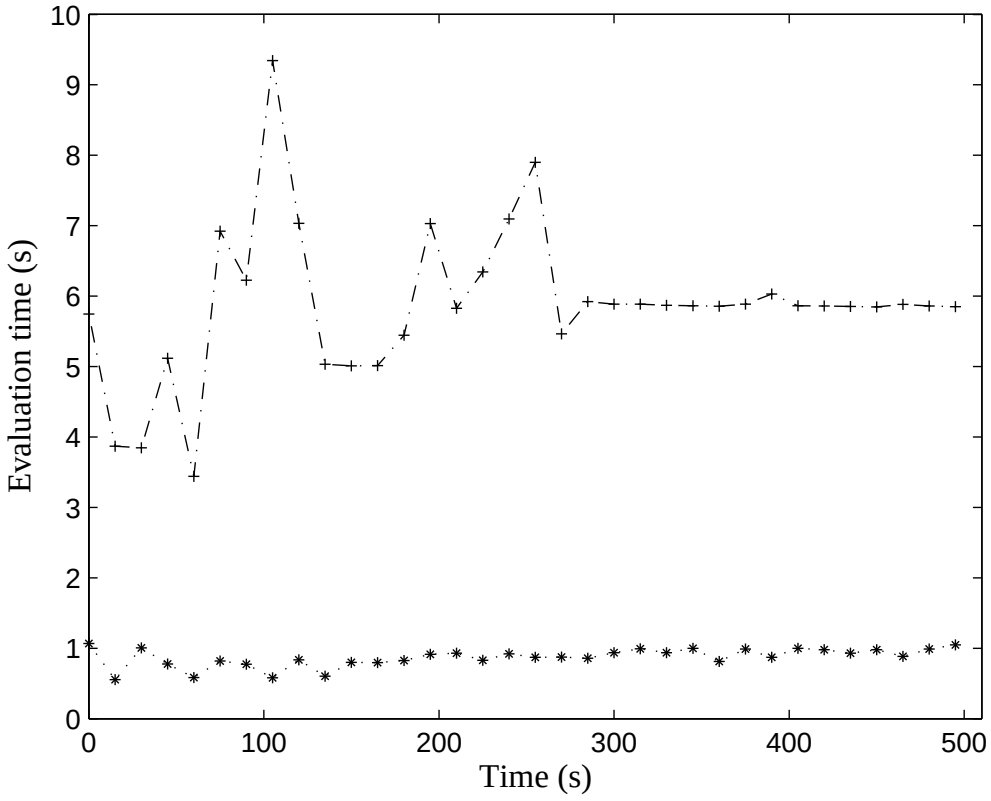


Figure 6.6: The total evaluation time needed for each evaluation of the control action by the centralized LMPC design (dashed-dotted line with plus) and by the slow LMPC of the fast-slow LMPC design (dotted line with asterisks).

composite fast-slow LMPC computes the control action significantly faster compared to the centralized LMPC. Specifically, the results in Fig. 6.5 and Fig. 6.6 indicate that the fast LMPC needs on average one second for the control action evaluation at each fast sampling time, and compared to the centralized LMPC, it is at least five times faster on average. Similarly, the slow LMPC is at least five times faster on average in evaluating its control actions at each slow sampling time compared to the time needed for the centralized LMPC.

6.8 Conclusions

This chapter focused on the theoretical development of a composite fast-slow LMPC architecture for nonlinear singularly perturbed systems in standard form and its application to a chemical process which consists of two continuous stirred tank reactors (CSTRs) and a flash separator with recycle. The proposed fast-slow MPC design is decentralized in nature and enforces stability and near-optimality in the closed-loop singularly perturbed system provided the singular perturbation parameter is sufficiently small. Extensive simulations were carried out to compare the proposed architecture with existing centralized LMPC techniques from computational time and closed-loop performance points of view.

Chapter 7

Conclusions

This dissertation presented methods for designing distributed model predictive control (MPC) systems for nonlinear and two-time-scale process networks and demonstrated their application using various chemical process network examples.

Specifically, Chapter 2 presented two different architectures of distributed MPC for nonlinear process systems: sequential distributed LMPC and iterative distributed LMPC. In both architectures, the MPC controllers were designed via LMPC techniques. In the sequential distributed LMPC architecture, the distributed LMPC controllers adopt a one-directional communication strategy and are evaluated in sequence and once at each sampling time; in the iterative distributed LMPC architecture, the distributed LMPC controllers utilize a bi-directional communication strategy, are evaluated in parallel and iterate to improve closed-loop performance. Each LMPC controller in both architectures incorporates a suitable stability constraint which ensures that the state of the closed-loop system under the proposed distributed MPC architectures is ultimately bounded in an invariant set. Extensive simulations using a catalytic alkylation of benzene process example were carried out to compare the proposed distributed MPC architectures with existing centralized LMPC algorithms

from computational time and closed-loop performance points of view.

In Chapter 3, we designed sequential and iterative DMPC schemes for large scale nonlinear systems subject to asynchronous and delayed state feedback. First, we focused on nonlinear systems subject to asynchronous measurements without delays. In this case, we first extended our previous sequential DMPC design [66] to include asynchronous measurements and then re-designed the iterative DMPC scheme proposed in [63] to take explicitly into account asynchronous feedback. Following that, we focused on the design of an iterative DMPC scheme for nonlinear systems subject to delayed measurements. This design took advantage of the bi-directional communication network already used in the iterative DMPC framework. Mathematical analyses were carried out to derive sufficient conditions under which the proposed distributed control designs guarantee that the states of the closed-loop system are ultimately bounded in regions that contain the origin. Through a catalytic alkylation of benzene process example, we successfully compared the proposed DMPC designs with two centralized LMPC designs from stability, evaluation time and performance points of view.

Chapter 4 presented an application of centralized LEMPC and sequential distributed LEMPC architectures to a catalytic alkylation of benzene process network which consists of four continuous stirred tank reactors and a flash separator. In the sequential distributed LEMPC design, three separate Lyapunov-based model predictive controllers were designed to control the process in a sequential coordinated fashion. The closed-loop stability properties of the sequential distributed LEMPC design were rigorously analyzed and sufficient conditions for closed-loop stability were established. Simulations were carried out to compare the proposed economic MPC architectures with a centralized LMPC which uses a quadratic cost function that includes penalty

on the deviation of the states and inputs from their economically optimal steady-state values, from computational time and closed-loop performance points of view.

Chapter 5 focused on model predictive control of a class of nonlinear singularly perturbed systems. The motivation for this work is provided by broad classes of large-scale process networks that involve coupled variables that evolve in disparate (fast and slow) time scales. For such process networks, direct application of model predictive control to compute the control actions for all manipulated inputs leads to very high-order optimization problems that may not be solvable in real-time. Instead, we proposed a control system using multirate sampling (i.e., fast sampling of easy-to-measure fast-evolving variables and slow sampling of slow-evolving variables) and consisting of an explicit feedback controller that stabilizes the fast dynamics and a model predictive controller that stabilizes the slow dynamics and enforces desired performance objectives in the slow subsystem was proposed. In this way, the model predictive controller solves an optimization problem with a substantially smaller number of decision variables, and thus, it requires less computational time. Sufficient conditions under which the closed-loop system stability, accounting for multirate sampling and sample-and-hold implementation of the predictive controller, is guaranteed were provided. The applicability and effectiveness of the proposed control system was illustrated via a large-scale nonlinear reactor-separator process network which exhibits two-time-scale behavior and the computational effectiveness of distributed predictive control implementation was demonstrated.

In Chapter 6, we emphasized on the theoretical development of a composite fast-slow LMPC architecture for nonlinear singularly perturbed systems in standard form and its application to a chemical process which consists of two continuous stirred tank reactors (CSTRs) and a flash separator with recycle. The proposed fast-slow MPC

design is decentralized in nature and enforces stability and near-optimality in the closed-loop singularly perturbed system provided the singular perturbation parameter is sufficiently small. Extensive simulations were carried out to compare the proposed architecture with existing centralized LMPC techniques from computational time and closed-loop performance points of view.

Future research in distributed predictive control as well as related areas includes the development of general methods for the handling of broad types of communication disruptions between distributed controllers, the design of distributed state estimation systems which provide fast and guaranteed convergence and the development of distributed plant monitoring and fault-tolerant control systems. The reader may refer to [17, 90, 92] for more discussions on the related open problems.

Bibliography

[1] D. Angeli, R. Amrit, and J. B. Rawlings. Receding horizon cost optimization for overly constrained nonlinear plants. In *Proceedings of the 48th IEEE Conference on Decision and Control and the 28th Chinese Control Conference*, pages 7972–7977, Shanghai, China, 2009.

[2] T. Backx, O. Bosgra, and W. Marquardt. Integration of model predictive control and optimization of processes : Enabling technology for market driven process operation. In *Proceedings of the IFAC Symposium on Advanced Control of Chemical Processes*, pages 249–260, Pisa, Italy, 2000.

[3] M. Baldea and P. Daoutidis. Control of integrated chemical process systems using underlying DAE models. *Control and Optimization with Differential-Algebraic Constraints, L. T. Biegler, S. Campbell, V. Mehrmann, eds.*, SIAM, to appear, 2011.

[4] P. I. Barton and C. C. Pantelides. Modeling of combined discrete/continuous processes. *AIChE Journal*, 40:966–979, 1994.

[5] A. Bemporad and M. Morari. Control of systems integrating logic, dynamics and constraints. *Automatica*, 35:407–427, 1999.

[6] W. B. Bequette. Nonlinear control of chemical processes: A review. *Industrial & Engineering Chemistry Research*, 30:1391–1413, 1991.

[7] L. T. Biegler and I. E. Grossmann. Part II. Future perspective on optimization. *Computers & Chemical Engineering*, 28:1193–1218, 2004.

[8] L. T. Biegler and I. E. Grossmann. Retrospective on optimization. *Computers & Chemical Engineering*, 28:1169–1192, 2004.

[9] M. A. Brdys, M. Grochowski, T. Gminski, K. Konarczak, and M. Drewa. Hierarchical predictive control of integrated wastewater treatment systems. *Control Engineering Practice*, 16:751–767, 2008.

[10] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar. Distributed model predictive control. *IEEE Control Systems Magazine*, 22:44–52, 2002.

[11] X. Chen, M. Heidarinejad, J. Liu, and P. D. Christofides. Composite fast-slow MPC design for nonlinear singularly perturbed systems. *AIChE Joural*, 58:1802–1811, 2012.

[12] X. Chen, M. Heidarinejad, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Model predictive control of nonlinear singularly perturbed systems: Application to a large-scale process network. *Journal of Process Control*, 21:1296–1305, 2011.

[13] X. Chen, M. Heidarinejad, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Distributed economic MPC: Application to a nonlinear chemical process network. *Journal of Process Control*, 22:689–699, 2012.

[14] X. Chen, J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Sequential and iterative distributed model predictive control of nonlinear process systems subject to asynchronous measurements. In *Proceedings of the 9th International Symposium on Dynamics and Control of Process Systems*, pages 611–616, Leuven, Belgium, 2010.

[15] P. D. Christofides. *Nonlinear and Robust Control of PDE Systems: Methods and Applications to Transport-Reaction Processes*. Birkhäuser, Boston, 2001.

[16] P. D. Christofides and P. Daoutidis. Feedback control of two-time-scale nonlinear systems. *International Journal of Control*, 63:965–994, 1996.

[17] P. D. Christofides, J. F. Davis, N. H. El-Farra, D. Clark, K. R. D. Harris, and J. N. Gipson. Smart plant operations: Vision, progress and challenges. *AIChE Journal*, 53:2734–2741, 2007.

[18] P. D. Christofides and N. H. El-Farra. *Control of nonlinear and hybrid process systems: Designs for uncertainty, constraints and time-delays*. Springer-Verlag, Berlin, Germany, 2005.

[19] P. D. Christofides, J. Liu, and D. Muñoz de la Peña. *Networked and Distributed Predictive Control - Methods and Nonlinear Process Network Applications*. Advances in Induatrial Control Series. Springer-Verlag, Berlin, Germany, 2011.

[20] P. D. Christofides and A. R. Teel. Singular perturbations and input-to-state stability. *IEEE Transactions on Automatic Control*, 41:1645–1650, 1996.

[21] P. D. Christofides, A. R. Teel, and P. Daoutidis. Robust semi-global output tracking for nonlinear singularly perturbed systems. *International Journal of Control*, 65:639–666, 1996.

[22] F. Clarke, Y. Ledyaev, and E. Sontag. Asymtotic controllability implies feedback stabilization. *IEEE Transactions on Automatic Control*, 42:1394–1407, 1997.

[23] M. Corless. Control of uncertain nonlinear systems. *Journal of Dynamic Systems, Measurement, and Control*, 115:362–372, 1993.

[24] J. F. Davis. *Report from NSF Workshop on Cyberinfrastructure in Chemical and Biological Systems: Impact and Directions,* (see http://www.oit.ucla.edu/nsfci/NSFCIFullReport.pdf for the pdf file of this report). Technical report, 2007.

[25] R. A. Decarlo, M. S. Branicky, S. Petterson, and B. Lennartson. Perspectives and results on the stability and stabilizability of hybrid systems. *Proceedings of the IEEE*, 88:1069–1082, 2000.

[26] M. Diehl, R. Amrit, and J. B. Rawlings. A Lyapunov function for economic optimizing model predictive control. *IEEE Transactions on Automatic Control*, 56:703–707, March 2011.

[27] W. B. Dunbar. Distributed receding horizon control of dynamically coupled nonlinear systems. *IEEE Transactions on Automatic Control*, 52:1249–1263, 2007.

[28] N. H. El-Farra and P. D. Christofides. Integrating robustness, optimality and constraints in control of nonlinear processes. *Chemical Engineering Science*, 56:1841–1868, 2001.

[29] N. H. El-Farra and P. D. Christofides. Bounded robust control of constrained multivariable nonlinear processes. *Chemical Engineering Science*, 58:3025–3047, 2003.

[30] N. H. El-Farra and P. D. Christofides. Coordinated feedback and switching for control of hybrid nonlinear processes. *AIChE Journal*, 49:2079–2098, 2003.

[31] N. H. El-Farra, P. Mhaskar, and P. D. Christofides. Hybrid predictive control of nonlinear systems: Method and applications to chemical processes. *International Journal of Robust and Nonlinear Control*, 14:199–225, 2004.

[32] S. Engell, S. Kowalewski, C. Schulz, and O. Stursberg. Continuous-discrete interactions in chemical processing plants. *Proceedings of the IEEE*, 88:1050–1068, 2000.

[33] C. A. Floudas. *Nonlinear and Mixed-Integer Optimization: Fundamentals and Applications*. Oxford University Press, New-York, 1995.

[34] E. Franco, L. Magni, T. Parisini, M. M. Polycarpou, and D. M. Raimondo. Cooperative constrained control of distributed agents with nonlinear dynamics and delayed information exchange: A stabilizing receding-horizon approach. *IEEE Transactions on Automatic Control*, 53:324–338, 2008.

[35] H. Ganji, J. S. Ahari, A. Farshi, and M. Kakavand. Modelling and simulation of benzene alkylation process reactors for production of ethylbenzene. *Petroleum and Coal*, 46:55–63, 2004.

[36] C. E. García, D. M. Prett, and M. Morari. Model predictive control: Theory and practice–A survey. *Automatica*, 25:335–348, 1989.

[37] V. Garcia-Onorio and B. E. Ydstie. Distributed, asynchronous and hybrid simulation of process networks using recording controllers. *International Journal of Robust and Nonlinear Control*, 14:227–248, 2004.

[38] I. E. Grossmann, S. A. van den Heever, and I. Harjukoski. Discrete optimization methods and their role in the integration of planning and scheduling. In *Proceedings of 6th International Conference on Chemical Process Control*, pages 124–152, Tucson, AZ, 2001.

[39] I. Harjunkoski, V. Jain, and I. E. Grossmann. Hybrid mixed-integer/constrained logic programming strategies for solving scheduling and combinatorial optimization problems. *Comp. & Chem. Eng.*, 24:337–343, 2000.

[40] M. Heidarinejad, J. Liu, and P. D. Christofides. Economic model predictive control of nonlinear process systems using Lyapunov techniques. *AIChE Journal*, 58:855–870, 2012.

[41] M. A. Henson and D. E. Seborg. *Nonlinear Process Control*. Prentice-Hall, Englewood Cliffs, NJ, 1997.

[42] J. P. Hespanha and A. S. Morse. Stability of switched systems with average dwell time. In *Proceedings of 38th IEEE Conference on Decision and Control*, pages 2655–2660, Phoenix, AZ, 1999.

[43] B. Hu, X. Xu, P. J. Antsaklis, and A. N. Michel. Robust stabilizing control law for a class of second-order switched systems. *Systems and Control Letters*, 38:197–207, 1999.

[44] R. Huang, E. Harinath, and L. T. Biegler. Lyapunov stability of economically-oriented NMPC for cyclic processes. *Journal of Process Control*, 21:501–509, 2011.

[45] S. C. Jeong and P. Park. Constrained MPC algorithm for uncertain time-varying systems with state-delay. *IEEE Transactions on Automatic Control*, 50:257–263, 2005.

[46] D. Jia and B. Krogh. Min-max feedback model predictive control for distributed control with communication. In *Proceedings of the American Control Conference*, pages 4507–4512, Anchorage, Alaska, 2002.

[47] J. V. Kadam and W. Marquardt. Integration of economical optimization and control for intentionally transient process operation. In R. Findeisen, F. Allgöwer, and L. T. Biegler, editors, *Assessment and Future Directions of Nonlinear Model Predictive Control, LNCIS*, volume 358, pages 419–434, 2007.

[48] N. Kapoor and P. Daoutidis. Stabilization of systems with input constraints. *International Journal of Control*, 34:653–675, 1998.

[49] N. Kazantzis and C. Kravaris. Energy-predictive control: a new synthesis approach for nonlinear process control. *Chemical Engineering Science*, 54:1697–1709, 1999.

[50] T. Keviczky, F. Borrelli, and G. J. Balas. Decentralized receding horizon control for large scale dynamically decoupled systems. *Automatica*, 42:2105–2115, 2006.

[51] H. Khalil. Robust servomechanism output feedback controller for feedback linearizable systems. *Automatica*, 30:1587–1599, 1994.

[52] H. K. Khalil. *Nonlinear systems*. Prentice Hall, second edition, 1996.

[53] P. Kokotovic and M. Arcak. Constructive nonlinear control: a historical perspective. *Automatica*, 37:637–662, 2001.

[54] P. Kokotovic, H. K. Khalil, and J. O'Reilly. *Singular Perturbation Methods in Control: Analysis and Design*. Academic Press, London, 1986.

[55] X. D. Koutsoukos, P. J. Antsaklis, J. A. Stiver, and M. D. Lemmon. Supervisory control of hybrid systems. *Proceedings of the IEEE*, 88:1026 –1049, 2000.

[56] C. Kravaris and Y. Arkun. Geometric nonlinear control - an overview. In *Proceedings of 4th International Conference on Chemical Process Control*, pages 477–515, Y. Arkun and W. H. Ray Eds., Padre Island, TX, 1991.

[57] A. Kumar and P. Daoutidis. Nonlinear dynamics and control of process systems with recycle. *Journal of Process Control*, 12:475–484, 2002.

[58] W. J. Lee. *Ethylbenzene dehydrogenation into styrene: kinetic modeling and reactor simulation*. PhD thesis, Texas A&M University, College Station, TX, USA, 2005.

[59] Y. Lin and E. D. Sontag. A universal formula for stabilization with bounded controls. *Systems and Control Letters*, 16:393–397, 1991.

[60] Y. Lin, E. D. Sontag, and Y. Wang. A smooth converse Lyapunov theorem for robust stability. *SIAM Journal on Control and Optimization*, 34:124–160, 1996.

[61] G.-P. Liu, Y. Xia, J. Chen, D. Rees, and W. Hu. Networked predictive control of systems with random networked delays in both forward and feedback channels. *IEEE Transactions on Industrial Electronics*, 54:1282–1297, 2007.

[62] J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Iterative distributed model predictive control of nonlinear systems: Handling delayed measurements. In *Proceedings of the 49th IEEE Conference on Decision and Control*, pages 7251–7258, Atlanta, Georgia, 2010.

[63] J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Sequential and iterative architectures for distributed model predictive control of nonlinear process systems. *AIChE Journal*, 56:2137–2149, 2010.

[64] J. Liu, X. Chen, D. Muñoz de la Peña, and P. D. Christofides. Iterative distributed model predictive control of nonlinear systems: Handling asynchronous, delayed measurements. *IEEE Transactions on Automatic Control*, 57:528–534, 2012.

[65] J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Distributed model predictive control of nonlinear process systems. *AIChE Journal*, 55:1171–1184, 2009.

[66] J. Liu, D. Muñoz de la Peña, and P. D. Christofides. Distributed model predictive control of nonlinear systems subject to asynchronous and delayed measurements. *Automatica*, 46:52–61, 2010.

[67] J. Liu, D. Muñoz de la Peña, P. D. Christofides, and J. F. Davis. Lyapunov-based model predictive control of nonlinear systems subject to time-varying measurement delays. *International Journal of Adaptive Control and Signal Processing*, 23:788–807, 2009.

[68] J. Liu, D. Muñoz de la Peña, B. J. Ohran, P. D. Christofides, and J. F. Davis. A two-tier architecture for networked process control. *Chemical Engineering Science*, 63:5394–5409, 2008.

[69] J. Liu, D. Muñoz de la Peña, B. J. Ohran, P. D. Christofides, and J. F. Davis. A two-tier control architecture for nonlinear process systems with continuous/asynchronous feedback. *International Journal of Control*, 83:257–272, 2010.

[70] P. Daoutidis M. Baldea and Z.K. Nagy. Nonlinear model predictive control of integrated process systems. *In NOLCOS 2010, 8th IFAC Symposium on Nonlinear Control Systems*, Bologna, Italy, 2010.

[71] J. M. Maestre, David Muñoz de la Peña, and Eduardo F. Camacho. A distributed MPC scheme with low communication requirements. In *Proceedings of the American Control Conference*, pages 2797–2802, Saint Louis, MO, USA, 2009.

[72] L. Magni and R. Scattolini. Stabilizing decentralized model predictive control of nonlinear systems. *Automatica*, 42:1231–1236, 2006.

[73] N. A. Mahmoud and H. K. Khalil. Asymptotic regulation of minimum phase nonlinear systems using output feedback. *IEEE Transactions on Automatic Control*, 41:1402–1412, 1996.

[74] T. E. Marlin and A. N. Hrymak. Real-time operations optimization of continuous processes. In *AIChE Symposium Series on CPC V*, pages 156–164, 1997.

[75] J. L. Massera. Contributions to stability theory. *Annals of Mathematics*, 64:182–206, 1956.

[76] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36:789–814, 2000.

[77] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Predictive control of switched nonlinear systems with scheduled mode transitions. *IEEE Transactions on Automatic Control*, 50:1670–1680, 2005.

[78] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Robust hybrid predictive control of nonlinear systems. *Automatica*, 41:209–217, 2005.

[79] P. Mhaskar, N. H. El-Farra, and P. D. Christofides. Stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control. *Systems and Control Letters*, 55:650–659, 2006.

[80] D. Muñoz de la Peña and P. D. Christofides. Lyapunov-based model predictive control of nonlinear systems subject to data losses. *IEEE Transactions on Automatic Control*, 53:2076–2089, 2008.

[81] P. Mhaskar N. H. El-Farra and P. D. Christofides. Output feedback control of switched nonlinear systems using multiple lyapunov functions. *Systems & Control Letters*, 54:1163–1182, 2005.

[82] D. Nešić, A Teel, and P. Kokotovic. Sufficient conditions for stabilization of sampled-data nonlinear systems via discrete time approximations. *Systems and Control Letters*, 38:259–270, 1999.

[83] P. Neumann. Communication in industrial automation: What is going on? *Control Engineering Practice*, 15:1332–1347, 2007.

[84] C. Perego and P. Ingallina. Combining alkylation and transalkylation for alkylaromatic production. *Green Chemistry*, 6:274–279, 2004.

[85] D. M. Raimondo, L. Magni, and R. Scattolini. Decentralized MPC of nonlinear system: An input-to-state stability approach. *International Journal of Robust and Nonlinear Control*, 17:1651–1667, 2007.

[86] J. B. Rawlings. Tutorial overview of model predictive control. *IEEE Control Systems Magazine*, 20:38–52, 2000.

[87] J. B. Rawlings and R. Amrit. Optimizing process economic performance using model predictive control. In L. Magni, D. M. Raimondo, and F. Allgöwer, editors, *Nonlinear Model Predictive Control, Lecture Notes in Control and Information Science Series*, volume 384, pages 119–138, Berlin, 2009. Springer.

[88] J. B. Rawlings and K. R. Muske. The stability of constrained receding horizon control. *IEEE Transactions on Automatic Control*, 38:1512–1516, 1993.

[89] J. B. Rawlings and B. T. Stewart. Coordinating multiple optimization-based controllers: New opportunities and chanllenges. In *Procedings of the 8th IFAC Symposium on Dynamics and Control of Process*, volume 1, pages 19–28, Cancun, Mexico, 2007.

[90] J. B. Rawlings and B. T. Stewart. Coordinating multiple optimization-based controllers: New opportunities and challenges. *Journal of Process Control*, 18:839–845, 2008.

[91] A. Richards and J. P. How. Robust distributed model predictive control. *International Journal of Control*, 80:1517–1531, 2007.

[92] R. Scattolini. Architectures for distributed and hierarchical model predictive control - A review. *Journal of Process Control*, 19:723–731, 2009.

[93] E. Sontag. A 'universal' construction of Artstein's theorem on nonlinear stabilization. *Systems and Control Letters*, 13:117–123, 1989.

[94] M. Soroush and S. Valluri. Optimal directionality compensation in processes with input saturation nonlinearities. *International Journal of Control*, 72:1555–1564, 1999.

[95] B. T. Stewart, A. N. Venkat, J. B. Rawlings, S. J. Wright, and G. Pannocchia. Coorperative distributed model predictive control. *Systems and Control Letters*, 59:460–469, 2010.

[96] A. R. Teel. Global stabilization and restricted tracking for multiple integrators with bounded controls. *System and Control Letters*, 18:165–171, 1992.

[97] S. Valluri and M. Soroush. Analytical control of SISO nonlinear processes with input constraints. *AIChE Journal*, 44:116–130, 1998.

[98] S. Valluri, M. Soroush, and M. Nikravesh. Shortest-prediction-horizon nonlinear model-predictive control. *Chemical Engineering Science*, 53:273–292, 1998.

[99] E. J. Van Henten and J. Bontsema. Time-scale decomposition of an optimal control problem in greenhouse climate management. *Control Engineering Practice*, 17:88–96, 2009.

[100] A. N. Venkat, J. B. Rawlings, and S. J. Wright. Stability and optimality of distributed model predictive control. In *Proceedings of the 44th IEEE Conference on Decision and Control and the European Control Conference ECC 2005*, pages 6680–6685, Seville, Spain, 2005.

[101] A. Wächter and L. T. Biegler. On the implementation of primal-dual interior point filter line search algorithm for large-scale nonlinear programming. *Mathematical Programming*, 106:25–57, 2006.

[102] G. Walsh, O. Beldiman, and L. Bushnell. Asymptotic behavior of nonlinear networked control systems. *IEEE Transactions on Automatic Control*, 46:1093–1097, 2001.

[103] G. Walsh, H. Ye, and L. Bushnell. Stability analysis of networked control systems. *IEEE Transactions on Control Systems Technology*, 10:438–446, 2002.

[104] M. Wogrin and L. Glielmo. An MPC scheme with guaranteed stability for linear singularly perturbed systems. In *Proceedings of 49th IEEE Conference on Decision and Control*, pages 5289–5295, Atlanta, GA, USA, 2010.

[105] Guy B. Woodle. *Ethylbenzene*, volume I, chapter Petrochemicals and Petrochemical Processing, pages 929–941. Taylor & Francis Group, New York, 2006.

[106] E. C. Yamalidou and J. Kantor. Modeling and optimal control of discrete-event chemical processes using petri nets. *Comp. & Chem. Eng.*, 15:503–519, 1990.

[107] E. B. Ydstie. Certainty equivalence adaptive control: Paradigms puzzles and switching. In *Proceedings of 5th International Conference on Chemical Process Control*, pages 9–23, Tahoe City, CA, 1997.

[108] E. B. Ydstie. New vistas for process control: Integrating physics and communication networks. *AIChE Journal*, 48:422–426, 2002.

[109] H. Ye, A. N. Michel, and L. Hou. Stability theory for hybrid dynamical systems. *IEEE Transactions on Automatic Control*, 43:461–474, 1998.

[110] H. You, W. Long, and Y. Pan. The mechanism and kinetics for the alkylation of benzene with ethylene. *Petroleum Science and Technology*, 24:1079–1088, 2006.