

# Chapter 6

## Two-Layer EMPC Systems

### 6.1 Introduction

As discussed in Chap. 1, in the traditional paradigm to optimization and control, a hierarchical strategy is employed using real-time optimization (RTO) to compute economically optimal steady-states that are subsequently sent down to a tracking MPC layer. The tracking MPC computes control actions that are applied to the closed-loop system to force the state to the optimal steady-state. RTO may periodically update the optimal steady-state to account for time-varying factors that may shift the optimal operating conditions and send the updated steady-state to the MPC layer. On the other hand, EMPC merges economic optimization and control and thus, employs a one-layer approach to optimization and control. While EMPC merges optimization and control, the extent that EMPC takes on all the responsibilities of RTO remains to be seen. For example, many EMPC methods are formulated using a steady-state, which potentially could be the economically optimal steady-state. RTO is also responsible for other tasks besides economic optimization. Therefore, one may envision that future optimization and control structures will maintain some aspects of the hierarchical approach within the context of industrial applications. Moreover, in some applications, maintaining a division between economic optimization and control is suitable, especially for applications where there is an explicit time-scale separation between the process/system dynamics and the update frequency or time-scale of evolution of economic factors and/or other factors that shift optimal operating conditions, e.g., disturbances.

In an industrial control architecture, which features a high degree of complexity, a hierarchical approach to dynamic economic optimization and control may be more applicable. Motivated by the aforementioned considerations, several two-layer approaches to dynamic economic optimization and control are discussed in this chapter. The upper layer, utilizing an EMPC, is used to compute economically optimal policies and potentially, also, control actions that are applied to the closed-loop system. The economically optimal policies are sent down to a lower layer MPC scheme which may be a tracking MPC or an EMPC. The lower layer MPC scheme

forces the closed-loop state to closely follow the economically optimal policy computed in the upper layer EMPC.

The unifying themes of the two-layer EMPC implementations described in this chapter are as follows. First, the upper layer EMPC may employ a long prediction horizon. The long prediction horizon ideally prevents the EMPC from dictating an operating policy based on myopic decision-making, which may lead to poor closed-loop economic performance. Considering a one-layer EMPC approach with a long horizon, the computational time and complexity of the resulting optimization problem (thousands of decisions variables for large-scale systems) may make it unsuitable for real-time application. Second, the upper layer dynamic economic optimization problem, i.e., the EMPC problem, is formulated with explicit control-oriented constraints which allow for guaranteed closed-loop stability properties. This is a departure from other two-layer approaches to dynamic optimization and control such as those featuring dynamic-RTO, e.g. [1–10]. Third, the upper layer is solved infrequently in the sense that it is not solved every sampling time like a standard one-layer EMPC method with a receding horizon implementation. The rate that the upper layer EMPC is solved may be considered a tuning parameter of the optimization and control architectures. However, the upper layer does not need to wait until the system has reached steady-state owing to the fact that a dynamic model of the process is used in the optimization layer.

The lower layer MPC, which may be either a tracking MPC or an EMPC, may be formulated with a shorter prediction horizon and potentially, a smaller sampling period if state measurement feedback is available. It is used to force the closed-loop state to track the operating policy computed by the upper layer EMPC and to ensure closed-loop stability and robustness. Owing to the fact that the upper layer EMPC is solved infrequently and the lower layer MPC utilizes a shorter prediction horizon, one of the benefits of a two-layer EMPC implementation is improved computational efficiency compared to a one-layer EMPC method. The results of this chapter originally appeared in [11–13].

### 6.1.1 Notation

Given that this chapter deals with control elements arranged in a multi-layer configuration, an extended amount of notation is needed to describe the control system. To aid the reader, Table 6.1 summarizes the notation used in this chapter. Some of the notation will be made more precise in what follows. To clarify the difference between open-loop and closed-loop trajectories, consider a time sequence:  $\{\tilde{t}_i\}_{i=k}^{k+\tilde{N}}$  where  $\tilde{t}_i = i\tilde{\Delta}$ ,  $\tilde{\Delta} > 0$  is a constant and  $\tilde{N} \geq 1$  is a positive integer. Given a function  $\tilde{u} : [\tilde{t}_k, \tilde{t}_{k+\tilde{N}}) \rightarrow \mathbb{U}$ , which is right-continuous piecewise constant with constant hold period  $\tilde{\Delta}$ , the open-loop predicted state trajectory under the open-loop input trajectory  $\tilde{u}$  is the solution to the differential equation:

**Table 6.1** A summary of the notation used to describe the two-layer EMPC structure

Notation	Description
$\Delta_E$	Zeroth-order hold period used for the upper layer control parameterization
$N_E$	Number of zeroth-order hold periods in the upper layer EMPC prediction horizon
$K_E$	Number of hold periods, $\Delta_E$ , that the upper layer EMPC is solved
$t'$	Operating period length with $t' = K_E \Delta_E$
$\{\hat{t}_k\}_{k \geq 0}$	Computation time sequence of upper layer with $\hat{t}_k = kt'$ ( $k \in \mathbb{I}_+$ )
$z(\cdot \hat{t}_k)$	Open-loop predicted state trajectory under an auxiliary controller computed at $\hat{t}_k$
$v(\cdot \hat{t}_k)$	Open-loop input trajectory computed by an auxiliary controller computed at $\hat{t}_k$
$x_E(\cdot \hat{t}_k)$	Open-loop predicted state trajectory under the upper layer MPC at $\hat{t}_k$
$u_E(\cdot \hat{t}_k)$	Open-loop input trajectory computed by the upper layer MPC at $\hat{t}_k$
$\Delta$	Sampling period size of the lower layer
$N$	Number of sampling periods in the lower layer MPC prediction horizon
$\{t_j\}_{j \geq 0}$	Sampling time sequence of lower layer with $t_j = j\Delta$ ( $j \in \mathbb{I}_+$ )
$\bar{x}(\cdot t_j)$	Open-loop predicted state trajectory under the lower layer MPC at $t_j$
$u(\cdot t_j)$	Open-loop input trajectory computed by the upper layer MPC at $t_j$
$x(\cdot)$	Closed-loop state trajectory under the two-layer control structure
$u^*(t_j t_j)$	Control action applied to the closed-loop system computed at $t_j$ and applied from $t_j$ to $t_{j+1}$
$(x_s, u_s)$	Steady-state and steady-state input pair

$$\dot{\bar{x}}(t) = f(\bar{x}(t), \bar{u}(t), 0) \quad (6.1)$$

for  $t \in [\bar{t}_k, \bar{t}_{k+N})$  with initial condition  $\bar{x}(\bar{t}_k) = x(\bar{t}_k)$  where  $x(\bar{t}_k)$  is a state measurement of the closed-loop system at time  $\bar{t}_k$ . The open-loop predicted state and input trajectories are denoted as  $\bar{x}(\cdot|\bar{t}_k)$  and  $\bar{u}(\cdot|\bar{t}_k)$  to make clear that both of these trajectories, which are functions of time, have been computed at  $\bar{t}_k$  with a state measurement at  $\bar{t}_k$ .

The term closed-loop system refers to the resulting sampled-data system of Eq. 4.1 under an MPC scheme. The closed-loop state trajectory is the solution to:

$$\dot{x}(t) = f(x(t), k(x(t_j)), w(t)) \quad (6.2)$$

for  $t \in [t_j, t_{j+1})$  with  $t_j = j\Delta$  for some  $\Delta > 0$  and  $j = 0, 1, \dots$ . The initial time is taken to be zero. The mapping  $k(\cdot)$  is a state feedback control law.

In the context of MPC, the state feedback control law is implicitly defined from the solution of an optimization problem and the receding horizon implementation. Specifically, the MPC receives a state measurement at a sampling time  $t_j$ , computes a control action, and applies it in a sample-and-hold fashion over the sample period, i.e., from  $t_j$  to  $t_{j+1}$ . The notation  $u^*(t_j|t_j)$  is used to denote the computed control action by the MPC scheme at sampling time  $t_j$  with a state measurement  $x(t_j)$ . Under an MPC scheme, the closed-loop system is written similarly to Eq. 6.2 by replacing

$k(x(t_j))$  with  $u^*(t_j|t_j)$ . Finally, the notation  $\cdot^*$ , e.g.,  $y^*$ , is used to denote that the quantity, which may be a vector with real elements or a function defined over an appropriate domain and range, is optimal with respect to a cost function (or cost functional) and some constraints.

## 6.2 Two-Layer Control and Optimization Framework

In this section, a two-layer dynamic economic optimization and control framework featuring EMPC in the upper layer and tracking MPC in the lower layer is discussed. The same nonlinear dynamic model is used in each layer to avoid modeling inconsistencies. Control-oriented constraints are employed in the dynamic optimization layer to ensure closed-loop stability. A rigorous theoretical treatment of the stability properties of the closed-loop system with the control architecture is provided. Variants and extensions of the two-layer optimization and control framework are discussed. The two-layer optimization and control framework is applied to a chemical process example.

### 6.2.1 Class of Systems

While a similar class of nonlinear systems is considered as that described by Eq. 4.1, the manipulated inputs are split into two groups; that is, the input vector is given by  $u := [u_1 \ u_2]^T$  where  $u_1 \in \mathbb{R}^{m_1}$ ,  $u_2 \in \mathbb{R}^{m_2}$ , and  $m_1 + m_2 = m$ . Loosely speaking, the inputs are partitioned into two groups based on their main responsibility. The input  $u_1$  is directly responsible for economic optimization and/or has the most significant impact on the closed-loop economic performance, while the input  $u_2$  is responsible for maintaining closed-loop stability. In the chemical process example of Sect. 6.2.3, the inputs are partitioned using this rationale as a basis. Additional methods may be employed to help identify the inputs that have the most significant impact on the economic performance such as the methods presented in [14].

With the two sets of inputs, the following state-space model is written to emphasize the dependence of the vector field on each group of inputs:

$$\dot{x} = f(x, u_1, u_2, w) \quad (6.3)$$

where  $x \in \mathbb{X} \subseteq \mathbb{R}^n$  denotes the state vector,  $u_1 \in \mathbb{U}_1 \subset \mathbb{R}^{m_1}$  and  $u_2 \in \mathbb{U}_2 \subset \mathbb{R}^{m_2}$  denote the two manipulated input vectors or the two sets of manipulated inputs,  $w \in \mathbb{W} \subset \mathbb{R}^l$  denotes the disturbance vector and  $f$  is assumed to be a locally Lipschitz vector function on  $\mathbb{X} \times \mathbb{U}_1 \times \mathbb{U}_2 \times \mathbb{W}$ . The sets  $\mathbb{U}_1$  and  $\mathbb{U}_2$  are assumed to be nonempty compact sets. The disturbance is assumed to be bounded, i.e.,  $\mathbb{W} := \{w \in \mathbb{R}^l : |w| \leq \theta\}$  where  $\theta > 0$ . The origin of the nominal unforced system of Eq. 6.3 is assumed to be an equilibrium point ( $f(0, 0, 0, 0) = 0$ ). The state

of the system is sampled synchronously at the time instants indicated by the time sequence  $\{t_j\}_{j \geq 0}$  where  $t_j = j\Delta$ ,  $j = 0, 1, \dots$  and  $\Delta > 0$  is the sampling period.

A stabilizability assumption is imposed on the system of Eq. 6.3 in the sense that the existence of a stabilizing feedback control law that renders the origin of the system of Eq. 6.3 asymptotically stable is assumed. The stabilizing feedback control law is given by the pair:

$$(h_1(x), h_2(x)) \in \mathbb{U}_1 \times \mathbb{U}_2 \quad (6.4)$$

for all  $x \in \mathbb{X}$ . While the domain of the stabilizing controller is taken to be  $\mathbb{X}$ , it renders the origin asymptotically stable with some region of attraction that may be a subset of  $\mathbb{X}$ . Applying converse Lyapunov theorems [15, 16], there exists a continuous differentiable Lyapunov function  $V : D \rightarrow \mathbb{R}_+$  that satisfies the following inequalities:

$$\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \quad (6.5a)$$

$$\frac{\partial V(x)}{\partial x} f(x, h_1(x), h_2(x), 0) \leq -\alpha_3(|x|) \quad (6.5b)$$

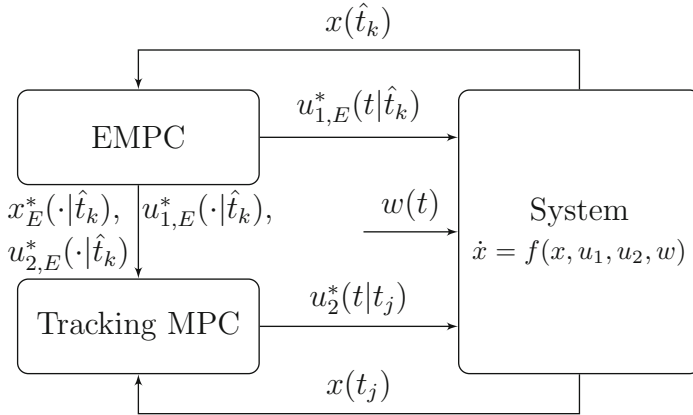
$$\left| \frac{\partial V(x)}{\partial x} \right| \leq \alpha_4(|x|) \quad (6.5c)$$

for all  $x \in D$  where  $\alpha_i \in \mathcal{K}$  for  $i = 1, 2, 3, 4$  and  $D$  is an open neighborhood of the origin. The region  $\Omega_\rho \subseteq D$  such that  $\Omega_\rho \subseteq \mathbb{X}$  is the (estimated) stability region of the closed-loop system under the stabilizing controller.

## 6.2.2 Formulation and Implementation

The dynamic economic optimization and control framework consists of EMPC in the upper layer and tracking MPC in the lower layer. A block diagram of the framework is given in Fig. 6.1. The prediction horizons of the EMPC and the MPC may be different. This allows for the EMPC to be formulated with a long prediction horizon. The number of sampling periods in the prediction horizon of the EMPC is denoted as  $N_E \in \mathbb{I}_{\geq 1}$ , and that of the MPC is denoted as  $N \in \mathbb{I}_{\geq 1}$ . For simplicity, the sampling periods of the upper layer EMPC and lower layer MPC are assumed to be the same ( $\Delta_E = \Delta$ ) and  $\Delta$  will be used to denote the sampling period. The two-layer framework may be extended to the case where  $\Delta_E > \Delta$ .

The upper layer EMPC problem is solved infrequently, i.e., not every sampling time. Let  $K_E \geq \mathbb{I}_+$  be the number of sampling times that the upper layer is resolved. The time sequence  $\{\hat{t}_k\}_{k \geq 0}$  denotes the time steps that the upper layer EMPC is solved. Owing to the implementation strategy, the time sequence is not necessarily a synchronous partitioning of time. For sake of simplicity, let  $N \leq N_E - K_E$  to ensure that the upper layer EMPC problem is computed at a rate needed to ensure that the economically optimal trajectory is defined over the prediction horizon of



**Fig. 6.1** A block diagram of the two-layer integrated framework for dynamic economic optimization and control with EMPC in the upper layer and tracking MPC in the lower layer. Both the upper and lower layers compute control actions that are applied to the system

the lower layer tracking MPC. If this is not satisfied, one could employ a shrinking horizon in the lower layer MPC. While the upper layer EMPC computes optimal input trajectories for both sets of manipulated inputs, it sends control actions for the manipulated input  $u_1$  to the control actuators to be applied in an open-loop fashion. The implementation strategy is described below. The optimal operating trajectory over the prediction horizon of the EMPC is computed by the upper layer EMPC and sent to the lower layer tracking MPC to force the closed-loop state to track the optimal operating trajectory. In other words, the upper layer EMPC trajectory is used as the reference trajectory in the tracking MPC. The optimal operating trajectory is defined below.

**Definition 6.1** Let  $(u_{1,E}^*(t|\hat{t}_k), u_{2,E}^*(t|\hat{t}_k))$ , which is defined for  $t \in [\hat{t}_k, \hat{t}_k + N_E \Delta)$ , be the optimal input pair computed by the upper layer EMPC and let  $x(\hat{t}_k)$  be the state measurement at the sampling time  $\hat{t}_k$ . The economically optimal state trajectory  $x_E^*(t|\hat{t}_k)$  for  $t \in [\hat{t}_k, \hat{t}_k + N_E \Delta)$  of the system of Eq. 6.3 is the solution of

$$\dot{x}_E^*(t) = f(x_E^*(t), u_{1,E}^*(\tau_i|\hat{t}_k), u_{2,E}^*(\tau_i|\hat{t}_k), 0), \quad t \in [\tau_i, \tau_{i+1}) \quad (6.6)$$

for  $i = 0, 1, \dots, N_E - 1$  with  $x_E(\hat{t}_k) = x(\hat{t}_k)$  where  $\tau_i := \hat{t}_k + i \Delta$ .

The lower layer MPC is implemented with a receding horizon implementation, i.e., is solved at every sampling time. The notation  $t_j$  will be reserved to denote a sampling time that the lower layer MPC problem is solved. To provide closed-loop stability guarantees on the resulting control framework, the upper layer EMPC is formulated as an LEMPC (Eq. 4.3) and the lower layer tracking MPC is formulated as an LMPC (Eq. 2.51). The advantage of the formulation of the upper and lower

layer controllers is that the LEMPC computes a reference trajectory that the tracking LMPC layer may force the system to track and unreachable set-points are avoided.

An assumption is needed to ensure feasibility and stability with the resulting two-layer framework. Owing to the fact that the upper layer EMPC is solved infrequently and applies its computed trajectory for  $u_1$  in an open-loop fashion, an assumption is needed to ensure that it is possible to maintain stability with the input  $u_2$  in the sense that for any  $u_1 \in \mathbb{U}_1$  it is possible to find a  $u_2 \in \mathbb{U}_2$  that ensures that the time-derivative of the Lyapunov function is negative. This is stated in the following assumption. Also, the assumption further clarifies how the manipulated inputs are divided into the two input groups.

**Assumption 6.1** For any fixed  $u_{1,E} \in U_1$ , there exists  $u_2 \in U_2$  such that:

$$\frac{\partial V(x)}{\partial x} f(x, u_{1,E}, u_2, 0) \leq \frac{\partial V(x)}{\partial x} f(x, h_1(x), h_2(x), 0) \quad (6.7)$$

for all  $x \in \Omega_\rho$ .

Variations of the assumption are discussed in Sect. 6.2.2.1. This assumption is not needed for the case that the upper layer LEMPC does not apply any control actions to the system.

The upper layer LEMPC has a similar formulation as that of Eq. 4.3 with one modification discussed below. The upper layer LEMPC problem is given by the following optimization problem:

$$\min_{u_{1,E}, u_{2,E} \in S(\Delta)} \int_{\hat{t}_k}^{\hat{t}_k + N_E \Delta} l_e(x_E(\tau), u_{1,E}(\tau), u_{2,E}(\tau)) d\tau \quad (6.8a)$$

$$\text{s.t. } \dot{x}_E(t) = f(x_E(t), u_{1,E}(t), u_{2,E}(t), 0) \quad (6.8b)$$

$$x_E(\hat{t}_k) = x(\hat{t}_k) \quad (6.8c)$$

$$u_{1,E}(t) \in \mathbb{U}_1, u_{2,E}(t) \in \mathbb{U}_2, \forall t \in [\hat{t}_k, \hat{t}_k + N_E \Delta) \quad (6.8d)$$

$$V(x_E(t)) \leq \rho_e, \forall t \in [\hat{t}_k, \hat{t}_k + N_E \Delta), \\ \text{if } V(x(\hat{t}_k)) \leq \rho_e \text{ and } \hat{t}_k < t_s \quad (6.8e)$$

$$\frac{\partial V(x_E(\tau_i))}{\partial x} f(x_E(\tau_i), u_{1,E}(\tau_i), u_{2,E}(\tau_i), 0) \\ \leq \frac{\partial V(x_E(\tau_i))}{\partial x} f(x_E(\tau_i), h_1(x_E(\tau_i)), h_2(x_E(\tau_i)), 0), \\ i = 0, 1, \dots, N_E - 1, \text{ if } V(x(\hat{t}_k)) > \rho_e \text{ or } \hat{t}_k \geq t_s \quad (6.8f)$$

where  $\tau_i := \hat{t}_k + i \Delta$ . The main difference between the upper layer LEMPC formulation and the LEMPC formulation of Eq. 4.3 is the mode 2 contractive constraint (Eq. 6.8f). In the upper layer LEMPC formulation, the mode 2 contractive constraint is imposed at each time instance of the prediction horizon. This ensures that the Lyapunov function value decays over the prediction horizon and thus, the lower layer LMPC attempts to force the closed-loop state along a reference trajectory that either

converges to  $\Omega_{\rho_e}$  if  $\hat{t}_k < t_s$  or converges to a neighborhood of the origin if  $\hat{t}_k \geq t_s$ . The optimal trajectories computed by the upper layer LEMPC are denoted by  $x_E^*(t|\hat{t}_k)$ ,  $u_{E,1}^*(t|\hat{t}_k)$ , and  $u_{E,2}^*(t|\hat{t}_k)$  and are defined for  $t \in [\hat{t}_k, \hat{t}_k + N_E \Delta]$ .

The stage cost function used in the LMPC is formulated to penalize deviations of the state and inputs from the economically optimal trajectories. Additionally, the LMPC is equipped with dual-mode constraints similar to that imposed in LEMPC. The dual-mode LMPC problem is given by the following optimization problem:

$$\min_{u_2 \in S(\Delta)} \int_{t_j}^{t_{j+N}} l_T(\tilde{x}(\tau), x_E^*(\tau|\hat{t}_k), u_2(\tau), u_{2,E}^*(\tau|\hat{t}_k)) d\tau \quad (6.9a)$$

$$\text{s.t. } \dot{\tilde{x}}(t) = f(\tilde{x}(t), u_{1,E}^*(t|\hat{t}_k), u_2(t), 0) \quad (6.9b)$$

$$\tilde{x}(t_j) = x(t_j) \quad (6.9c)$$

$$u_2(t) \in \mathbb{U}_2, \forall t \in [t_j, t_{j+N}] \quad (6.9d)$$

$$V(\tilde{x}(t)) \leq \rho_e, \forall t \in [t_j, t_{j+N}], \\ \text{if } V(x(t_j)) \leq \rho_e \text{ and } t_j < t_s \quad (6.9e)$$

$$\frac{\partial V(x(t_j))}{\partial x} f(x(t_j), u_{1,E}^*(t_j|\hat{t}_k), u_2(t_j), 0) \\ \leq \frac{\partial V(x(t_j))}{\partial x} f(x(t_j), h_1(x(t_j)), h_2(x(t_j)), 0), \\ \text{if } V(x(t_j)) > \rho_e \text{ and } t_j \geq t_s \quad (6.9f)$$

where the stage cost of LMPC is given by:

$$l_T(\tilde{x}, x_E^*, u_2, u_{2,E}^*) = |\tilde{x} - x_E^*|_{Q_c}^2 + |u_2 - u_{2,E}^*|_{R_c}^2 \quad (6.10)$$

and  $Q_c$  and  $R_c$  are positive definite tuning matrices. The constraint of Eq. 6.9e defines mode 1 operation of the LMPC and serves a similar purpose as the mode 1 constraint of LEMPC (Eq. 4.3e). Under mode 2 operation of the LMPC, which is defined when the constraint of Eq. 6.9f is active, the LMPC computes control actions to ensure that the contractive Lyapunov-based constraint is satisfied. The optimal solution of Eq. 6.9 is denoted as  $u_2^*(t|t_j)$  for  $t \in [t_j, t_{j+N}]$ .

The two-layer optimization and control framework has a number of tunable parameters. Specifically, the tuning parameters include the weighting matrices  $Q_c$  and  $R_c$ , the prediction horizons  $N$  and  $N_E$ , the number of sampling times that the upper layer recomputes a solution  $K_E$ , the subset of the stability region that the control framework may operate the system in a time-varying fashion ( $\Omega_{\rho_e}$ ), the sampling period  $\Delta$  and the triple  $(h_1, h_2, V)$ , i.e., stabilizing controller design and Lyapunov function which are used in the Lyapunov-based constraints.

If the optimal state trajectory has been computed using mode 2 operation of the LEMPC and the current time is less than the switching time  $t_s$ , it is advantageous from a performance perspective, to recompute a new LEMPC solution using mode 1 once the state converges to the set  $\Omega_{\rho_e}$ . This is captured in the implementation strategy of the two-layer optimization and control framework that is described by the



following algorithm. Let the index  $l \in \mathbb{I}_+$  be the number of sampling times since the last time that the upper layer LEMPC problem has been solved and  $m_k$  be the mode of operation of the LEMPC used to solve the LEMPC problem at  $\hat{t}_k$ . To initialize the algorithm, let  $\hat{t}_0 = 0$ ,  $k = 0$ , and  $j = 0$ .

1. At  $\hat{t}_k$ , the upper layer LEMPC receives a state measurement  $x(\hat{t}_k)$  and set  $l = 0$ . If  $x(\hat{t}_k) \in \Omega_{\rho_e}$  and  $\hat{t}_k < t_s$ , go to Step 1.1. Else, go to Step 1.2.
  - 1.1 The mode 1 constraint (Eq. 6.8e) is active and the mode 2 constraint (Eq. 6.8f) is inactive. Set  $m_k = 1$  and go to Step 1.3.
  - 1.2 The mode 2 constraint (Eq. 6.8f) is active and the mode 1 constraint (Eq. 6.8e) is inactive. Set  $m_k = 2$  and go to Step 1.3.
  - 1.3 Solve the optimization problem of Eq. 6.8 to compute the optimal trajectories  $x_E^*(t|\hat{t}_k)$ ,  $u_{E,1}^*(t|\hat{t}_k)$ , and  $u_{E,2}^*(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_k + N_E \Delta)$ . Send these trajectories to the lower layer LMPC and go to Step 2.
2. At  $t_j$ , the lower layer LMPC receives a state measurement  $x(t_j)$ . If  $x(t_j) \in \Omega_{\rho_e}$ ,  $t_j < t_s$ , and  $m_k = 2$ , set  $\hat{t}_{k+1} = t_j$  and  $k \leftarrow k + 1$ , and go to Step 1. Else if  $x(t_j) \in \Omega_{\rho_e}$  and  $t_j < t_s$ , go to Step 2.1. Else, go to Step 2.2.
  - 2.1 The mode 1 constraint (Eq. 6.9e) is active and the mode 2 constraint (Eq. 6.9f) is inactive. Go to Step 2.3.
  - 2.2 The mode 2 constraint (Eq. 6.9f) is active and the mode 1 constraint (Eq. 6.9e) is inactive. Go to Step 2.3.
  - 2.3 Solve the optimization problem of Eq. 6.9 to compute the optimal input trajectory  $u^*(t|t_j)$  defined for  $t \in [t_j, t_{j+N})$ . Apply the input pair  $(u_{1,E}^*(t_j|\hat{t}_k), u_{2,E}^*(t_j|\hat{t}_k))$  to the system of Eq. 6.3 from  $t_j$  to  $t_{j+1}$ . Go to Step 3.
3. If  $l + 1 = K_E$ , set  $\hat{t}_{k+1} = t_{j+1}$ ,  $k \leftarrow k + 1$ , and  $j \leftarrow j + 1$ , and go to Step 1. Else, go to Step 2 and set  $l \leftarrow l + 1$  and  $j \leftarrow j + 1$ .

The two-layer implementation strategy allows for computational advantages over one-layer EMPC structures. When the LEMPC is operating in mode 1, the LEMPC problem is only computed once every  $K_E$  sampling times. The LMPC is less computationally expensive to solve than the LEMPC because the LMPC does not compute control actions for all of the manipulated inputs. Additionally, the LMPC may use a smaller prediction horizon than the LEMPC. Owing to these considerations, the two-layer framework is more computationally efficient compared to one-layer EMPC structures.

It is important to point out the two limiting cases of the optimization and control framework. If all the inputs are in the group  $u_2$ , then the control framework is reminiscent of current two-layer frameworks where economic optimization, which in this case is a dynamic optimization problem, and control are divided into separate layers. If, on the other hand, all inputs are placed in the group  $u_1$  or the upper layer LEMPC is solved every sampling time ( $K_E = 1$ ), then this would correspond to a one-layer implementation of LEMPC. For the case that all inputs are in the  $u_1$  group, the LEMPC would need to be computed every sampling time to ensure stability and robustness of the closed-loop system.

### 6.2.2.1 Variants of the Two-Layer Optimization and Control Framework

While the stability results presented in Sect. 6.2.2.2 apply to the two-layer framework described above, one may consider many variations to the two-layer framework design. A few such variants are listed here.

1. The upper layer EMPC does not compute input trajectories for all inputs. For instance, some inputs in the  $u_2$  set may have little impact on the economic performance. For these inputs, a constant input profile, for instance, could be assumed in the upper layer EMPC. This could further improve the computational efficiency of the two-layer framework. The inputs that are held constant in the EMPC problem could be used as additional degrees of freedom in the lower layer MPC to help force the closed-loop state to track the economically optimal state trajectory.
2. The lower layer MPC computes control actions for all the inputs, i.e., the upper layer EMPC does not apply any control actions directly to the system, but rather, is used to compute reference trajectories for the lower layer MPC. This approach is similar to current optimization and control structures but employs dynamic economic optimization with explicit control-oriented constraints imposed in the optimization layer.
3. Other assumptions to ensure feasibility and stability of the two-layer framework than Assumption 6.1 may be considered. For example, it may be possible to consider the input  $u_1$  as a perturbation to the system and derive the explicit stabilizing controller on the basis of the inputs  $u_2$ . Specifically, if there exists an explicit controller  $h_2 : \mathbb{X} \rightarrow \mathbb{U}_2$  and Lyapunov function that satisfies:

$$\frac{\partial V(x)}{\partial x} f(x, u_1, h_2(x), 0) \leq -\bar{\alpha}_3(|x|) \quad (6.11)$$

for all  $u_1 \in \mathbb{U}_1$  and  $x \in \Omega_\rho \setminus B$  where  $B \subset \Omega_\rho$  is some set containing the origin and  $\bar{\alpha}(\cdot)$  is a class  $\mathcal{K}$  function, then this assumption could be used to guarantee closed-loop stability and feasibility of the control problems. This assumption is essentially an input-to-state stability assumption of the closed-loop system of Eq. 6.3 under the controller  $h_2$  with respect to the input  $u_1$ .

4. The two-layers could use a different sampling period size. In particular, the upper layer could use a larger sampling period than the lower layer.
5. Since the conditions that guarantee closed-loop stability presented below are independent of the objective function of the lower layer MPC, the stage cost used in the lower layer MPC may be readily modified. For example, one may include rate of change penalties on the inputs and soft constraints in the stage cost function or use the economic stage cost function. One such variant employing the latter concept, i.e., use an EMPC in the lower layer, is presented later in this chapter (Sect. 6.4).

### 6.2.2.2 Stability Analysis

In this section, sufficient conditions are presented that guarantee that the closed-loop system with the two-layer dynamic economic optimization and control framework is stable in the sense that the system state remains bounded in a compact set for all times. Two propositions are needed, which are straightforward extensions of Propositions 4.1 and 4.2, respectively. The propositions are restated here for convenience. The first proposition provides an upper bound on the deviation of the open-loop state trajectory, obtained using the nominal model (Eq. 6.3 with  $w \equiv 0$ ), from the closed-loop state trajectory.

**Proposition 6.1** (Proposition 4.1) *Consider the systems*

$$\begin{aligned}\dot{x}_a(t) &= f(x_a(t), u_1(t), u_2(t), w(t)) \\ \dot{x}_b(t) &= f(x_b(t), u_1(t), u_2(t), 0)\end{aligned}\quad (6.12)$$

with initial states  $x_a(t_0) = x_b(t_0) \in \Omega_\rho$  and inputs  $u_1(t) \in \mathbb{U}_1$  and  $u_2(t) \in \mathbb{U}_2$  for  $t \geq t_0$ . If the states of the two systems are maintained in  $\Omega_\rho$  for all  $t \in [t_0, t_1]$  ( $t_1 > t_0$ ), there exists a class  $\mathcal{K}$  function  $\alpha_w(\cdot)$  such that

$$|x_a(t) - x_b(t)| \leq \alpha_w(t - t_0), \quad (6.13)$$

for all  $w(t) \in W$  and  $t \in [t_0, t_1]$ .

The following proposition bounds the difference between the Lyapunov function of two different states in  $\Omega_\rho$ .

**Proposition 6.2** (Proposition 4.2) *Consider the Lyapunov function  $V(\cdot)$  of the system of Eq. 6.3. There exists a quadratic function  $\alpha_V(\cdot)$  such that:*

$$V(x) \leq V(\hat{x}) + \alpha_V(|x - \hat{x}|) \quad (6.14)$$

for all  $x, \hat{x} \in \Omega_\rho$ .

Theorem 6.1 provides sufficient conditions such that the two-layer dynamic economic optimization and control framework guarantees that the state of the closed-loop system is always bounded in  $\Omega_\rho$ . The result is similar to that of the closed-loop stability properties under LEMPC.

**Theorem 6.1** *Consider the system of Eq. 6.3 in closed-loop under the two-layer framework with the LEMPC of Eq. 6.8 in the upper layer and the LMPC of Eq. 6.9 in the lower layer both based on the explicit stabilizing controller that satisfies Eqs. 6.5a–6.5c. Let  $\varepsilon_w > 0$ ,  $\Delta > 0$ ,  $N_E \geq 1$ ,  $N \geq 1$  ( $N \leq N_E - K_E$ ),  $\rho > \rho_e > \rho_{\min} > \rho_s > 0$ , and  $L'_x$ ,  $L'_w$  and  $M$  are positive constants (the existence of these constants follows from the assumptions on the system of Eq. 6.3) satisfy:*

$$\rho_e < \rho - \alpha_V(\alpha_w(\Delta)), \quad (6.15)$$

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + L'_x M \Delta + L'_w \theta \leq -\varepsilon_w / \Delta, \quad (6.16)$$

and

$$\rho_{\min} = \max_{s \in [0, \Delta]} \{V(x(s)) : V(x(0)) \leq \rho_s\}. \quad (6.17)$$

If  $x(0) \in \Omega_\rho$  and Assumption 6.1 is satisfied, then the state  $x(t)$  of the closed-loop system is always bounded in  $\Omega_\rho$  for all  $t \geq 0$ . Moreover, if  $t_s$  is finite, the closed-loop state is ultimately bounded in  $\Omega_{\rho_{\min}}$ .

*Proof* The proof is organized into three parts. In part 1, feasibility of the optimization problems of Eqs. 6.8 and 6.9 is proved when the state measurement given to each problem is in  $\Omega_\rho$ . In part 2, boundedness of the closed-loop state in  $\Omega_\rho$  is established. Finally, (uniform) ultimate boundedness of the closed-loop state in a small state-space set containing the origin is proved when the switching time is finite.

*Part 1:* When the closed-loop state is maintained in  $\Omega_\rho$ , which will be proved in Part 2, the sample-and-hold input trajectory obtained from the stabilizing feedback controller is a feasible solution to the upper layer LEMPC optimization problem of Eq. 6.8. Specifically, let  $\hat{x}(t)$  denote the solution at time  $t$  to the system:

$$\dot{\hat{x}}(t) = f(\hat{x}(t), h_1(\hat{x}(\tau_i)), h_2(\hat{x}(\tau_i)), 0) \quad (6.18)$$

for  $t \in [\tau_i, \tau_{i+1})$  ( $\tau_i := \hat{t}_k + i\Delta, i = 0, 1, \dots, N_E - 1$ ) with initial condition  $\hat{x}(\hat{t}_k) = x(\hat{t}_k) \in \Omega_\rho$ . Defining the pair  $(\hat{u}_1(t), \hat{u}_2(t)) := (h_1(\hat{x}(\tau_i)), h_2(\hat{x}(\tau_i)))$  for  $t \in [\tau_i, \tau_{i+1}), i = 0, 1, \dots, N_E - 1$ , the input trajectory pair  $(\hat{u}_1, \hat{u}_2)$  is a feasible solution to the LEMPC problem. Specifically, for mode 2 operation of the LEMPC, the pair  $(\hat{u}_1, \hat{u}_2)$  meets the input constraints since it is computed from the stabilizing controller, which satisfies the input constraints (Eq. 6.4). Also, the mode 2 contractive constraint of Eq. 6.8f is trivially satisfied with the input pair  $(\hat{u}_1, \hat{u}_2)$ . For mode 1 operation, the region  $\Omega_{\rho_e}$  is forward invariant under the stabilizing controller applied in a sample-and-hold fashion when  $\Omega_{\rho_{\min}} \subseteq \Omega_{\rho_e} \subset \Omega_\rho$  where  $\Omega_{\rho_{\min}}$  will be explained further in Parts 2 and 3.

If Assumption 6.1 is satisfied, the feasibility of the lower layer LMPC problem of Eq. 6.9 follows because there exists an input trajectory  $u_1(t)$  for  $t \in [t_j, t_{j+N})$  that decreases the Lyapunov function by at least the rate given by the Lyapunov-based controller at each sampling time instance along the prediction horizon. Using similar arguments as that used for feasibility of the LEMPC, mode 2 operation of the LMPC is feasible. Assumption 6.1 further implies that there exists a sample-and-hold input trajectory such that  $\Omega_{\rho_e}$  is forward invariant when  $\Omega_{\rho_{\min}} \subseteq \Omega_{\rho_e} \subset \Omega_\rho$  which guarantees that mode 1 operation of the LMPC is feasible.

*Part 2:* To show that the state is maintained in  $\Omega_\rho$  when  $x(0) \in \Omega_\rho$ , two cases must be considered. The first case occurs when the state  $x(t_j) \in \Omega_{\rho_e}$  and  $t_j < t_s$  and the second case occurs when  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_e}$  or  $t_j \geq t_s$ . It is sufficient to show that  $x(t) \in \Omega_\rho$  for all  $t \in [t_j, t_{j+1}]$ . Through recursive application of this result,

boundedness of the closed-loop state in  $\Omega_\rho$  for all  $t \geq 0$  follows if the initial state is in  $\Omega_\rho$ .

**Case 1:** If  $x(t_j) \in \Omega_{\rho_e}$  and  $t_j < t_s$ , the lower layer LMPC operates in mode 1 operation. Regardless if the upper layer LEMPC has been computed under mode 1 or mode 2, there exists a control action  $\hat{u}_2(t_j)$  such that when applied to the model of Eq. 6.9b in a sample-and-hold fashion over one sampling period the state at the next sampling time will be predicted to be in  $\Omega_{\rho_e}$  (this follows from Part 1). However, the closed-loop system of Eq. 6.3 does not evolve according to the model of Eq. 6.9b owing to the forcing of the disturbance  $w$ .

Let  $\rho_e$  satisfy Eq. 6.15. The proof proceeds by contradiction. Assume there exists a time  $\tau^* \in [t_j, t_{j+1}]$  such that  $V(x(\tau^*)) > \rho$ . Define  $\tau_1 := \inf\{\tau \in [t_j, t_{j+1}] : V(x(\tau)) > \rho\}$ . A standard continuity argument in conjunction with the fact that  $V(x(t_j)) \leq \rho_e < \rho$  shows that  $\tau_1 \in (t_j, t_{j+1}]$ ,  $V(x(t)) \leq \rho$  for all  $t \in [t_j, \tau_1]$  with  $V(x(\tau_1)) = \rho$ , and  $V(x(t)) > \rho$  for some  $t \in (\tau_1, t_{j+1}]$ . If  $\rho_e$  satisfies Eq. 6.15, then

$$\begin{aligned} \rho = V(x(\tau_1)) &\leq V(\tilde{x}(\tau_1)) + \alpha_V(\alpha_w(\tau_1)) \\ &\leq \rho_e + \alpha_V(\alpha_w(\Delta)) < \rho \end{aligned} \quad (6.19)$$

where the first inequality follows from Propositions 6.1, 6.2 and the second inequality follows from the fact that  $\alpha_V \circ \alpha_w \in \mathcal{K}$  and  $\tau_1 \leq \Delta$ . Eq. 6.19 leads to a contradiction. Thus,  $x(t_{j+1}) \in \Omega_\rho$  if Eq. 6.15 is satisfied.

**Case 2:** When  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_e}$  or  $t_j \geq t_s$ , the lower layer LMPC operates in mode 2. To cover both possibilities, consider any  $x(t_j) \in \Omega_\rho$  and that mode 2 operation of the LEMPC is active. From the constraint of Eq. 6.9f and the condition of Eq. 6.5b, the computed control action at  $t_j$  satisfies:

$$\begin{aligned} &\frac{\partial V(x(t_j))}{\partial x} f(x(t_j), u_{1,E}^*(t_j|\hat{t}_k), u_2^*(t_j|t_j), 0) \\ &\leq \frac{\partial V(x(t_j))}{\partial x} f(x(t_j), h_1(x(t_j)), h_2(x(t_j)), 0) \leq -\alpha_3(|x(t_j)|) \end{aligned} \quad (6.20)$$

where  $x(t_j)$  denotes the closed-loop state at sampling time  $t_j$ . Over the sampling period ( $\tau \in [t_j, t_{j+1})$ ), the time derivative of the Lyapunov function of the closed-loop system is given by:

$$\dot{V}(x(\tau)) = \frac{\partial V(x(\tau))}{\partial x} f(x(\tau), u_{1,E}^*(t_j|\hat{t}_k), u_2^*(t_j|t_j), w(\tau)) \quad (6.21)$$

for  $\tau \in [t_j, t_{j+1})$ . Adding and subtracting the first term of Eq. 6.20 to/from Eq. 6.21 and accounting for the bound of Eq. 6.20, the time-derivative of the Lyapunov function over the sampling period is bounded by:

$$\begin{aligned} \dot{V}(x(\tau)) \leq & -\alpha_3(|x(t_j)|) + \frac{\partial V(x(\tau))}{\partial x} f(x(\tau), u_{1,E}^*(t_j|\hat{t}_k), u_2^*(t_j|t_j), w(\tau)) \\ & - \frac{\partial V(x(t_j))}{\partial x} f(x(t_j), u_{1,E}^*(t_j|\hat{t}_k), u_2^*(t_j|t_j), 0) \end{aligned} \quad (6.22)$$

for all  $\tau \in [t_j, t_{j+1})$ .

Since the sets  $\Omega_\rho$ ,  $\mathbb{U}_1$ ,  $\mathbb{U}_2$ , and  $\mathbb{W}$  are compact, the vector field  $f$  is locally Lipschitz, and the Lyapunov function is continuously differentiable, there exist  $L'_x > 0$  and  $L'_w > 0$  such that:

$$\left| \frac{\partial V(x)}{\partial x} f(x, u_1, u_2, w) - \frac{\partial V(x')}{\partial x} f(x', u_1, u_2, 0) \right| \leq L'_x |x - x'| + L'_w |w| \quad (6.23)$$

for all  $x, x' \in \Omega_\rho$ ,  $u_1 \in \mathbb{U}_1$ ,  $u_2 \in \mathbb{U}_2$ , and  $w \in \mathbb{W}$ . From Eqs. 6.22 to 6.23 and the fact that the disturbance is bounded in  $\mathbb{W} = \{w \in \mathbb{R}^l : |w| \leq \theta\}$ , the time-derivative of the Lyapunov function over the sampling period may be bounded as follows:

$$\dot{V}(x(\tau)) \leq -\alpha_3(|x(t_j)|) + L'_x |x(\tau) - x(t_j)| + L'_w \theta \quad (6.24)$$

for all  $\tau \in [t_j, t_{j+1})$ . Again, by the fact that the sets  $\Omega_\rho$ ,  $\mathbb{U}_1$ ,  $\mathbb{U}_2$ , and  $\mathbb{W}$  are compact and the vector field  $f$  is locally Lipschitz, there exists  $M > 0$  such that

$$|f(x, u_1, u_2, w)| \leq M \quad (6.25)$$

for all  $x \in \Omega_\rho$ ,  $u_1 \in \mathbb{U}_1$ ,  $u_2 \in \mathbb{U}_2$ , and  $w \in \mathbb{W}$ . From Eq. 6.25 and continuity of  $x(\tau)$  for  $\tau \in [t_j, t_{j+1})$ , the difference between the state at  $\tau$  and  $t_j$  is bounded by:

$$|x(\tau) - x(t_j)| \leq M \Delta \quad (6.26)$$

for all  $\tau \in [t_j, t_{j+1})$ . From Eqs. 6.24 to 6.26 and for any  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_s}$ , the inequality below follows:

$$\dot{V}(x(\tau)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + L'_x M \Delta + L'_w \theta \quad (6.27)$$

for all  $\tau \in [t_j, t_{j+1})$  where the fact that  $|x| \geq \alpha_2^{-1}(\rho_s)$  for all  $x \in \Omega_\rho \setminus \Omega_{\rho_s}$  follows from Eq. 6.5a.

If the condition of Eq. 6.16 is satisfied, there exists  $\varepsilon_w > 0$  such that the following inequality holds for  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_s}$ ,

$$\dot{V}(x(\tau)) \leq -\varepsilon_w / \Delta$$

for all  $\tau \in [t_j, t_{j+1})$ . Integrating the bound for  $\tau \in [t_j, t_{j+1})$ , the following two bounds on the Lyapunov function value are obtained:

$$V(x(t_{j+1})) \leq V(x(t_j)) - \varepsilon_w \quad (6.28)$$

$$V(x(\tau)) \leq V(x(t_j)), \quad \forall \tau \in [t_j, t_{j+1}) \quad (6.29)$$

for all  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_s}$  and when mode 2 operation of the lower layer LMPC is active.

If  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_e}$ , the closed-loop state converges to  $\Omega_{\rho_e}$  in a finite number of sampling times without leaving the stability region  $\Omega_\rho$  which follows by applying Eq. 6.28 recursively. If  $t_j \geq t_s$  and  $x(t_j) \in \Omega_\rho \setminus \Omega_{\rho_s}$ , the closed-loop state converges to  $\Omega_{\rho_s}$  in a finite number of sampling times without leaving the stability region  $\Omega_\rho$  (again, by recursive application of Eq. 6.28). Moreover, once the state converges to  $\Omega_{\rho_s}$ , it remains inside  $\Omega_{\rho_{\min}}$  for all times. This statement holds by the definition of  $\rho_{\min}$ . Therefore, from Case 1 and Case 2, the closed-loop state is bounded in  $\Omega_\rho$  for all  $t \geq 0$  when  $x(0) \in \Omega_\rho$ .

*Part 3:* If  $t_s$  is finite, the lower layer LMPC will switch to mode 2 operation only and the closed-loop state will be ultimately bounded in  $\Omega_{\rho_{\min}}$ , which follows from Part 2.

*Remark 6.1* The closed-loop stability result presented in Theorem 6.1 is boundedness of the closed-loop state inside of  $\Omega_\rho$ . Additional elements in the lower layer MPC are usually needed to guarantee that the closed-loop state will track the reference trajectories computed in the upper layer LEMPC. Nevertheless, acceptable tracking performance may usually be achieved in practice through a careful tuning of the weighting matrices  $Q_c$  and  $R_c$  and a sufficiently long prediction horizon in the lower layer LMPC, which is the case in the example below.

### 6.2.3 Application to a Chemical Process

The two-layer framework for dynamic economic optimization and process control is implemented on the benchmark chemical reactor example presented in Sect. 1.3.1. Recall, the nonlinear dynamic model that describes the evolution of the reactor (Eqs. 1.7–1.10) has four states: the vapor density in the reactor ( $x_1$ ), the ethylene concentration in the reactor ( $x_2$ ), the ethylene oxide concentration in the reactor ( $x_3$ ), and the reactor temperature ( $x_4$ ) and three inputs: the volumetric flow rate of the reactor feed, the ethylene concentration in the reactor feed, and the reactant coolant temperature. With abuse of notation, the notation  $u_1$ ,  $u_2$ , and  $u_3$  is used to denote the three inputs, respectively. The reactor has an asymptotically stable steady-state:

$$x_s^T = [0.998 \ 0.424 \ 0.032 \ 1.002] \quad (6.30)$$

which corresponds to the steady-state input:

$$u_{1,s} = 0.35, \quad u_{2,s} = 0.5, \quad u_{3,s} = 1.0. \quad (6.31)$$

The control objective considered here is to optimize the time-averaged yield of ethylene oxide by operating the reactor in a time-varying fashion around the stable steady-state. Owing to the fact that closed-loop stability is not an issue for this application, the optimization and control framework operates with mode 1 operation only. The time-averaged yield of ethylene oxide over an operating length of  $t_f$  is given by

$$Y = \frac{\int_0^{t_f} x_3(t)x_4(t)u_1(t) dt}{\int_0^{t_f} u_1(t)u_2(t) dt} . \quad (6.32)$$

Owing to practical considerations, the average amount of ethylene that may be fed into the process over the length of operation is fixed, which is given by the following integral constraint:

$$\frac{1}{t_f} \int_0^{t_f} u_1(t)u_2(t)dt = u_{1,s}u_{2,s} = 0.175 \quad (6.33)$$

where  $u_{1,s}$  and  $u_{2,s}$  are the steady-state inlet volumetric flow rate and ethylene concentration, respectively. Since the average ethylene fed to the reactor is fixed, which fixes the denominator of the yield (Eq. 6.32), the economic stage cost used in the formulation of the upper layer LEMPC is

$$l_e(x, u) = -x_3x_4u_1 . \quad (6.34)$$

In the implementation of the two-layer dynamic optimization and control framework, the manipulated inputs are partitioned into two sets. The first set of manipulated inputs consists of the inlet flow rate and ethylene feed concentration inputs. As pointed out in Sect. 3.2, periodic switching of these two inputs may improve economic performance. Additionally, these two inputs are constrained by the integral constraint of Eq. 6.33. The first set of inputs is controlled by the upper layer LEMPC, i.e., the upper layer LEMPC computes control actions for these manipulated inputs that are applied to the reactor. The second set of manipulated inputs consists of the coolant temperature input that the lower layer LMPC (Eq. 6.9) controls.

To characterize the region  $\Omega_{\rho_e}$ , which is used in the two-layer framework design, an explicit stabilizing controller is designed and a Lyapunov function is constructed. Specifically, the explicit controller is designed as a proportional controller for the input  $u_3$ :  $h_2(x) = K(x_{s,3} - x_3) + u_{s,3}$  with  $K = 0.1$ . A quadratic Lyapunov function is found for the closed-loop reactor under the proportional controller, which is given by:

$$V(x) = (x - x_s)^T P (x - x_s)$$

where  $P = \text{diag}([10 \ 0.01 \ 10 \ 10])$ . The closed-loop stability region of the reactor under the explicit controller with the inputs  $u_1$  and  $u_2$  fixed at their steady-state



values is taken to be a level set of the Lyapunov function where the time-derivative of the Lyapunov function is negative definite for all points contained in the level set. The constructed level set is subsequently taken to be  $\Omega_{\rho_e}$  with  $\rho_e = 0.53$  and in this case,  $\Omega_{\rho} = \Omega_{\rho_e}$ .

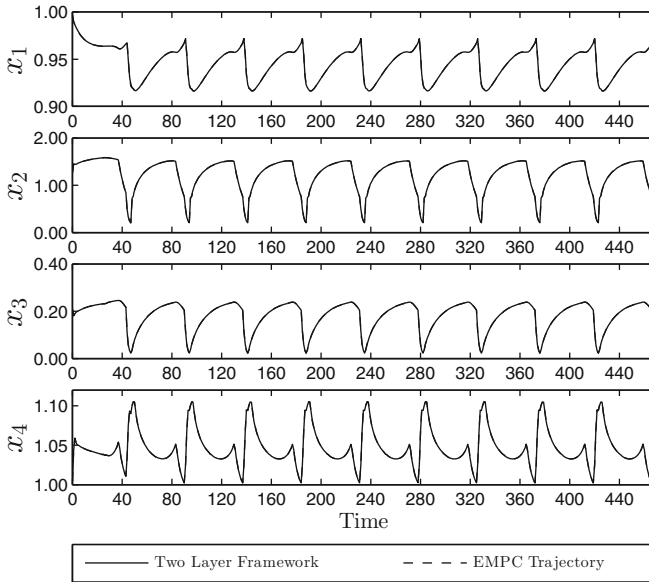
The prediction horizon of the upper layer LEMPC and lower layer LMPC are  $N_E = 47$  and  $N = 3$ , respectively, the sampling period is  $\Delta = 1.0$ , the number of sampling times that the upper layer LEMPC is recomputed is  $K_E = 47$ , which is the same as the prediction horizon in this case, and a shrinking horizon employed in the lower layer LMPC when the prediction horizon extends past the time that the upper layer optimal trajectory is defined. To ensure that the integral constraint of Eq. 6.33 is satisfied over the length of operation, the computed input trajectory of the upper layer LEMPC must satisfy the integral constraint, i.e., it is enforced over each operating windows of length 47 (dimensionless time). The weighting matrices of the lower layer LMPC are  $Q_c = P$ , and  $R_c = 0.01$  which have been tuned to achieve close tracking of the optimal trajectory. The optimization problems of upper layer LEMPC and lower layer LMPC are solved using Ipopt [17].

In the first set of simulations, the two-layer framework is applied to the reactor without disturbances or plant-model mismatch. The reactor is initialized at a transient initial condition given by:

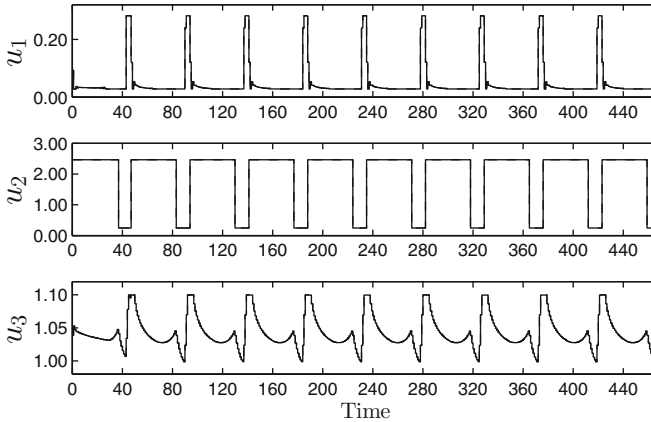
$$x_0^T = [0.997 \ 1.264 \ 0.209 \ 1.004] .$$

The closed-loop state and input trajectories of the reactor under the two-layer optimization and control framework are shown in Figs. 6.2 and 6.3, respectively. From the state trajectories (Fig. 6.2), the lower layer LMPC is able to force the system to track the optimal state trajectory. Recall, the sampling periods of the upper and lower layer are the same, and the closed-loop system is not subjected to any uncertainties or disturbances. Therefore, perfect tracking of the optimal trajectory is expected.

As described above, a motivating factor for the design of a two-layer optimization and control architecture is to achieve a computation benefit relative to a one-layer EMPC approach. To compare the computational time of the two-layer framework with a one-layer EMPC approach, a one-layer LEMPC implementation is considered. The LEMPC is implemented with mode 1 operation only and with a shrinking prediction horizon. The shrinking horizon implementation is described as follows: the LEMPC is initialized with a prediction horizon of 47 (dimensionless time) and at every subsequent sampling time, the prediction horizon is decreased by one sampling period. Every 47 sampling times, the prediction horizon is reset to 47. It is important to point out that the closed-loop performance achieved under the two-layer LEMPC and that under the one-layer LEMPC are equal owing to the fact there is no plant-model mismatch. Also, a fixed-horizon one-layer LEMPC implementation strategy requires more computation time on average relative to the shrinking horizon implementation.



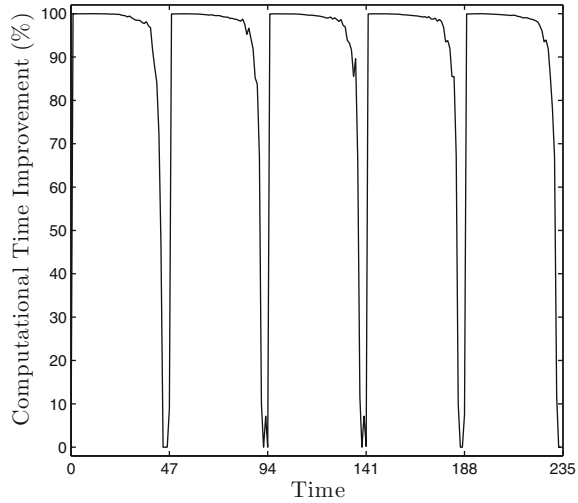
**Fig. 6.2** The closed-loop state trajectories of the reactor under the two-layer dynamic economic optimization and control framework (the two trajectories are overlapping)



**Fig. 6.3** The closed-loop input trajectories computed by two-layer dynamic economic optimization and control framework (the two trajectories are overlapping)

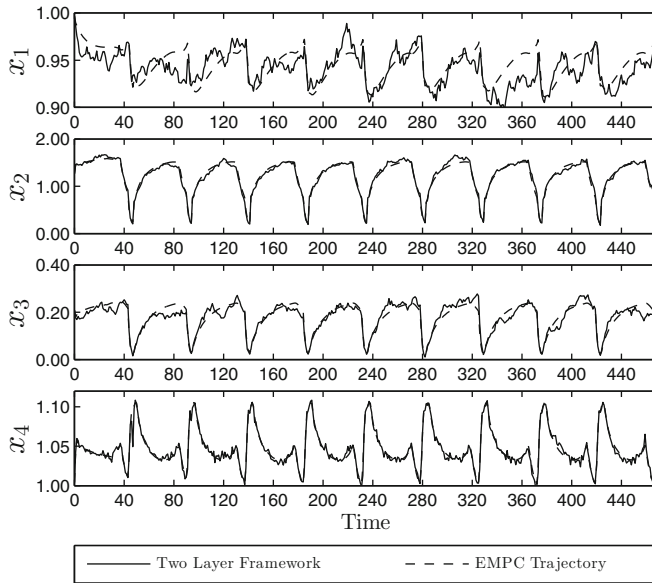
Figure 6.4 gives the computational time reduction achieved with the two-layer optimization and control framework relative to the one-layer LEMPC implementation. For this example, the lower layer LMPC computation time is insignificant compared to the computation time of the upper layer LEMPC. The two-layer framework only solves the LEMPC optimization problem once every 47 sampling times. Every

**Fig. 6.4** The computational time reduction of the two-layer optimization and control framework relative to the one-layer implementation of LEMPC

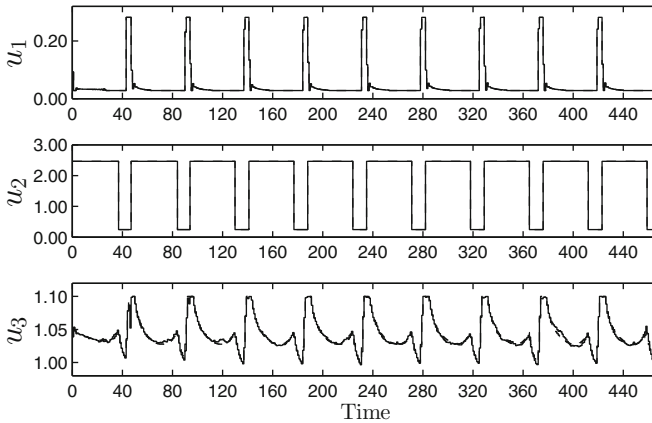


47 sampling times when the upper layer LEMPC is solved and at the end of each operating interval of length 47 when the one-layer LEMPC horizon has decreased to a comparable length as the horizon of the lower layer LEMPC, the computational burden of the two-layer framework compared to that of the one-layer LEMPC is comparable, i.e., approximately a zero percent computational time improvement is achieved (Fig. 6.4). For the other sampling times, the computation of the LMPC which computes control actions for the set of manipulated inputs  $u_2$  is much better than that compared to the one-layer LEMPC. For this case, an average of 89.4% reduction of the computational time with the two-layer framework is achieved relative to that of the one-layer LEMPC implementation.

In the second set of simulations, significant process noise is added to the system states. The noise is assumed to be bounded Gaussian white noise with zero mean and standard deviation of 0.005, 0.03, 0.01, and 0.02 and bounds given by 0.02, 0.1, 0.03, and 0.08 for the four states, respectively. To simulate the process noise, a new random number is generated and applied to the process over each sampling period. The results of a closed-loop simulation are shown in Figs. 6.5 and 6.6. Because of the added process noise, the closed-loop trajectories do not perfectly track the reference trajectories. Also, the added process noise has an effect on the closed-loop economic performance. However, this effect was minimal in the sense that the time-averaged yield of the closed-loop system under the two-layer framework is 10.3% with the added process disturbance and 10.4% without the added process disturbance. Even with the process noise, the closed-loop reactor performance is better than that at the steady-state (the yield at steady-state is 6.4%).



**Fig. 6.5** The closed-loop state trajectories of the catalytic reactor under the two-layer dynamic economic optimization and control framework and with process noise added to the states



**Fig. 6.6** The closed-loop input trajectories computed by two-layer dynamic economic optimization and control framework and with process noise added to the states (the two trajectories are nearly overlapping)

### 6.3 Unifying Dynamic Optimization with Time-Varying Economics and Control

In the previous section, a two-layer framework for dynamic optimization and control is presented. However, the framework treats the economic considerations, e.g., demand, energy pricing, variable feedstock quality, and product grade changes as time-invariant. This paradigm may be effective, especially for the applications where there is a sufficient time-scale separation between the time constants of the process/system dynamics and the update frequency of the economic parameters. However, including the time-variation of the economic considerations in the formulation of the economic stage cost may be needed to achieve good closed-loop performance when the time-scales are comparable. One class of examples where there may not be such a time-scale separation is energy systems with real-time pricing.

In this section, a two-layer framework for optimization and control of systems of the form of Eq. 4.1 is considered where the economic stage cost may be time-dependent in the sense that the system of Eq. 4.1 is equipped with a time-dependent economic cost  $l_e : [0, \infty) \times \mathbb{X} \times \mathbb{U} \rightarrow \mathbb{R} ((t, x, u) \mapsto l_e(t, x, u)$  where  $t$  is the time). The framework design is similar to that in the previous section. Specifically, the upper layer dynamic economic optimization problem (EMPC) is used to generate an economically optimal state trajectory defined over a finite-time horizon. In the lower layer, a tracking MPC is used to force the states to track the economically optimal trajectory. However, the main differences of the two-layer approach presented in this section compared to that of the previous section are the EMPC is formulated with an economic stage cost that may be explicitly time-dependent, the formulations of the layers are different, and the underlying theory and analysis are different.

Explicit constraints are used in the upper layer dynamic optimization problem to ensure that the lower layer tracking MPC may force the closed-loop state to track the trajectory computed in the optimization layer. In particular, the optimization layer is constrained to compute an optimal trajectory that is slowly time-varying. The resulting slowly time-varying trajectory vector is denoted as  $x_E(t) \in \Gamma \subset \mathbb{R}^n$  for  $t \geq 0$  where  $\Gamma$  is a compact set and the rate of change of the reference trajectory is bounded by

$$|\dot{x}_E(t)| \leq \gamma_E \quad (6.35)$$

for all  $t \geq 0$ . The deviation between the actual state trajectory and the slowly-varying reference trajectory is defined as  $e := x - x_E$  with its dynamics described by

$$\begin{aligned} \dot{e} &= f(x, u, w) - \dot{x}_E \\ &= f(e + x_E, u, w) - \dot{x}_E \\ &=: g(e, x_E, \dot{x}_E, u, w). \end{aligned} \quad (6.36)$$

The state  $e$  of the system of Eq. 6.36 will be referred to as the deviation state in the remainder of this section.

**Assumption 6.2** The system of Eq. 6.36 has a continuously differentiable, isolated equilibrium for each fixed  $x_E \in \Gamma$  in the sense that there exists a  $\hat{u} \in \mathbb{U}$  for a fixed  $x_E \in \Gamma$  to make  $e = 0$  the equilibrium of Eq. 6.36 ( $g(0, x_E, 0, \hat{u}, 0) = 0$ ).

In what follows, the upper layer EMPC computes a reference trajectory that evolves according to the nominal system dynamics  $\dot{x}_E = f(x_E, u_E, 0)$  while maintaining the state trajectory to be in the set  $\Gamma$  where  $\Gamma$  is an equilibrium manifold in the sense that  $\Gamma = \{x_E \in \mathbb{X} : \exists u_E \in \mathbb{U} \text{ s.t. } f(x_E, u_E, 0) = 0\}$ . Nevertheless, the theory applies to a more general case where the following hold:  $|\dot{x}_E| \leq \gamma_E$ ,  $\Gamma$  is compact and Assumption 6.2 is satisfied. One conceptually straightforward extension of the two-layer framework is to consider steady-state optimization in the upper layer instead of dynamic optimization. Specifically,  $x_E$  could be taken as a steady-state and varied slowly to account for the time-varying economic considerations.

### 6.3.1 Stabilizability Assumption

For each fixed  $x_E \in \Gamma$ , there exists a Lyapunov-based controller that makes the origin of the nonlinear system given by Eq. 6.36 without uncertainty ( $w \equiv 0$ ) asymptotically stable under continuous implementation. This assumption is essentially equivalent to the assumption that the nominal system of Eq. 4.1 is stabilizable at each  $x_E \in \Gamma$ . More specifically, for each fixed  $x_E \in \Gamma$ , the existence of a mapping  $h : D_s \times \Gamma \rightarrow \mathbb{U}$  and a continuously differentiable function  $V : D_s \times \Gamma \rightarrow \mathbb{R}_+$  is assumed that satisfies:

$$\alpha_1(|e|) \leq V(e, x_E) \leq \alpha_2(|e|), \quad (6.37a)$$

$$\frac{\partial V(e, x_E)}{\partial e} g(e, x_E, 0, h(e, x_E), 0) \leq -\alpha_3(|e|), \quad (6.37b)$$

$$\left| \frac{\partial V(e, x_E)}{\partial e} \right| \leq \alpha_4(|e|), \quad (6.37c)$$

$$\left| \frac{\partial V(e, x_E)}{\partial x_E} \right| \leq \alpha_5(|e|), \quad (6.37d)$$

for all  $e \in D_s$  where  $\alpha_i \in \mathcal{K}$ ,  $i = 1, 2, 3, 4, 5$ ,  $D_s$  is an open neighborhood of the origin, and  $h$  is the Lyapunov-based controller. In this sense, the function  $V$  is a Lyapunov function for each  $x_E \in \Gamma$ . While the inequalities of Eqs. 6.37a–6.37c are similar to the inequalities of standard Lyapunov functions, Eq. 6.37d is needed to account for the time-varying nature of  $x_E$ . More precisely, the special requirement that the inequalities hold uniformly in  $x_E$  is required to handle the perturbation, which results from the fact that  $x_E$  is not constant, but rather, a time-varying function.

For a fixed  $x_E \in \Gamma \subset \mathbb{R}^n$ , the symbol  $\Omega_{\rho(x_E)}$  is a level set of the Lyapunov function, i.e.,  $\Omega_{\rho(x_E)} := \{e \in \mathbb{R}^n : V(e, x_E) \leq \rho(x_E)\}$  where  $\rho(x_E) > 0$  depends on  $x_E$ .

The region  $\Omega_{\rho^*}$  is the intersection of stability regions  $\Omega_{\rho(x_E)}$  of the closed-loop system under the Lyapunov-based controller for all  $x_E \in \Gamma$ .

For broad classes of nonlinear systems arising in the context of chemical process control applications, quadratic Lyapunov functions using state deviation variables, i.e.,  $V(x) = (x - x_s)^T P(x - x_s)$ , where  $x_s$  is a steady-state, have been widely used and have been demonstrated to yield acceptable estimates of closed-loop stability regions (see [18] and the references therein). In the example of Sect. 6.3.3, a quadratic Lyapunov function is used where instead of a fixed equilibrium  $x_s$  a time-varying reference trajectory  $x_E$  is used, i.e., at time  $t$ , the Lyapunov function is given by:  $V(e(t), x_E(t)) = e^T(t) P e(t)$  where  $e(t) = x(t) - x_E(t)$ .

*Remark 6.2* If the equilibrium point  $e = 0$  of the frozen system forced by an explicit controller ( $\dot{e} = g(e, x_E, 0, h(e, x_E), 0)$ ) is exponentially stable uniformly in  $x_E$  and under some additional mild smoothness requirements, then there exists a Lyapunov function satisfying Eqs. 6.37a–6.37d [15, Lemma 9.8].

*Remark 6.3* The set  $\Omega_{\rho^*}$  is such that for any  $e \in \Omega_{\rho^*}$ , the ability to drive the state with the Lyapunov-based controller asymptotically to any fixed  $x_E \in \Gamma$  is guaranteed. This set may be estimated in the following way: first, the set  $\Gamma$  is chosen. Second, the regions  $\Omega_{\rho(x_E)}$  for a sufficiently large number of  $x_E$  in the set  $\Gamma$  are estimated. The regions  $\Omega_{\rho(x_E)}$  may be estimated as the level set (ideally the largest) of  $V$  for a fixed  $x_E \in \Gamma$  where  $\dot{V} < 0$  with the Lyapunov-based controller. Lastly, the stability region  $\Omega_{\rho^*}$  may be constructed from the intersection of these computed regions. It is important to point out that the design of the upper layer EMPC does not employ  $\Omega_{\rho^*}$ . Therefore, for practical design purposes, an explicit construction of  $\Omega_{\rho^*}$  is not needed.

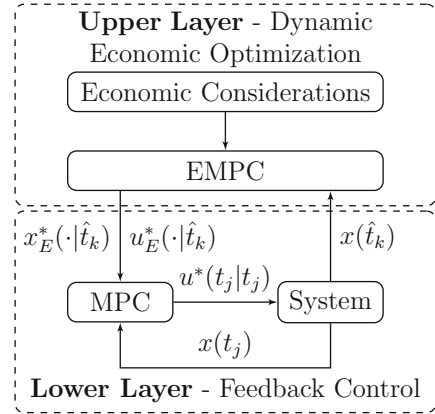
### 6.3.2 Two-Layer EMPC Scheme Addressing Time-Varying Economics

In this section, the two-layer framework for dynamic economic optimization and control for handling time-varying economics is described and the stability and robustness properties of the closed-loop system are given.

#### 6.3.2.1 Formulation and Implementation

To address time-dependent economics, a two-layer framework is presented. The two-layer framework for optimization and control may be considered an intermediate approach between existing steady-state operation and one-layer EMPC schemes. A block diagram of the two-layer control framework is given in Fig. 6.7. In this framework, optimization and control are effectively divided into separate tasks. However, the upper optimization layer is formulated with specific control-oriented constraints

**Fig. 6.7** A block diagram of the dynamic economic optimization and control framework for handling time-varying economics



to ensure stability. In the upper layer, an EMPC, formulated with a time-varying economic stage cost, computes economically optimal state and input trajectories over a finite-time horizon. The optimal trajectories are sent down to a lower layer tracking MPC to force the system to track the economically optimal state trajectory. For computational efficiency, the EMPC optimization problem is solved infrequently, i.e., it does not employ a standard receding horizon implementation strategy. Instead, the operating time is partitioned into finite-time intervals of length  $t'$  called operating periods. The operating period is chosen based on the time-scale of the process dynamics and update frequency of the economic parameters in the economic cost function, e.g., the update frequency of the energy price, product demand, or product transitions. The length of the operating period may be considered a tuning parameter of the control architecture. At the beginning of each operating period, the EMPC problem is initialized with a state measurement and is solved. The lower layer tracking MPC is solved every sampling time to maintain closed-loop stability and robustness and is formulated with a stage cost that penalizes deviations from the optimal trajectories. While in the lower layer any MPC tracking controller could be used, Lyapunov-based MPC (LMPC) is used here owing to its unique stability and robustness properties.

A summary of the implementation strategy and the notation is as follows (a detailed algorithm is given below after the formulations of the control problems are given). The operating time is divided into finite-time operating periods of length  $t' = K_E \Delta_E$  where  $K_E$  is some integer greater than or equal to one. At the beginning of the operating period denoted by  $\hat{t}_k = kt'$  where  $k = 0, 1, \dots$ , the upper layer EMPC, with hold period  $\Delta_E > 0$  (zeroth-order control parameterization is employed in the upper layer EMPC) and prediction horizon of  $T_E = N_E \Delta_E$  where  $N_E \in \mathbb{I}_+$ , receives a state measurement and computes the economically optimal state and input trajectories. The prediction horizon of the EMPC is chosen to be sufficiently large to cover the operating period plus the transition to the next operating period, i.e.,  $T_E \geq t' + T$  where  $T = N\Delta$  is the prediction horizon of the lower layer LMPC,  $\Delta > 0$  denotes the sampling period of the lower layer LMPC that is less than or



equal to  $\Delta_E$ , and  $N \in \mathbb{I}_+$  is the number of sampling periods in the prediction horizon of the LMPC. Between  $\hat{t}_k$  and  $\hat{t}_k + t'$ , the lower layer LMPC computes control actions that work to force the closed-loop state to track the optimal trajectories.

The upper layer EMPC optimization problem is as follows:

$$\min_{u_E \in S(\Delta_E)} \int_{\hat{t}_k}^{\hat{t}_k + T_E} l_e(\tau, x_E(\tau), u_E(\tau)) d\tau \quad (6.38a)$$

$$\text{s.t. } \dot{x}_E(t) = f(x_E(t), u_E(t), 0) \quad (6.38b)$$

$$x_E(\hat{t}_k) = \text{proj}_{\Gamma}(x(\hat{t}_k)) \quad (6.38c)$$

$$u_E(t) \in \mathbb{U}, \forall t \in [\hat{t}_k, \hat{t}_k + T_E) \quad (6.38d)$$

$$|\dot{x}_E(t)| \leq \gamma_E, \forall t \in [\hat{t}_k, \hat{t}_k + T_E) \quad (6.38e)$$

$$x_E(t) \in \Gamma, \forall t \in [\hat{t}_k, \hat{t}_k + T_E) \quad (6.38f)$$

where  $S(\Delta_E)$  is the family of piecewise constant functions with period  $\Delta_E$ ,  $l_e$  is the time-dependent economic measure which defines the cost function, the state  $x_E$  is the predicted trajectory of the system with the input trajectory  $u_E$  computed by the EMPC and  $x(\hat{t}_k)$  is the state measurement obtained at time  $\hat{t}_k$ . The optimal state and input trajectory computed by the EMPC are denoted as  $x_E^*(t|\hat{t}_k)$  and  $u_E^*(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_k + T_E)$ , respectively.

In the optimization problem of Eq. 6.38, the constraint of Eq. 6.38b is the nominal dynamic model of the system ( $w \equiv 0$ ) used to predict the future evolution under the sample-and-hold input trajectory. The constraint of Eq. 6.38c defines the initial condition of the optimization problem which is a projection of the state measurement at  $\hat{t}_k$  onto the set  $\Gamma$  where the symbol  $\text{proj}_{\Gamma}(x)$  denotes the projection of  $x$  onto the set  $\Gamma$ . The constraint of Eq. 6.38d ensures that the computed input trajectory takes values in the set of admissible inputs. The constraint of Eq. 6.38e limits the rate of change of the economically optimal state trajectory. Finally, the constraint of Eq. 6.38f ensures that the state evolution is maintained in the region  $\Gamma$ .

The constraint of Eq. 6.38c is used to ensure that the optimization problem is feasible. The projection operator may be any projection operator that projects the current state  $x(\hat{t}_k)$  onto a near (ideally the nearest) point in the set  $\Gamma$ . In some cases, when the sampling periods of the upper and lower layers and the bounded disturbance are sufficiently small, it may also be sufficient to use the predicted state  $x_E^*(\hat{t}_k|\hat{t}_{k-1})$  derived from the solution of the optimization problem of Eq. 6.38 that was solved at the beginning of the preceding operating period. Another potential option is to allow for the initial condition  $x_E(\hat{t}_k)$  be a decision variable in the optimization problem by including another term in the objective function penalizing the deviation of the computed initial condition from the current state  $x(\hat{t}_k)$ . In this sense, the framework offers a degree of flexibility in the selection of the projection operator.

The last two constraints of the optimization problem of Eq. 6.38 are used to guarantee closed-loop stability under the integrated framework and to ensure that the lower layer may force the system to track the optimal state trajectory, i.e., they

are control-oriented constraints. This is a departure from other types of two-layer dynamic economic optimization and control architectures featuring, for example, dynamic real-time optimization in the upper optimization layer. Also, the constraint imposed in the upper layer EMPC on the rate of change of the optimal trajectory (Eq. 6.38e) does pose a restriction on the feasible set of the optimization problem of Eq. 6.38 and could affect closed-loop economic performance achieved under the resulting two-layer framework. However, allowing the optimal state trajectory to have a large rate of change may be undesirable for many applications based on practical considerations like excessive strain on control actuators as well as the difficulty of forcing the system to track a rapidly changing reference trajectory in the presence of disturbances.

At the lower feedback control level, LMPC is employed to force the state to track the economically optimal state trajectory. The LMPC is implemented with a standard receding horizon implementation, i.e., the LMPC recomputes an updated input trajectory synchronously every sampling time. Let  $\{t_j\}_{j \geq 0}$  where  $t_j = j\Delta$ ,  $j = 0, 1, \dots$  denote the sampling time sequence of the LMPC. Also, the dynamic model used in the LMPC is that of Eq. 6.36, which is the deviation system. The LMPC optimization problem is given by:

$$\min_{u \in S(\Delta)} \int_{t_j}^{t_j+T} \left( |\tilde{e}(\tau)|_{Q_c}^2 + |u(\tau) - u_E^*(\tau|\hat{t}_k)|_{R_c}^2 \right) d\tau \quad (6.39a)$$

$$\text{s.t. } \dot{\tilde{e}}(t) = g(\tilde{e}(t), x_E^*(t|\hat{t}_k), \dot{x}_E^*(t|\hat{t}_k), u(t), 0) \quad (6.39b)$$

$$\tilde{e}(t_j) = x(t_j) - x_E^*(t_j|\hat{t}_k) \quad (6.39c)$$

$$u(t) \in \mathbb{U}, \forall t \in [t_j, t_j + T) \quad (6.39d)$$

$$\begin{aligned} & \frac{\partial V(\tilde{e}(t_j), x_E^*(t_j|\hat{t}_k))}{\partial e} g(\tilde{e}(t_j), x_E^*(t_j|\hat{t}_k), 0, u(t_j), 0) \\ & \leq \frac{\partial V(\tilde{e}(t_j), x_E^*(t_j|\hat{t}_k))}{\partial e} g(\tilde{e}(t_j), x_E^*(t_j|\hat{t}_k), 0, h(\tilde{e}(t_j), x_E(t_j|\hat{t}_k)), 0) \end{aligned} \quad (6.39e)$$

where  $S(\Delta)$  is the family of piecewise constant functions with sampling period  $\Delta$ ,  $N$  is the prediction horizon of the LMPC,  $\tilde{e}$  is the predicted deviation between the state trajectory predicted by the nominal model under the input trajectory computed by the LMPC and the economically optimal state trajectory  $x_E^*(\cdot|\hat{t}_k)$ . The optimal solution of the optimization problem of Eq. 6.39 is denoted by  $u^*(t|t_j)$  defined for  $t \in [t_j, t_{j+N})$ .

In the optimization problem of Eq. 6.39, the constraint of Eq. 6.39b is the nominal model of the deviation system. The constraint of Eq. 6.39c is the initial condition to the dynamic optimization problem. The constraint of Eq. 6.39d defines the control energy available to all manipulated inputs. The constraint of Eq. 6.39e ensures that the Lyapunov function of the closed-loop system with the LMPC decreases by at least the rate achieved by the Lyapunov-based controller. The last constraint ensures

that the closed-loop state trajectory converges to a neighborhood of the optimal state trajectory computed by the upper layer EMPC.

The implementation strategy of the dynamic economic optimization and control framework is summarized by the following algorithm.

1. At  $\hat{t}_k$ , the EMPC receives a state measurement  $x(\hat{t}_k)$  and projects the current state  $x(\hat{t}_k)$  onto the set  $\Gamma$ . Go to Step 2.
2. The EMPC computes the economically optimal state and input trajectories:  $x_E^*(t|\hat{t}_k)$  and  $u_E^*(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_k + T_E)$ . Go to Step 3.
3. For  $\hat{t}_k + t'$  (one operating period), repeat:
  - 3.1 The LMPC receives a state measurement  $x(t_j)$  and computes the deviation of the current state from the optimal state trajectory. The error  $e(t_j)$  is used to initialize the dynamic model of the LMPC. Go to Step 3.2.
  - 3.2 The LMPC optimization problem is solved to compute an optimal input trajectory  $u^*(t|t_j)$  defined for  $t \in [t_j, t_j + T)$ . Go to Step 3.3.
  - 3.3 The control action computed for the first sampling period of the prediction horizon is sent to the control actuators to be applied from  $t_j$  to  $t_{j+1}$ . If  $t_{j+1} > \hat{t}_k + t'$ , go to Step 4 and let  $j \leftarrow j + 1$ . Else, go to 3.1 and let  $j \leftarrow j + 1$ .
4. Go to Step 1,  $k \leftarrow k + 1$ .

### 6.3.2.2 Stability Analysis

In this subsection, the stability properties of the two-layer control framework with the EMPC of Eq. 6.38 in the upper layer and the LMPC of Eq. 6.39 in the lower layer when applied the system of Eq. 4.1. Before these properties may be presented, several properties are presented that are needed in the analysis. Owing to the fact that  $\Omega_{\rho^*}$ ,  $\Gamma$ ,  $\mathbb{U}$ , and  $\mathbb{W}$  are compact sets and  $f$  is locally Lipschitz, there exists  $M_x > 0$  such that

$$|f(e + x_E, u, w)| \leq M_x \quad (6.40)$$

for all  $e \in \Omega_{\rho^*}$ ,  $x_E \in \Gamma$ ,  $u \in \mathbb{U}$ , and  $w \in \mathbb{W}$ . From similar conditions and since the rate of change of  $x_E$  is bounded, there exists  $M > 0$  such that

$$|g(e, x_E, \dot{x}_E, u, w)| \leq M \quad (6.41)$$

for all  $e \in \Omega_{\rho^*}$ ,  $x_E \in \Gamma$ ,  $u \in \mathbb{U}$ ,  $w \in \mathbb{W}$  and  $|\dot{x}_E| \leq \gamma_E$ . In addition, since the Lyapunov function  $V$  is continuously differentiable (in both arguments) and the fact that  $f$  is locally Lipschitz, there exist positive constants  $L_e, L_w, L'_e, L'_E, L''_E, L'_w$  such that

$$|g(e, x_E, \dot{x}_E, u, w) - g(e', x_E, \dot{x}_E, u, 0)| \leq L_e|e - e'| + L_w|w|, \quad (6.42)$$

$$\left| \frac{\partial V(e, x_E)}{\partial e} g(e, x_E, \dot{x}_E, u, w) - \frac{\partial V(e', x'_E)}{\partial e} g(e', x'_E, \dot{x}'_E, u, 0) \right| \quad (6.43)$$

$$\leq L'_e |e - e'| + L'_E |x_E - x'_E| + L''_E |\dot{x}_E - \dot{x}'_E| + L'_w |w|$$

for all  $e, e' \in \Omega_{\rho^*}$ ,  $x_E, x'_E \in \Gamma$ ,  $u \in \mathbb{U}$ ,  $w \in \mathbb{W}$ ,  $|\dot{x}_E| \leq \gamma_E$ , and  $|\dot{x}'_E| \leq \gamma_E$ .

The following Lemma gives the feasibility properties of the EMPC and therefore, by the constraint of Eq. 6.38f, the optimal state trajectory  $x_E^*(t|\hat{t}_k)$  is embedded in the set  $\Gamma$  for  $t \in [\hat{t}_k, \hat{t}_{k+1}]$ .

**Lemma 6.1** *Consider the system of Eq. 6.38b over the prediction horizon. If Assumption 6.2 is satisfied, the optimization problem of Eq. 6.38 is feasible and therefore, the optimal state trajectory  $x_E^*(t|\hat{t}_k)$  for  $t \in [\hat{t}_k, \hat{t}_k + T_E]$  computed by applying the optimal input trajectory  $u_E^*(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_k + T_E]$  takes values in the set  $\Gamma$ .*

*Proof* When the EMPC optimization problem of Eq. 6.38 is solved with an initial condition satisfying  $x_E(\hat{t}_k) \in \Gamma$  (this is guaranteed through the projection procedure), the feasibility of the optimization problem follows if Assumption 6.2 is satisfied because maintaining the state at the initial condition along the predicted horizon is a feasible solution to the optimization problem as it satisfies all the constraints, i.e., there exists a constant input trajectory  $u_E(t) = \bar{u}_E \in \mathbb{U}$  for  $t \in [\hat{t}_k, \hat{t}_k + T_E]$  that maintains the state trajectory at its initial condition:  $x_E(t) = \text{proj}_\Gamma(x(\hat{t}_k))$  for  $t \in [\hat{t}_k, \hat{t}_k + T_E]$ . Owing to the fact that the problem is feasible and imposing the constraint of Eq. 6.38f, the optimal state trajectory  $x_E^*(t|\hat{t}_k)$  is bounded in the set  $\Gamma$  for  $t \in [\hat{t}_k, \hat{t}_k + T_E]$ .

Theorem 6.2 provides sufficient conditions such that the LMPC may track the economically optimal trajectory  $x_E^*$ . More specifically, the deviation state gets small over time until it is bounded in a small ball containing the origin.

**Theorem 6.2** *Consider the system of Eq. 4.1 in closed-loop under the tracking LMPC of Eq. 6.39 based on the Lyapunov-based controller that satisfies the conditions of Eqs. 6.37a, 6.37d with the reference trajectory  $x_E^*$  computed by the upper layer EMPC of Eq. 6.38. Let  $\varepsilon_{error} > 0$ ,  $\mu > 0$ ,  $\varepsilon_w > 0$ ,  $\Delta > 0$ ,  $\Delta_E > 0$ ,  $N \geq 1$ ,  $N_E \geq 1$ ,  $N_E \Delta_E \geq t' + N \Delta$ , and  $\gamma_E > 0$  satisfy*

$$\mu > \alpha_3^{-1} \left( \frac{(L'_x M + L'_E \gamma_E) \Delta + (L''_E + \alpha_5 (\alpha_1^{-1} (\rho^*))) \gamma_E + L'_w \theta}{\hat{\theta}} \right) \quad (6.44)$$

for some  $\hat{\theta} \in (0, 1)$ ,

$$\varepsilon_{error} = \max_{s \in [0, \Delta]} \{|e(s)| : e(0) \in B_\mu \text{ for all } x_E \in \Gamma\}, \quad (6.45)$$

and  $B_\mu \subset B_{\varepsilon_{error}} \subset \Omega_{\rho^*}$ . If  $(x(0) - x_E^*(0)) \in \Omega_{\rho^*}$ , then the deviation state of the system of Eq. 6.36 is always bounded in  $\Omega_{\rho^*}$  and therefore, also, the closed-loop

state trajectory  $x$  is always bounded. Furthermore, the deviation between the state trajectory of Eq. 4.1 and the economically optimal trajectory is ultimately bounded in  $B_{\varepsilon_{\text{error}}}$ .

*Proof* The proof consists of two parts. First, the LMPC optimization problem of Eq. 6.39 is shown to be feasible for all deviation states in  $\Omega_{\rho^*}$ . Subsequently, the deviation state is proved to be bounded in  $\Omega_{\rho^*}$  and to be ultimately bounded in  $B_{\varepsilon_{\text{error}}}$ .

*Part 1:* When the deviation state is maintained in  $\Omega_{\rho^*}$  (which will be proved in Part 2), the feasibility of the LMPC of Eq. 6.39 follows because the input trajectory obtained from the Lyapunov-based controller is a feasible solution. Specifically, define the trajectory  $v$  such that:

$$\begin{aligned}\dot{z}(t) &= g(z(t), x_E^*(t|\hat{t}_k), \dot{x}_E^*(t|\hat{t}_k), v(t), 0) \\ v(t) &= h(z(t_i), x_E^*(t_i|\hat{t}_k))\end{aligned}$$

for  $t \in [t_i, t_{i+1})$ ,  $i = j, j + 1, \dots, N - 1$  where  $z(t_j) = e(t_j)$ . The trajectory  $v$  is a feasible solution to the optimization problem of Eq. 6.39 since the trajectory satisfies the input and the Lyapunov function constraints of Eq. 6.39. This is guaranteed by the closed-loop stability property of the Lyapunov-based controller.

*Part 2:* At  $\hat{t}_k$ , the EMPC computes an optimal trajectory  $x_E^*(\cdot|\hat{t}_k)$  for the LMPC to track for one operating period. The computed trajectory is such that  $x_E^*(t|\hat{t}_k)$  and  $|\dot{x}_E^*(t|\hat{t}_k)| \leq \gamma_E$  for all  $t \in [\hat{t}_k, \hat{t}_{k+1})$  (Lemma 6.1). For simplicity of notation, let  $x_E(\tau) = x_E^*(\tau|\hat{t}_k)$ ,  $\dot{x}_E(\tau) = \dot{x}_E^*(\tau|\hat{t}_k)$ ,

$$\frac{\partial V(\tau)}{\partial e} := \frac{\partial V(e(\tau), x_E(\tau))}{\partial e}, \text{ and } \frac{\partial V(\tau)}{\partial x_E} := \frac{\partial V(e(\tau), x_E(\tau))}{\partial x_E} \quad (6.46)$$

for any  $\tau \in [t_j, t_{j+1})$ . At any sampling time  $t_j \in [\hat{t}_k, \hat{t}_k + t')$  of the LMPC, consider  $e(t_j) \in \Omega_{\rho^*}$  (recursive arguments will be applied to show this is always the case when  $e(0) \in \Omega_{\rho^*}$ ). The computed control action at  $t_j$  satisfies:

$$\begin{aligned}\frac{\partial V(t_j)}{\partial e} g(e(t_j), x_E(t_j), 0, u^*(t_j|t_j), 0) &\leq \frac{\partial V(t_j)}{\partial e} g(e(t_j), x_E(t_j), 0, h(e(t_j), x_E(t_j)), 0) \\ &\leq -\alpha_3(|e(t_j)|)\end{aligned} \quad (6.47)$$

for all  $e(t_j) \in \Omega_{\rho^*}$ . For all  $\tau \in [t_j, t_{j+1})$ , the time derivative of the Lyapunov function is given by:

$$\dot{V}(e(\tau), x_E(\tau)) = \frac{\partial V(\tau)}{\partial e} \dot{e}(\tau) + \frac{\partial V(\tau)}{\partial x_E} \dot{x}_E(\tau). \quad (6.48)$$

Adding and subtracting the left-hand term of Eq. 6.47 and from the bound of Eq. 6.47, the time-derivative of the Lyapunov function may be upper bounded as follows:

$$\begin{aligned} \dot{V}(e(\tau), x_E(\tau)) &\leq -\alpha_3(|e(t_j)|) + \frac{\partial V(\tau)}{\partial e} g(e(\tau), x_E(\tau), \dot{x}_E(\tau), u^*(t_j|t_j), w(\tau)) \\ &\quad - \frac{\partial V(t_j)}{\partial e} g(e(t_j), x_E(t_j), 0, u^*(t_j|t_j), 0) + \frac{\partial V(\tau)}{\partial x_E} \dot{x}_E(\tau) \end{aligned} \quad (6.49)$$

for all  $\tau \in [t_j, t_{j+1})$ . From Eq. 6.43, the time derivative of the Lyapunov function (Eq. 6.49) may be further upper bounded:

$$\begin{aligned} \dot{V}(e(\tau), x_E(\tau)) &\leq -\alpha_3(|e(t_j)|) + L'_x |e(\tau) - e(t_j)| + L'_E |x_E(\tau) - x_E(t_j)| \\ &\quad + L''_E |\dot{x}_E(\tau)| + L'_w |w(\tau)| + \alpha_5(|e(\tau)|) |\dot{x}_E(\tau)| \\ &\leq -\alpha_3(|e(t_j)|) + L'_x |e(\tau) - e(t_j)| + L'_E |x_E(\tau) - x_E(t_j)| \\ &\quad + (L''_E + \alpha_5(|e(\tau)|)) \gamma_E + L'_w \theta \end{aligned} \quad (6.50)$$

for all  $e(t_j) \in \Omega_{\rho^*}$  and  $\tau \in [t_j, t_{j+1})$  where the second inequality follows from the fact that  $|\dot{x}_E(\tau)| \leq \gamma_E$  and  $w(\tau) \in \mathbb{W}$ .

Taking into account Eq. 6.41 and the fact that  $|\dot{x}_E(\tau)| \leq \gamma_E$  and the continuity of  $e$  and  $x_E$ , the following bounds may be derived for all  $\tau \in [t_j, t_{j+1})$ :

$$|e(\tau) - e(t_j)| \leq M \Delta, \quad (6.51)$$

$$|x_E(\tau) - x_E(t_j)| \leq \gamma_E \Delta. \quad (6.52)$$

From Eqs. 6.50–6.52, the following inequality is obtained:

$$\begin{aligned} \dot{V}(e(\tau), x_E(\tau)) &\leq -\alpha_3(|e(t_j)|) + (L'_x M + L'_E \gamma_E) \Delta \\ &\quad + (L''_E + \alpha_5(|e(\tau)|)) \gamma_E + L'_w \theta \end{aligned} \quad (6.53)$$

for all  $\tau \in [t_j, t_{j+1})$ .

If  $\Delta$ ,  $\gamma_E$  and  $\theta$  are sufficiently small such that there exist  $\hat{\theta} \in (0, 1)$  and  $(\mu, \varepsilon_{\text{error}})$  satisfying Eqs. 6.44, 6.45 with  $B_\mu \subset B_{\varepsilon_{\text{error}}} \subset \Omega_{\rho^*}$ , the following bound on the time-derivative of the Lyapunov function follows:

$$\dot{V}(e(\tau), x_E(\tau)) \leq -(1 - \hat{\theta}) \alpha_3(|e(t_j)|) \quad (6.54)$$

for all  $\tau \in [t_j, t_{j+1})$  and  $e(t_j) \in \Omega_{\rho^*} \setminus B_\mu$ . Integrating this bound on  $t \in [t_j, t_{j+1})$ , one obtains that:

$$V(e(t_{j+1}), x_E(t_{j+1})) \leq V(e(t_j), x_E(t_j)) - (1 - \hat{\theta}) \Delta \alpha_3(|e(t_j)|) \quad (6.55)$$

$$V(e(t), x_E(t)) \leq V(e(t_j), x_E(t_j)) \quad \forall t \in [t_j, t_{j+1}) \quad (6.56)$$

for all  $e(t_j) \in \Omega_{\rho^*} \setminus B_\mu$ . Using the above inequalities recursively, it may be proved that if  $e(t_j) \in \Omega_{\rho^*} \setminus B_\mu$ , the deviation between the actual state trajectory and the economic optimal trajectory converges to  $B_\mu$  in a finite number of sampling times without going outside the set  $\Omega_{\rho^*}$ . Since the deviation state is always embedded in

the set  $\Omega_{\rho^*}$  and from Lemma 6.1, the economically optimal state trajectory is always embedded in the set  $\Gamma$ , the boundedness of the closed-loop state trajectory of Eq. 4.1 under the lower layer LMPC follows because  $\Omega_{\rho^*}$  and  $\Gamma$  are compact sets.

To summarize, if  $e(t_j) \in \Omega_{\rho^*} \setminus B_\mu$ , then

$$V(e(t_{j+1}), x_E(t_{j+1})) < V(e(t_j), x_E(t_j)). \quad (6.57)$$

Furthermore, the deviation between the state trajectory and the economic optimal trajectory is ultimately bounded in  $B_{\varepsilon_{\text{error}}}$  where satisfies Eq. 6.45 and  $B_\mu \subset B_{\varepsilon_{\text{error}}} \subset \Omega_{\rho^*}$ . This statement holds because if the deviation state comes out of the ball  $B_\mu$ , the deviation state is maintained within the ball  $B_{\varepsilon_{\text{error}}}$  owing to Eq. 6.45. Once the deviation comes out of the ball  $B_\mu$ , the Lyapunov function becomes decreasing.

#### Notes and remarks on results:

- Three factors influences the time-derivative of the Lyapunov function when  $e(t_j) \in \Omega_{\rho^*} \setminus B_\mu$  as observed in Eq. 6.53: the sampling period of the lower layer LMPC, the bound on the disturbance, and the bound on the rate of change of the economically optimal trajectory. While the bound on the disturbance is a property of the system, two of the other properties may be used to achieve a desired level of tracking: the sampling period of the lower level control loop and the rate of change of the economically optimal tracking trajectory.
- Theorem 6.2 clarifies how the parameter  $\gamma_E$  arises and why it is needed in the formulation of the EMPC of Eq. 6.38.
- No guarantee is made that the closed-loop economic performance with the two-layer framework is better compared to the performance using a steady-state model in the upper layer. In some cases, it may be the case that closed-loop performance is the same or possibly better using a steady-state model in the upper layer EMPC. In this case, the stability result presented here may be extended to the case where the optimal steady-state varies sufficiently slow.

### 6.3.3 Application to a Chemical Process Example

Consider a well-mixed, non-isothermal continuous stirred tank reactor (CSTR) where an elementary (first-order) reaction takes place of the form  $A \rightarrow B$ . The feed to the reactor consists of pure  $A$  at volumetric flow rate  $F$ , temperature  $T_0 + \Delta T_0$  and molar concentration  $C_{A0} + \Delta C_{A0}$  where  $\Delta T_0$  and  $\Delta C_{A0}$  are disturbances. A jacket around the reactor is used to provide/remove heat to the reactor. The dynamic equations describing the behavior of the system, obtained through material and energy balances under standard modeling assumptions, are given below:

**Table 6.2** Process parameters of the CSTR of Eq. 6.58

$F$	$5.0\text{m}^3\text{h}^{-1}$	$\Delta H$	$-1.2 \times 10^4 \text{ kJ kmol}^{-1}$
$V_R$	$1.0\text{m}^3$	$k_0$	$3.0 \times 10^7 \text{ h}^{-1}$
$T_0$	$300\text{K}$	$E$	$5.0 \times 10^4 \text{ kJ kmol}^{-1}$
$R$	$8.314\text{kJ kmol}^{-1} \text{ K}^{-1}$	$\rho_L$	$1000\text{kg m}^{-3}$
$C_p$	$0.231 \text{ kJ kg}^{-1} \text{ K}^{-1}$		

$$\frac{dT}{dt} = \frac{F}{V_R} (T_0 + \Delta T_0 - T) - \frac{\Delta H k_0}{\rho_L C_p} e^{\frac{-E}{RT}} C_A + \frac{Q}{\rho_L C_p V_R} \quad (6.58a)$$

$$\frac{dC_A}{dt} = \frac{F}{V_R} (C_{A0} + \Delta C_{A0} - C_A) - k_0 e^{\frac{-E}{RT}} C_A \quad (6.58b)$$

where  $C_A$  is the concentration of the reactant  $A$  in the reactor,  $T$  is the reactor temperature,  $Q$  is the rate of heat input/removal,  $V_R$  is the reactor volume,  $\Delta H$  is the heat of the reaction,  $k_0$  and  $E$  are the pre-exponential constant and activation energy of the reaction, respectively,  $C_p$  and  $\rho_L$  denote the heat capacity and the density of the fluid in the reactor, respectively. The values of the process parameters are given in Table 6.2. The state vector is  $x = [T \ C_A]^T$  and the manipulated inputs are the heat rate  $u_1 = Q$  where  $u_1 \in [-2, 2] \times 10^5 \text{ kJ h}^{-1}$  and the inlet reactant concentration  $u_2 = C_{A0}$  where  $u_2 \in [0.5, 8.0] \text{ kmol m}^{-3}$ . The feed disturbances are simulated as bounded Gaussian white noise with zero mean, variances  $20 \text{ K}^2$  and  $0.1 \text{ kmol}^2 \text{ m}^{-6}$ , and bounds given by  $|\Delta T_0| \leq 15 \text{ K}$  and  $|\Delta C_{A0}| \leq 1.0 \text{ kmol m}^{-3}$ .

The control objective is to force the system to track the economically optimal time-varying operating trajectories computed by the upper layer EMPC. The set  $\Gamma$  is defined as

$$\Gamma := \{x \in \mathbb{R}^2 : 340 \leq x_1 \leq 390 \text{ K}, 0.5 \leq x_2 \leq 3.0 \text{ kmol m}^{-3}\}. \quad (6.59)$$

In this example, the time-varying economic stage cost penalizes energy consumption, credits conversion of the reactant to the product, and penalizes the deviation of temperature from  $365.0 \text{ K}$  and is given by:

$$l_e(t, x, u) = p_1(t)u_1^2 - p_2(t)\frac{(u_2 - x_2)}{u_2} + p_3(t)(x_1 - 365.0 \text{ K})^2 \quad (6.60)$$

where  $p_1$ ,  $p_2$ , and  $p_3$  are the potentially time-varying weighting factors. The last term in the economic stage cost is used to prevent the system from operating on the boundary of  $\Gamma$  for long periods of time. The magnitudes of the economic weighting factors have been chosen so that all terms in the economic cost have the same order of magnitude. For this example,  $p_1$  and  $p_3$  are chosen to be time-varying and  $p_2 = 10$  is constant. The time-varying weight  $p_1(t)$ , over four hours of operation, is given by



$$p_1(t) = \begin{cases} 1.0 \times 10^{-7}, & t < 1.0 \text{ h} \\ 5.0 \times 10^{-8}, & 1.0 \text{ h} \leq t < 2.0 \text{ h} \\ 1.0 \times 10^{-8}, & 2.0 \text{ h} \leq t < 3.0 \text{ h} \\ 5.0 \times 10^{-8}, & 3.0 \text{ h} \leq t \leq 4.0 \text{ h} \end{cases}$$

and is used to model the time-varying energy cost. The time-varying weight  $p_3(t)$  is given by

$$p_3(t) = \begin{cases} 1.0 \times 10^{-2}, & t < 1.0 \text{ h} \\ 7.5 \times 10^{-3}, & 1.0 \text{ h} \leq t < 2.0 \text{ h} \\ 5.0 \times 10^{-3}, & 2.0 \text{ h} \leq t < 3.0 \text{ h} \\ 7.5 \times 10^{-3}, & 3.0 \text{ h} \leq t \leq 4.0 \text{ h} \end{cases}$$

The rationale for varying  $p_3$  is to allow the CSTR be operated over a larger temperature range when the energy cost decreases and thus, take advantage of the decreased energy cost.

The upper layer EMPC is implemented with a sampling period of  $\Delta_E = 36s$  and prediction horizon of  $N_E = 60$  sampling periods. It is solved every 0.50 h, i.e., the operating period is chosen to be  $t' = 0.50$  h. The prediction horizon and operating period are chosen to account for the update frequency of the economic weighting parameters. It is found that defining and imposing a rate of change constraint in the upper layer EMPC, i.e., defining the parameter  $\gamma_E$ , is not needed for this particular example because the closed-loop system under the lower layer LMPC is able to achieve acceptable tracking performance without imposing a rate of change constraint in the upper layer EMPC. The projection operator is such that it projects the current state to the closest boundary of  $\Gamma$  if the current state is outside the set  $\Gamma$ , e.g., if  $x = [400 \text{ K } 2.0 \text{ kmol m}^{-3}]^T$ , then  $\text{proj}_\Gamma(x) = [390 \text{ K } 2.0 \text{ kmol m}^{-3}]^T$ .

To design the lower layer LMPC, a Lyapunov-based controller is designed for the CSTR, which is essentially two proportional controllers that account for the input constraints. Specifically, the two proportional controllers are given by:

$$\begin{aligned} -K_1(x_1 - x_{E,1}^*) + u_{s,1}, \\ -K_2(x_2 - x_{E,2}^*) + u_{s,2} \end{aligned} \quad (6.61)$$

where  $K_1 = 8000$ ,  $K_2 = 0.01$ , and  $u_s$  is the steady-state input corresponding to the steady-state  $x_E^*$ , i.e., the input vector that makes the right-hand side of Eqs. 6.58a, 6.58b equal to zero with the state vector  $x_E^*$ . The resulting Lyapunov-based controller design for the CSTR is derived by accounting for the input constraints in the controller design of Eq. 6.61 as well as for the fact that  $u_s$  may be written as a function of  $x_E$ , i.e., the resulting Lyapunov-based controller is a mapping  $h$  that maps the pair  $(e, x_E)$  to  $h(e, x_E) \in \mathbb{U}$ . A quadratic Lyapunov function of the form  $V(e, x_E) = e^T P e$  is constructed for the closed-loop system under the Lyapunov-based controller with

$$P = \begin{bmatrix} 10 & 1 \\ 1 & 100 \end{bmatrix}. \quad (6.62)$$

The LMPC is implemented with a sampling time  $\Delta = 36s$ , prediction horizon  $N = 5$ , and weighting matrices of  $Q_c = P$  and  $R_c = \text{diag} [10^{-7} \ 10]$ . The prediction horizon and weighting matrices of the lower layer LMPC are tuned to achieve a close tracking of the optimal state trajectory.

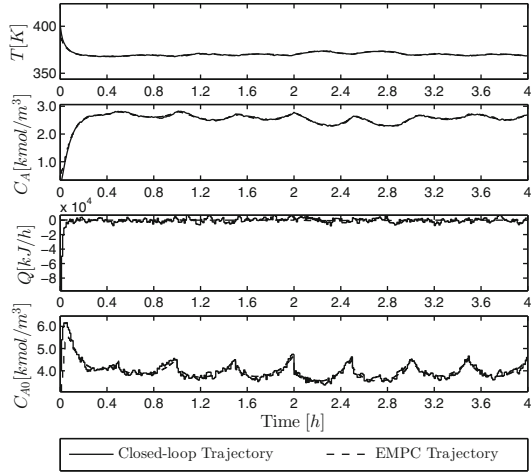
With the nonlinear system of Eqs. 6.58a–6.58b, the Lyapunov-based controller, and the Lyapunov function, the stability regions of the closed-system under the Lyapunov-based controller may be estimated for a sufficiently large number of points in  $\Gamma$ . This procedure is carried out as follows: fix  $x_E \in \Gamma$  and compute a level set of the Lyapunov function where  $\dot{V} < 0$  for all points contained in the level set. The intersection of all these level sets is taken to be an estimate of the closed-loop stability region  $\Omega_{\rho^*}$  of the CSTR under the Lyapunov-based controller. In this example,  $\Omega_{\rho^*}$  is estimated to be  $\rho^* = 110$ . Through the Lyapunov-based constraint on the LMPC of (Eq. 6.39e), the closed-loop system with the two-layer framework inherits the stability region  $\Omega_{\rho^*}$ .

To simulate the closed-loop system, explicit Euler method with integration step  $0.36s$  is used to integrate the ODEs and the open source interior point solver Ipopt [17] is used to solve the optimization problems. Three sets of closed-loop simulations are completed. In the first set of closed-loop simulations, the stability properties of the closed-loop system under the two-layer dynamic economic optimization and control framework are demonstrated. Second, time-varying operation with the two-layer dynamic economic optimization and control framework is analyzed. Third, the closed-loop economic performance of the CSTR under the two-layer framework is compared to the CSTR under a conventional approach to optimization and control.

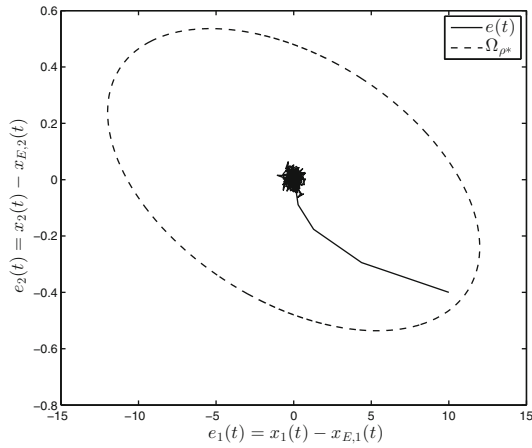
To demonstrate the closed-loop stability properties of the proposed two-layer framework, the CSTR is initialized at  $x_0 = [400 \text{ K}, 0.1 \text{ kmol m}^{-3}]$  which is outside of  $\Gamma$ , but inside the stability region  $\Omega_{\rho^*}$ . The projection operator of the upper layer EMPC projects the initial state onto the state  $x_{E,0} = [390 \text{ K}, 0.5 \text{ kmol m}^{-3}] \in \Gamma$  to use as an initial condition to the upper layer EMPC problem of Eq. 6.38. The evolution of the closed-loop system under the two-layer framework and with the inlet temperature and reactant concentration disturbance is shown in Figs. 6.8 and 6.9. From Fig. 6.9, the deviation of the actual closed-loop state and the economically optimal state is always maintained inside  $\Omega_{\rho^*}$ . Moreover, the deviation becomes small over time until it is ultimately bounded in a small ball.

Two simulations of the closed-loop system without the feed disturbances added are shown in Figs. 6.10 and 6.11 with two different initial conditions to analyze the time-varying operation with the two-layer dynamic economic optimization and process control framework. The initial state in Fig. 6.10 is  $x_0 = [400 \text{ K}, 3.0 \text{ kmol m}^{-3}]^T$ , while the initial state in Fig. 6.11 is  $x_0 = [320 \text{ K}, 3.0 \text{ kmol m}^{-3}]^T$ . The closed-loop evolution of the two cases is initially different. For the CSTR starting at the larger temperature, heat should be removed from the reactor and the minimum amount of reactant material should be supplied to the reactor to decrease the temperature of the reactor. In contrast, when the CSTR is initialized at the smaller temperature, heat should be supplied to the reactor and reactant material should be fed to the reactor to increase the reactor temperature. After a sufficiently long length of operation, the effect of the initial condition diminishes and the closed-loop evolution of the two

**Fig. 6.8** The closed-loop state and input trajectories of Eqs. 6.58a, 6.58b under the two-layer optimization and control framework with the feed disturbances and starting from 400 K and  $0.1 \text{ kmol m}^{-3}$



**Fig. 6.9** The closed-loop state trajectory of Eqs. 6.58a, 6.58b under the two-layer optimization and control framework with the feed disturbances and starting from 400 K and  $0.1 \text{ kmol m}^{-3}$  shown in deviation state-space



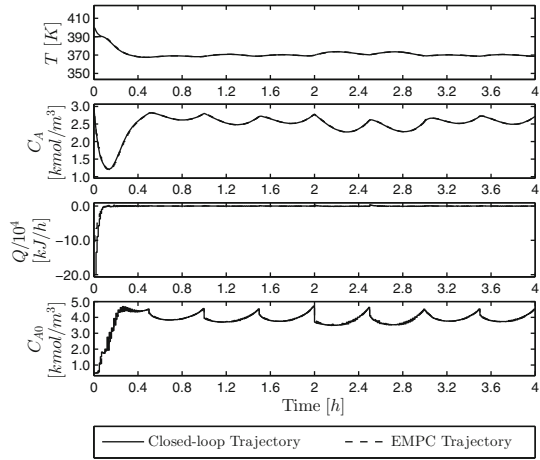
cases becomes similar. For both of these cases, the reactor is operated in a time-varying fashion, i.e., never converges to a steady-state.

To compare the closed-loop economic performance under the dynamic economic optimization and control framework and under a conventional approach to optimization and control, the total economic cost over the length of operation is defined as

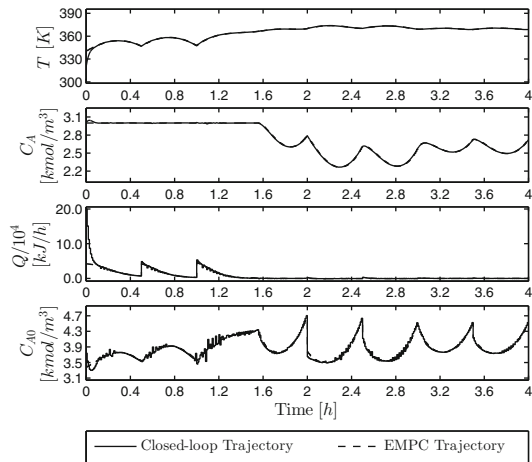
$$J_E = \sum_{j=0}^{M-1} \left( p_1(t_j) Q^2(t_j) + p_2 \frac{C_A(t_j)}{C_{A0}(t_j)} + p_3(t_j) (T(t_j) - 365)^2 \right) \quad (6.63)$$

where  $t_0$  is the initial time of the simulation and  $t_M = 4.0 \text{ h}$  is the end of the simulation. The conventional approach to optimization and control uses a steady-state economic

**Fig. 6.10** The closed-loop system states and inputs of Eqs. 6.58a, 6.58b without the feed disturbances and starting from 400 K and  $3.0 \text{ kmol m}^{-3}$



**Fig. 6.11** The closed-loop system states and inputs of Eqs. 6.58a, 6.58b without the feed disturbances and starting from 320 K and  $3.0 \text{ kmol m}^{-3}$



optimization problem to compute the optimal steady-states with respect to the time-varying economic cost weights. The optimal steady-states are used in a tracking MPC, which in this case is an LMPC, to force the CSTR states to the optimal steady-states. The optimal (time-varying) steady-state from steady-state economic optimization is

$$x_s^*(t) = \begin{cases} [370.0 \text{ K}, 2.576 \text{ kmol m}^{-3}]^T, & t < 1.0 \text{ h} \\ [371.7 \text{ K}, 2.447 \text{ kmol m}^{-3}]^T, & 1.0 \text{ h} \leq t < 2.0 \text{ h} \\ [375.2 \text{ K}, 2.205 \text{ kmol m}^{-3}]^T, & 2.0 \text{ h} \leq t < 3.0 \text{ h} \\ [371.7 \text{ K}, 2.447 \text{ kmol m}^{-3}]^T, & 3.0 \text{ h} \leq t \leq 4.0 \text{ h} \end{cases}$$

with the corresponding steady-state input of

$$u_s^*(t) = \begin{cases} [0.0 \text{ kJ h}^{-1}, 3.923 \text{ kmol m}^{-3}]^T, & t < 1.0 \text{ h} \\ [-0.5 \text{ kJ h}^{-1}, 3.827 \text{ kmol m}^{-3}]^T, & 1.0 \text{ h} \leq t < 2.0 \text{ h} \\ [0.0 \text{ kJ h}^{-1}, 3.653 \text{ kmol m}^{-3}]^T, & 2.0 \text{ h} \leq t < 3.0 \text{ h} \\ [-0.5 \text{ kJ h}^{-1}, 3.827 \text{ kmol m}^{-3}]^T, & 3.0 \text{ h} \leq t \leq 4.0 \text{ h} \end{cases}$$

An LMPC is implemented to drive the system to the time-varying optimal steady-state, which is formulated as follows:

$$\begin{aligned} \min_{u \in S(\Delta)} \quad & \int_{t_j}^{t_{j+N}} \left( |\tilde{x}(\tau) - x_s^*(\tau)|_{Q_c} + |u(\tau) - u_s^*(\tau)|_{R_c} \right) d\tau \\ \text{s.t.} \quad & \dot{\tilde{x}}(t) = f(\tilde{x}(t), u(t), 0), \\ & \tilde{x}(t_j) = x(t_j), \\ & -2 \times 10^5 \leq u_1(t) \leq 2 \times 10^5, \quad \forall t \in [t_j, t_{j+N}), \\ & 0.5 \leq u_2(t) \leq 8, \quad \forall t \in [t_j, t_{j+N}), \\ & \frac{\partial V(x(t_j))}{\partial x} f(x(t_j), u(t_j), 0) \\ & \leq \frac{\partial V(x(t_j))}{\partial x} f(x(t_j), h(x(t_j), x_s^*(t_j)), 0) \end{aligned} \quad (6.64)$$

where the Lyapunov function, the Lyapunov-based controller, the weighting matrices  $R_c$  and  $Q_c$ , the sampling period  $\Delta$ , and the prediction horizon  $N$  are all the same as the ones used in the tracking LMPC scheme.

To make a fair comparison, the same realization of the feed disturbances was applied to each closed-loop system simulation pair. The total economic cost values of several closed-loop simulations starting from different initial conditions and with and without the feed disturbances are given in Table 6.3. From the results of Table 6.3, substantial closed-loop economic performance is achieved under the two-layer framework than under the optimal steady-state tracking LMPC of Eq. 6.64. The largest economic cost improvement occurs when the CSTR is initialized at higher temperature. When the CSTR starts from a lower temperature, the amount of heat that needs to be supplied to the reactor initially is less than the amount of heat that needs to be initially removed when the CSTR starts at a higher temperature as explained above and demonstrated in Figs. 6.10 and 6.11. Thus, when the CSTR starts from a higher temperature, better closed-loop performance is achieved because less energy is required to be supplied/removed from the reactor.

**Table 6.3** Comparison of the total economic cost, given by Eq. 6.63, of the closed-loop system with and without the feed disturbances for four hours of operation

Initial conditions		Total economic cost					
$T(0)$ K	$C_A(0)$ kmol m <sup>-3</sup>	Steady-state optimization without disturbance	Two-layer Framework without Disturbance	Cost decrease (%)	Steady-state optimization with disturbance	Two-layer framework with disturbance	Cost decrease (%)
400.0	3.0	21970.5	14531.1	51.2	21642.4	14130.7	53.2
380.0	3.0	5235.4	3409.5	53.6	5060.1	3037.9	66.6
360.0	3.0	4261.8	3308.6	28.8	4083.2	2997.1	36.2
340.0	3.0	13732.2	10997.3	24.9	13554.9	10882.3	24.6
320.0	3.0	23719.4	19315.9	22.8	23729.1	19210.3	23.5
400.0	2.5	18546.8	10062.1	84.3	18283.4	9691.4	88.7
380.0	2.5	4558.7	3163.3	44.1	4387.2	2811.9	56.0
360.0	2.5	4496.4	3335.6	34.8	4322.7	3030.3	42.6
340.0	2.5	14078.3	11034.4	27.6	13910.2	10928.8	27.3
320.0	2.5	24052.2	19293.4	24.7	24002.2	19193.8	25.1
400.0	2.0	14831.5	6774.0	118.9	14682.4	6412.6	129.0
380.0	2.0	4073.2	3085.1	32.0	3905.0	2739.8	42.5
360.0	2.0	4765.4	3431.2	38.9	4596.4	3139.3	46.4
340.0	2.0	14395.5	11162.3	29.0	14236.8	11068.2	28.6
320.0	2.0	24202.7	19241.2	25.8	24223.5	19146.7	26.5
400.0	0.1	8146.1	4360.5	86.8	7999.4	4025.7	98.7

## 6.4 Addressing Closed-Loop Performance

An important theoretical consideration is the closed-loop performance of systems under EMPC because EMPC is formulated with a finite prediction horizon. The achievable closed-loop economic performance may strongly depend on the prediction horizon length. To address guaranteed closed-loop economic performance while formulating a computationally efficient control structure, a two-layer EMPC structure is presented in this section. In contrast to the two-layer EMPC methodologies presented in the previous sections, EMPC schemes are used in both layers of the two-layer EMPC structure to ensure economic performance improvement over a tracking controller, e.g., tracking MPC.

Each layer is formulated as an LEMPC scheme. The core idea of the two-layer LEMPC implementation is to solve the upper layer LEMPC infrequently (not every sampling period) while employing a long prediction horizon. Then, the solution generated by the upper layer LEMPC is subsequently used in the formulation of a lower

layer LEMPC. The lower layer LEMPC is formulated with a shorter prediction horizon and smaller sampling time than the upper layer LEMPC and computes control actions that are applied to the closed-loop system. The control actions of the lower layer LEMPC are constrained to maintain the state near the economically optimal trajectories computed in the upper layer. For guaranteed performance improvement with the two-layer LEMPC implementation scheme, both layers are formulated with explicit performance-based constraints computed by taking advantage of the availability of an auxiliary stabilizing controller. The performance-based constraints, i.e., terminal constraints, are similar to that presented in Sect. 4.4, and guarantee that both the finite-time and infinite-time closed-loop economic performance under the two-layer LEMPC scheme are at least as good as that under the stabilizing controller. The use of the two-layer control implementation allows for the control architecture to be computationally efficient. The two-layer LEMPC structure is applied to a chemical process example to demonstrate the closed-loop performance, stability, and robustness properties of the two-layer LEMPC structure.

### 6.4.1 Class of Systems

In this section, nominally operated systems are considered, i.e., the system of Eq. 4.1 with  $w \equiv 0$ . Specifically, the class of continuous-time nonlinear systems considered is described by the following state-space form:

$$\dot{x} = f(x, u) \quad (6.65)$$

where the state vector is  $x \in \mathbb{X} \subseteq \mathbb{R}^n$  and the input vector is  $u \in \mathbb{U} \subset \mathbb{R}^m$ . The vector function  $f : \mathbb{X} \times \mathbb{U} \rightarrow \mathbb{X}$  is a locally Lipschitz vector function on  $\mathbb{X} \times \mathbb{U}$ . The set of admissible inputs  $\mathbb{U}$  is assumed to be a compact set, and the state is synchronously sampled at time instances  $j\Delta$  with  $j = 0, 1, 2, \dots$  where  $\Delta > 0$  is the sampling period. As before, the initial time is taken to be zero, and the notation  $t$  will be used for the continuous-time, while the time sequence  $\{t_j\}_{j \geq 0}$  is the discrete sampling time sequence which is a synchronous partitioning of  $\mathbb{R}_+$  with  $t_j = j\Delta$ .

A time-invariant economic measure  $l_e : \mathbb{X} \times \mathbb{U} \rightarrow \mathbb{R}$  is assumed for the system of Eq. 6.3 that describes the real-time system economics. The economic measure is assumed to be continuous on  $\mathbb{X} \times \mathbb{U}$ . The optimal steady-state  $x_s^*$  and steady-state input  $u_s^*$  pair with respect to the economic cost function is computed as follows:

$$(x_s^*, u_s^*) = \arg \min_{x_s \in \mathbb{X}, u_s \in \mathbb{U}} \{l_e(x_s, u_s) : f(x_s, u_s) = 0\} .$$

The existence of a minimizing pair where the minimum is attained. For the sake of simplicity, the optimal steady-state pair is assumed to be unique and to be  $(x_s^*, u_s^*) = (0, 0)$ .

### 6.4.2 Stabilizability Assumption

A stronger stabilizability-like assumption than the assumption imposed in previous sections and chapters is needed here (stated in Assumption 6.3). In this section, the existence of a stabilizing controller that renders the origin of the closed-loop system exponentially stable under continuous implementation is assumed whereas, previously, the existence of a stabilizing controller is assumed that renders the closed-loop system only asymptotically stable under continuous implementation. The stronger assumption is needed to ensure that the stabilizing controller renders the origin of the closed-loop system exponentially (and therefore, asymptotically) stable under sample-and-hold implementation. This will be required to consider infinite-time closed-loop economic performance. Specifically, asymptotic convergence to the origin and not just convergence to a neighborhood of the steady-state (practical stability of the origin) will be required.

**Assumption 6.3** There exists a locally Lipschitz feedback controller  $h : \mathbb{X} \rightarrow \mathbb{U}$  with  $h(0) = 0$  for the system of Eq. 6.65 that renders the origin of the closed-loop system under continuous implementation of the controller locally exponentially stable. More specifically, there exist constants  $\rho > 0$ ,  $c_i > 0$ ,  $i = 1, 2, 3, 4$  and a continuously differentiable Lyapunov function  $V : \mathbb{X} \rightarrow \mathbb{R}_+$  such that the following inequalities hold:

$$c_1 |x|^2 \leq V(x) \leq c_2 |x|^2, \quad (6.66a)$$

$$\frac{\partial V(x)}{\partial x} f(x, h(x)) \leq -c_3 |x|^2, \quad (6.66b)$$

$$\left| \frac{\partial V(x)}{\partial x} \right| \leq c_4 |x|, \quad (6.66c)$$

for all  $x \in \Omega_\rho \subseteq \mathbb{X}$ .

Explicit feedback controllers that may be designed to satisfy Assumption 6.3 are, for example, feedback linearizing controller and some Lyapunov-based controllers, e.g., [15, 19]. The origin of the closed-loop system of Eq. 6.65 under the feedback controller,  $h(x)$ , implemented in a zeroth-order sample-and-hold fashion with a sufficiently small sampling period  $\Delta > 0$ , i.e., the controller is applied as an emulation controller may be shown to be exponentially stable (Corollary 2.2). Moreover, the proof of Corollary 2.2 shows that  $V$  is a Lyapunov function for the closed-loop sampled-data system in the sense that there exists a constant  $\hat{c}_3 > 0$  such that

$$\frac{\partial V(x(t))}{\partial x} f(x(t), h(x(t_j))) \leq -\hat{c}_3 |x(t)|^2 \quad (6.67)$$

for all  $t \in [t_j, t_{j+1})$  and integers  $j \geq 0$  where  $x(t)$  is the solution of Eq. 6.65 at time  $t$  starting from  $x(t_j) \in \Omega_\rho$  and with the input  $u(t) = h(x(t_j))$  for  $t \in [t_j, t_{j+1})$ . The stability region of the closed-loop system under the controller is defined as  $\Omega_\rho \subseteq X$ .



### 6.4.3 Two-Layer EMPC Structure

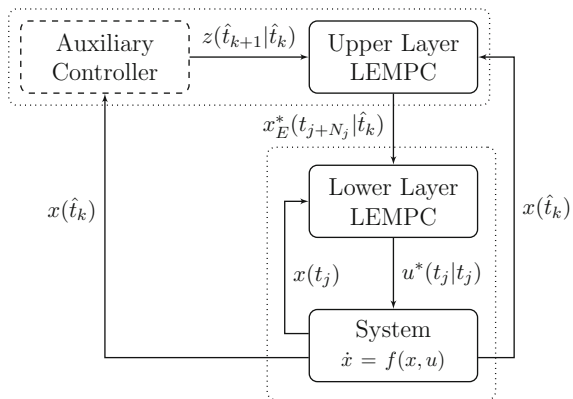
A detailed description of the two-layer LEMPC structure is provided which includes descriptions of the implementation strategy, the formulations of the upper and lower layer LEMPC schemes, and the provable stability and performance properties.

#### 6.4.3.1 Implementation Strategy

The objective of the two-layer LEMPC design is to ensure that both the finite-time and infinite-time closed-loop economic performance of the resulting closed-loop system will be at least as good the closed-loop performance under a stabilizing controller. To address this objective, performance-based constraints are employed in the formulation of the upper and lower layer LEMPC that have been computed from the stabilizing controller. The stabilizing controller may be any controller that satisfies Assumption 6.3. For example, the stabilizing controller may be an explicit controller that satisfies Assumption 6.3 or an LMPC scheme, which is equipped with a contractive Lyapunov constraint designed using an explicit controller that satisfies Assumption 6.3. The formulation of such an LMPC scheme is provided below. However, it is important to point out that the amount of computation required to solve the LMPC is generally greater than that required for an explicit controller. The stabilizing controller will be referred as the auxiliary controller for the remainder.

A block diagram of the two-layer LEMPC is given in Fig. 6.12. In the upper layer, an LEMPC is used to optimize dynamic operation over a long horizon while accounting for the performance-based constraints generated from the auxiliary controller. Both the auxiliary controller and the upper layer LEMPC compute their input trajectories at the beginning of some operating window, and thus, the auxiliary controller and upper layer LEMPC are computed once each operating window for computational efficiency. In the lower layer, an LEMPC, using a shorter prediction horizon

**Fig. 6.12** Block diagram of the two-layer EMPC structure addressing closed-loop performance and computational efficiency



and a smaller sampling period than the upper layer LEMPC, computes control inputs that are applied to the process. Terminal constraints that have been generated from the upper layer LEMPC optimal solution are used to ensure that the lower layer LEMPC guides the system along the optimal solution computed in the upper layer since it uses a shorter prediction horizon and a smaller sampling period. In this manner, the lower layer LEMPC is used to ensure robustness of the closed-loop system (recomputes its optimal trajectory every sampling period to incorporate feedback). The lower layer LEMPC may also provide additional economic cost improvement over the upper layer LEMPC solution owing to the use of a smaller sampling time.

To maintain consistency of the notation, the operating window is denoted as  $t'$  and is equal to  $N_E \Delta_E$  where  $N_E \in \mathbb{I}_+$  is the number of hold periods in the prediction horizon of the upper layer LEMPC and  $\Delta_E > 0$  is the hold period of the piecewise constant input trajectory computed in the upper layer (here,  $K_E = N_E$ ). The time sequence  $\{\hat{t}_k\}_{k \geq 0}$  denotes the discrete time steps that the upper layer computes a solution to its control problem where  $\hat{t}_k = kt'$  and  $k = 0, 1, \dots$

At the beginning of each operating window, the upper layer control problems are solved in a sequential manner: first, the auxiliary controller is solved to obtain its corresponding open-loop predicted state and input trajectories over the operating window and second, the upper layer LEMPC is solved to obtain its corresponding open-loop predicted state and input trajectories over the operation window. Specifically, the auxiliary controller computes the open-loop input trajectory that it would apply to the system over the time  $\hat{t}_k$  to  $\hat{t}_{k+1} = (k+1)t'$  along with the open-loop state trajectory. If the auxiliary controller is an explicit controller, then the open-loop state trajectory is computed by recursively solving:

$$\dot{z}(t) = f(z(t), h(z(\tau_i))) \quad (6.68)$$

for  $t \in [\tau_i, \tau_{i+1})$ ,  $i = 0, 1, \dots, N_E - 1$  where  $\tau_i := \hat{t}_k + i \Delta_E$ ,  $z(\hat{t}_k) = x(\hat{t}_k)$  is the initial condition, and  $x(\hat{t}_k)$  is a state measurement obtained at  $\hat{t}_k$ . If, instead, the auxiliary controller is an LMPC, then the open-loop state trajectory may be obtained directly from the solution of the optimization problem. The open-loop state and input trajectories under the auxiliary controller are denoted as  $z(t|\hat{t}_k)$  and  $v(t|\hat{t}_k)$  for  $t \in [\hat{t}_k, \hat{t}_{k+1}) = [kt', kt' + N_E \Delta_E)$ , respectively. The terminal state of the open-loop state trajectory,  $z(\hat{t}_{k+1}|\hat{t}_k)$ , is then sent to the upper level LEMPC.

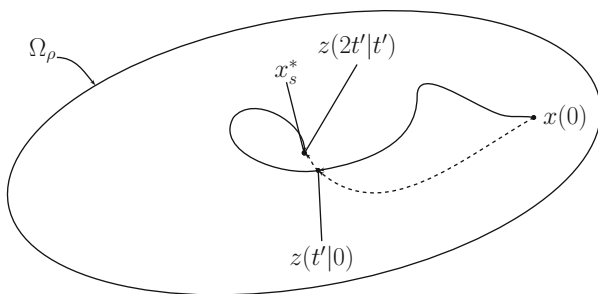
The upper layer LEMPC subsequently uses  $z(\hat{t}_{k+1}|\hat{t}_k)$  as a terminal equality constraint in the optimization problem. In this framework, no restrictions are placed on the type of operation achieved under the two-layer framework, i.e., it could be steady-state operation or some more general time-varying operating behavior. Therefore, the upper level LEMPC is an LEMPC (Sect. 4.2) equipped with mode 1 operation only. If steady-state operation is desirable, one could formulate the upper level LEMPC with a mode 2 constraint similar to that of Eq. 6.8f to ensure that the optimal steady-state is asymptotically stable under the two-layer LEMPC. However, the mode 2 constraint of LEMPC is not discussed further. After receiving  $z(\hat{t}_{k+1}|\hat{t}_k)$  from the auxiliary controller and a state measurement at  $\hat{t}_k$ , the upper layer LEMPC is solved

to compute its optimal state and input trajectories over the operating window, which are denoted as  $x_E^*(t|\hat{t}_k)$  and  $u_E^*(t|\hat{t}_k)$  for  $t \in [\hat{t}_k, \hat{t}_{k+1})$ , respectively.

The upper layer hold period is divided into  $\bar{N}$  subintervals of length  $\Delta$  ( $\Delta = \Delta_E/\bar{N}$  where  $\bar{N}$  is a positive integer). The subintervals define the sampling period of the lower layer LEMPC and correspond to the sampling time sequence  $\{t_j\}_{j \geq 0}$ . The lower layer LEMPC recomputes its optimal input trajectory employing a shrinking horizon. Namely, at the beginning of each hold period of the upper layer, the lower layer is initialized with a prediction horizon  $N_j = \bar{N}$ . The lower layer LEMPC receives a state measurement, denoted as  $x(t_j)$ , as well as  $x_E^*(t_{j+N_j}|\hat{t}_k)$  from the upper layer LEMPC. Using  $x_E^*(t_{j+N_j}|\hat{t}_k)$  as a terminal equality constraint in the lower layer LEMPC, the lower layer LEMPC is solved. The optimal input trajectory computed by the lower layer LEMPC is denoted as  $u^*(t|t_j)$ ,  $t \in [t_j, t_{j+N_j})$ . At the subsequent sampling period of the lower layer LEMPC, the prediction horizon decreases by one ( $N_{j+1} = N_j - 1$ ). If decreasing the horizon results in the horizon being set to zero, the prediction horizon is reset to  $\bar{N} = \Delta_E/\Delta$ . This happens at the beginning of the next hold period of the upper layer LEMPC.

The implementation strategy is summarized below and an illustration of the closed-loop system is given in Fig. 6.13. The lower layer LMPC is initialized with a prediction horizon of  $N_0 = \bar{N} = \Delta_E/\Delta$ . To initialize the algorithm, let  $k = 0$  and  $j = 0$ .

1. **Upper layer:** At  $\hat{t}_k$ , the auxiliary controller and the upper layer LEMPC are initialized with the state measurement  $x(\hat{t}_k)$ . Go to Step 1.1.
  - 1.1 The auxiliary controller computes its optimal input trajectory denoted as  $v(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_{k+1})$  and corresponding state trajectory denoted as  $z(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_{k+1})$ . The terminal state  $z(\hat{t}_{k+1}|\hat{t}_k)$  is sent to the upper layer LEMPC. Go to Step 1.2.



**Fig. 6.13** A state-space illustration of the evolution of the closed-loop system (*solid line*) in the stability region  $\Omega_\rho$  over two operating periods. The open-loop predicted state trajectory under the auxiliary controller is also given (*dashed line*). At the beginning of each operating window, the closed-loop state converges to the open-loop state under the auxiliary controller

- 1.2 The upper layer LEMPC receives  $z(\hat{t}_{k+1}|\hat{t}_k)$  from the auxiliary controller and computes its optimal input trajectory  $u_E^*(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_{k+1})$  and state trajectory  $x_E^*(t|\hat{t}_k)$  defined for  $t \in [\hat{t}_k, \hat{t}_{k+1})$ . Go to Step 2.
2. **Lower layer:** At  $t_j$ , the lower layer LEMPC receives a state measurement  $x(t_j)$  and the terminal state  $x_E^*(t_{j+N_j}|\hat{t}_k)$  from the upper layer LEMPC. Go to Step 2.1.
  - 2.1 The lower layer LEMPC computes its optimal input trajectory  $u^*(t|t_j)$  defined for  $t \in [t_j, t_{j+N_j})$ . Go to Step 2.2.
  - 2.2 The control action  $u^*(t_j|t_j)$ , which is the computed input for the first sampling period of the lower layer LEMPC prediction horizon, is applied to the system from  $t_j$  to  $\hat{t}_{j+1}$ . If  $N_j - 1 = 0$ , reset  $N_{j+1} = \bar{N}$ ; else, let  $N_{j+1} = N_j - 1$ . If  $t_{j+1} = \hat{t}_{k+1}$ , set  $j \leftarrow j + 1$  and  $k \leftarrow k + 1$  and go to Step 1. Else, set  $j \leftarrow j + 1$  and go to Step 2.

*Remark 6.4* Even though the lower layer LEMPC uses a shrinking horizon and nominal operation is considered, recomputing the lower layer LEMPC input at every subsequent sampling time is necessary regardless if the solution to the lower level LEMPC is the same or not. The incorporation of feedback allows for stabilization of open-loop unstable systems that cannot be accomplished with an open-loop implementation and ensures the robustness of the control solution with respect to infinitesimally small disturbances/uncertainty. For further explanation on this point, see, for example, [20].

### 6.4.3.2 Formulation

The formulations of the two LEMPC schemes are given. For convenience, the specific formulation of the LMPC needed if the auxiliary controller is chosen to be an LMPC scheme is given first. Specifically, the LMPC is given by the following optimization problem:

$$\min_{v \in S(\Delta_E)} \int_{\hat{t}_k}^{\hat{t}_{k+1}} (|z(t)|_{Q_c} + |v(t)|_{R_c}) dt \quad (6.69a)$$

$$\text{s.t. } \dot{z}(t) = f(z(t), v(t)) \quad (6.69b)$$

$$z(\hat{t}_k) = x(\hat{t}_k) \quad (6.69c)$$

$$v(t) \in \mathbb{U}, \forall t \in [\hat{t}_k, \hat{t}_{k+1}) \quad (6.69d)$$

$$\frac{\partial V(z(\tau_i))}{\partial z} f(z(\tau_i), v(\tau_i)) \leq \frac{\partial V(z(\tau_i))}{\partial z} f(z(\tau_i), h(x(\tau_i)))$$

$$\text{for } i = 0, 1, \dots, N_E - 1 \quad (6.69e)$$

where  $\tau_i := \hat{t}_k + i \Delta_E$  and  $z$  is the state trajectory of the system with input trajectory  $v$  calculated by the LMPC. The Lyapunov-based constraint of Eq. 6.69e differs from the Lyapunov-based constraint of Eq. 2.51e as it is imposed at each sampling period

along the prediction horizon of the LMPC to ensure that the state trajectory with input computed by the LMPC converges to the steady-state. Through enforcement of the Lyapunov-based constraint, the LMPC inherits the same stability properties as that of the explicit controller. The optimal solution of the optimization problem of Eq. 6.69 is denoted as  $v^*(t|\hat{t}_k)$  and is defined for  $t \in [\hat{t}_k, \hat{t}_{k+1})$ . From the optimal input trajectory, the optimal state trajectory  $z^*(t|\hat{t}_k)$ ,  $t \in [\hat{t}_k, \hat{t}_{k+1})$  may be computed for the operating window. When the LMPC is used as the auxiliary controller, the terminal state  $z^*(\hat{t}_{k+1}|\hat{t}_k)$  is sent to the upper layer LEMPC.

The formulation of the upper layer LEMPC is similar to the mode 1 LEMPC formulation with a terminal equality constraint computed from the auxiliary controller:

$$\min_{u_E \in S(\Delta_E)} \int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(x_E(t), u_E(t)) dt \quad (6.70a)$$

$$\text{s.t. } \dot{x}_E(t) = f(x_E(t), u_E(t)) \quad (6.70b)$$

$$x_E(\hat{t}_k) = x(\hat{t}_k) \quad (6.70c)$$

$$u_E(t) \in \mathbb{U}, \forall t \in [\hat{t}_k, \hat{t}_{k+1}) \quad (6.70d)$$

$$x_E(t) \in \Omega_\rho, \forall t \in [\hat{t}_k, \hat{t}_{k+1}) \quad (6.70e)$$

$$x_E(\hat{t}_{k+1}) = z(\hat{t}_{k+1}|\hat{t}_k) \quad (6.70f)$$

where  $x_E$  is the predicted state trajectory with the input trajectory  $u_E$  computed by the upper layer LEMPC. To ensure the existence of an input trajectory that has at least as good economic performance as the auxiliary LMPC input trajectory over the entire length of operation, the terminal constraint of Eq. 6.70f based on the auxiliary controller is used. The terminal constraint differs from traditional terminal equality constraints because  $z(\hat{t}_{k+1}|\hat{t}_k)$  is not necessarily the steady-state. It does, however, asymptotically converge to the economically optimal steady-state. The optimal solution to the optimization problem of the upper layer LEMPC is denoted as  $u_E^*(t|\hat{t}_k)$  and is defined for  $t \in [\hat{t}_k, \hat{t}_{k+1})$ . With the optimal solution, the optimal (open-loop) state trajectory may be computed and is denoted as  $x_E^*(t|\hat{t}_k)$ , for  $t \in [\hat{t}_k, \hat{t}_{k+1})$ .

The lower layer LEMPC formulation, which uses a terminal constraint computed from  $x_E^*(\cdot|\hat{t}_k)$ , is given by:

$$\min_{u \in S(\Delta)} \int_{t_j}^{t_{j+N_j}} l_e(\tilde{x}(t), u(t)) dt \quad (6.71a)$$

$$\text{s.t. } \dot{\tilde{x}}(t) = f(\tilde{x}(t), u(t)) \quad (6.71b)$$

$$\tilde{x}(t_j) = x(t_j) \quad (6.71c)$$

$$u(t) \in \mathbb{U}, \forall t \in [t_j, t_{j+N_j}) \quad (6.71d)$$

$$\tilde{x}(t) \in \Omega_\rho, \forall t \in [t_j, t_{j+N_j}] \quad (6.71e)$$

$$\tilde{x}(t_{j+N_j}) = x_E^*(t_{j+N_j}|\hat{t}_k) \quad (6.71f)$$

where  $\tilde{x}$  is the predicted state trajectory under the input trajectory  $u$ . The terminal constraint of Eq. 6.71f is computed from the upper layer LEMPC solution, and serves the same purpose as the terminal constraint of Eq. 6.70f. The optimal solution to the lower layer LEMPC is denoted as  $u^*(t|t_j)$  which is defined for  $t \in [t_j, t_{j+N_j})$ . The control input  $u^*(t_j|t_j)$  is sent to the control actuators to be applied to the system of Eq. 6.65 in a sample-and-hold fashion until the next sampling period.

*Remark 6.5* When the economic stage cost does not penalize the use of control energy, one may consider formulating constraints in the LEMPC problems to prevent the LEMPC from computing an input trajectory that uses excessive control energy. In particular, one straightforward extension of the two-layer LEMPC structure is to compute the total control energy used by the auxiliary controller over the operating window, i.e., integral of the input trajectory  $v$  over  $\hat{t}_k$  to  $\hat{t}_{k+1}$ . Then, enforce that the upper and lower layer LEMPCs compute an input trajectory that uses no more control energy than the auxiliary controller input profile over the operating window. This approach was employed in [21].

### 6.4.3.3 Closed-Loop Stability and Performance

The following proposition proves that the closed-loop system state under the two-layer EMPC structure is always bounded in the invariant set  $\Omega_\rho$  and the economic performance is at least as good as the closed-loop state with the auxiliary LMPC over each operating period.

**Proposition 6.3** *Consider the system of Eq. 6.65 in closed-loop under the lower layer LEMPC of Eq. 6.71. Let the terminal constraint of Eq. 6.71f computed from the upper layer LEMPC of Eq. 6.70, which has a terminal constraint formulated from the auxiliary controller that satisfies Assumption 6.3. Let  $\Delta_E \in (0, \Delta^*]$  where  $\Delta^*$  is defined according to Corollary 2.2,  $N_E \geq 1$ ,  $\bar{N} \geq 1$ , and  $\Delta = \Delta_E / \bar{N}$ . If  $x(\hat{t}_k) \in \Omega_\rho$ , then the state remains bounded in  $\Omega_\rho$  over the operating window with  $x(\hat{t}_{k+1}) = z(\hat{t}_{k+1}|\hat{t}_k) \in \Omega_\rho$ , the upper and lower LEMPCs remain feasible for all  $t \in [\hat{t}_k, \hat{t}_{k+1})$ , and the following inequality holds:*

$$\int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(x(t), u^*(t)) dt \leq \int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(z(t|\hat{t}_k), v(t|\hat{t}_k)) dt \quad (6.72)$$

where  $x$  and  $u^*$  are the closed-loop state and input trajectories and  $z(\cdot|\hat{t}_k)$  and  $v(\cdot|\hat{t}_k)$  denote the open-loop predicted state and input trajectories under the auxiliary computed at  $\hat{t}_k$ .

*Proof Stability:* If  $\Delta_E \in (0, \Delta^*]$  and the auxiliary controller satisfies Assumption 6.3, Eq. 6.67 implies forward invariance of the set  $\Omega_\rho$  under the auxiliary controller. The terminal constraint  $z(\hat{t}_{k+1}|\hat{t}_k)$  computed by the auxiliary controller is therefore in  $\Omega_\rho$ . If the optimization problems are feasible, boundedness of the

closed-loop state in  $\Omega_\rho$  over the operating window follows when  $x(\hat{t}_k) \in \Omega_\rho$  owing to the fact that the constraint of Eq. 6.71e is imposed in the lower layer LEMPC, which is responsible for computing control action for the closed-loop system. Also, the terminal constraint of Eq. 6.71f imposed in the lower layer LEMPC is always in  $\Omega_\rho$  as a result of the constraint of Eq. 6.70e imposed in the upper layer LEMPC.

**Feasibility:** Regarding feasibility of the upper layer LEMPC problem, the input trajectory  $v(\cdot|\hat{t}_k)$  obtained from the auxiliary controller is a feasible solution to the upper layer LEMPC for any  $x(\hat{t}_k) \in \Omega_\rho$  because it maintains the predicted state inside  $\Omega_\rho$  and forces the predicted state to the terminal constraint of Eq. 6.70f. If the auxiliary controller is an explicit controller that satisfies Assumption 6.3, then the input trajectory  $v$  is obtained from recursively solving Eq. 6.68. On the other hand, if the LMPC of Eq. 6.69 is used as the auxiliary controller, then  $v$  is the solution to the optimization problem of Eq. 6.69.

Consider any sampling time  $t_j \in [\hat{t}_k, \hat{t}_{k+1})$  such that  $t_j = \hat{t}_k + i\Delta_E$  for some  $i$  in the set  $\{0, \dots, N_E - 1\}$ , i.e., consider a sampling time of the lower layer LEMPC that corresponds to the beginning of a hold time of the upper layer. Let  $\{\bar{t}_i\}_{i=0}^{N_E-1}$  denote the sequence of such times. The constant input trajectory  $u(t) = u_E^*(t_j|\hat{t}_k)$  for all  $t \in [t_j, t_{j+\bar{N}})$  where  $t_{j+\bar{N}} = \bar{t}_{i+1} = \hat{t}_k + (i+1)\Delta_E$  is a feasible solution to the optimization problem of Eq. 6.71 because it maintains the state in  $\Omega_\rho$  and it forces the state to the terminal constraint of Eq. 6.71f. Owing to the shrinking horizon implementation of the lower layer LEMPC, the computed input trajectory by the lower layer LEMPC at  $t_j = \bar{t}_i$  is a feasible solution to the optimization problem at the next sampling time ( $t_{j+1}$ ) in the sense that if  $u^*(t|t_j)$  defined for  $t \in [t_j, t_j + \bar{N}\Delta)$  is the optimal solution at  $t_j$ , then  $u^*(t|t_j)$  for  $t \in [t_{j+1}, t_{j+1} + (\bar{N}-1)\Delta)$  is a feasible solution at  $t_{j+1}$ . Using this argument recursively until the sampling time  $\bar{t}_{i+1} = \hat{t}_k + (i+1)\Delta_E$  when the horizon is reinitialized to  $\bar{N}$  and then, repeating the arguments for  $\bar{t}_{i+1}$ , it follows that the lower layer LEMPC is feasible.

**Performance:** At  $\bar{t}_i$ , the lower layer LEMPC computes an optimal input trajectory that satisfies (by optimality):

$$\int_{\bar{t}_i}^{\bar{t}_{i+1}} l_e(x^*(t|\bar{t}_i), u^*(t|\bar{t}_i)) dt \leq \int_{\bar{t}_i}^{\bar{t}_{i+1}} l_e(x_E^*(t|\hat{t}_k), u_E^*(\bar{t}_i|\hat{t}_k)) dt \quad (6.73)$$

for all  $i \in \{0, \dots, N_E - 1\}$  (recall,  $\bar{t}_{i+1} = \bar{t}_i + \bar{N}\Delta$ ). Owing to the shrinking horizon and the principle of optimality, the closed-loop state and input trajectories are equal to the computed open-loop state and input trajectories computed at  $\bar{t}_i$  and thus,

$$\int_{\bar{t}_i}^{\bar{t}_{i+1}} l_e(x^*(t|\bar{t}_i), u^*(t|\bar{t}_i)) dt = \int_{\bar{t}_i}^{\bar{t}_{i+1}} l_e(x(t), u^*(t)) dt \quad (6.74)$$

where  $x^*(\cdot|\bar{t}_i)$  and  $u^*(\cdot|\bar{t}_i)$  denote the optimal open-loop state and input trajectories computed at  $\bar{t}_i$  and  $x$  and  $u^*$  are the closed-loop state and input trajectories. There-

fore, from Eqs. 6.73–6.74, the closed-loop performance over one operating period is bounded by:

$$\begin{aligned}
 \int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(x(t), u^*(t)) dt &= \sum_{i=0}^{N_E-1} \int_{\bar{t}_i}^{\bar{t}_{i+1}} l_e(x(t), u^*(t|\bar{t}_i)) dt \\
 &\leq \sum_{i=0}^{N_E-1} \int_{\hat{t}_i}^{\bar{t}_{i+1}} l_e(x_E^*(t|\hat{t}_k), u_E^*(t|\hat{t}_k)) dt \\
 &= \int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(x_E^*(t|\hat{t}_k), u_E^*(t|\hat{t}_k)) dt . \quad (6.75)
 \end{aligned}$$

At  $\hat{t}_k$ , the upper layer LEMPC computes an optimal input trajectory. Owing to optimality, the computed (open-loop) state and input trajectories of the upper layer LEMPC satisfies:

$$\int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(x_E^*(t|\hat{t}_k), u_E^*(t|\hat{t}_k)) dt \leq \int_{\hat{t}_k}^{\hat{t}_{k+1}} l_e(z(t|\hat{t}_k), v(t|\hat{t}_k)) dt . \quad (6.76)$$

From Eqs. 6.75–6.76, the result of Eq. 6.72 follows.

The following theorem provides sufficient conditions such that the two-layer EMPC structure maintains the closed-loop state inside the region  $\Omega_\rho$  and the closed-loop economic performance is at least as good as if the auxiliary LMPC was applied to the system of Eq. 6.65 over the entire length of operation which may be finite or infinite.

**Theorem 6.3** *Consider the closed-loop system of Eq. 6.65 under the lower layer LEMPC of Eq. 6.71. Let the terminal constraint of Eq. 6.71f computed from the upper layer LEMPC of Eq. 6.70, which has a terminal constraint formulated from the auxiliary controller that satisfies Assumption 6.3, and let the assumptions of Proposition 6.3 hold. If  $x(0) \in \Omega_\rho$ , then  $x(t) \in \Omega_\rho$  for all  $t \geq 0$  and the following inequality holds for finite-time operation:*

$$\int_0^T l_e(x(t), u^*(t)) dt \leq \int_0^T l_e(z(t), v(t)) dt \quad (6.77)$$

where  $T = KN_E\Delta_E$  and  $K$  is any strictly positive integer and  $x$  and  $u^*$  are the closed-loop state and input trajectory and  $z$  and  $v$  are the resulting state and input trajectory from the auxiliary controller defined over the interval  $[0, T]$  with initial condition  $z(0) = x(0) \in \Omega_\rho$ . The following inequality holds for infinite-time operation:

$$\limsup_{K \rightarrow \infty} \frac{1}{KN_E\Delta_E} \int_0^{KN_E\Delta_E} l_e(x(t), u^*(t)) dt \leq l_e(x_s^*, u_s^*) . \quad (6.78)$$



*Proof* Applying the results of Proposition 6.3 recursively over  $K$  operating periods, recursive feasibility of the optimization problems follows, and the closed-loop state is always bounded in  $\Omega_\rho$  if  $x(0) \in \Omega_\rho$ , and  $x(\hat{t}_k) = z(\hat{t}_k)$  for  $k = 1, 2, \dots, K$ . To show the result of Eq. 6.77, the length of operation is divided into  $K$  operating periods and let  $T = KN_E\Delta_E$ :

$$\int_0^T l_e(x(t), u^*(t)) dt = \int_0^{\hat{t}_1} l_e(x(t), u^*(t)) dt + \dots + \int_{\hat{t}_{K-1}}^{\hat{t}_K} l_e(x(t), u^*(t)) dt \quad (6.79)$$

where  $\hat{t}_K = T$ . By Proposition 6.3, the inequality of Eq. 6.72 holds over each operating window when  $x(\hat{t}_k) = z(\hat{t}_k)$  for  $k = 1, 2, \dots, K$  and thus, the inequality of Eq. 6.77 follows.

Owing to the result of Eq. 6.77, the average finite-time economic cost is given by:

$$\frac{1}{T} \int_0^T l_e(x(t), u^*(t)) dt \leq \frac{1}{T} \int_0^T l_e(z(t), v(t)) dt \quad (6.80)$$

for  $T = KN_E\Delta_E$  where  $K$  is any strictly positive integer. Recall, the economic cost function  $l_e$  is continuous on the compact set  $\Omega_\rho \times \mathbb{U}$  and  $x(t), z(t) \in \Omega_\rho$  and  $u^*(t), v(t) \in \mathbb{U}$  for all  $t \geq 0$ . Thus, both integrals of Eq. 6.80 are bounded for any  $T > 0$ . Since the auxiliary controller satisfies Assumption 6.3 and  $\Delta \in (0, \Delta^*]$ ,  $z$  and  $v$  asymptotically converge to the steady-state  $(x_s^*, u_s^*)$  (this follows from the inequality of Eq. 6.67).

Considering the limit of the right-hand side of Eq. 6.80 as  $T$  tends to infinity (or similarly, as  $K$  tends to infinity), the limit exists and is equal to  $l_e(x_s^*, u_s^*)$  because  $z$  and  $v$  asymptotically converge to optimal steady-state  $(x_s^*, u_s^*)$  while remaining bounded for all  $t \geq 0$ . To prove this limit, the following result is shown: given  $\varepsilon > 0$ , there exists  $T^* > 0$  such that for all  $T > T^*$ , the following holds:

$$\left| \frac{1}{T} \int_0^T l_e(z(t), v(t)) dt - l_e(x_s^*, u_s^*) \right| < \varepsilon. \quad (6.81)$$

Define  $I(0, T)$  as the following integral:

$$I(0, T) := \int_0^T l_e(z(t), v(t)) dt \quad (6.82)$$

where the arguments of  $I$  represent the lower and upper limits of integration, respectively. The trajectories  $z(t)$  and  $v(t)$  converge to  $x_s^*$  and  $u_s^*$ , respectively, as  $t$  tends to infinity. Furthermore,  $z(t) \in \Omega_\rho$  and  $v(t) \in \mathbb{U}$  for all  $t \geq 0$ , so for every  $\varepsilon > 0$ , there exists a  $\tilde{T} > 0$  such that

$$|l_e(z^*(t), v^*(t)) - l_e(x_s^*, u_s^*)| < \varepsilon/2 \quad (6.83)$$

for  $t \geq \tilde{T}$ . For any  $T > \tilde{T}$ :

$$\begin{aligned}
 |I(0, T) - Tl_e(x_s^*, u_s^*)| &= \left| I(0, \tilde{T}) + I(\tilde{T}, T) - Tl_e(x_s^*, u_s^*) \right| \\
 &\leq \int_0^{\tilde{T}} |l_e(z(t), v(t)) - l_e(x_s^*, u_s^*)| dt \\
 &\quad + \int_{\tilde{T}}^T |l_e(z(t), v(t)) - l_e(x_s^*, u_s^*)| dt \\
 &\leq \tilde{T}\tilde{M} + (T - \tilde{T})\varepsilon/2
 \end{aligned} \tag{6.84}$$

where  $\tilde{M} := \sup_{t \in [0, \tilde{T}]} \{|l_e(z(t), v(t)) - l_e(x_s^*, u_s^*)|\}$ . For any  $T > T^*$  where  $T^* = 2\tilde{T}(\tilde{M} - \varepsilon/2)/\varepsilon$ , the following inequality is satisfied:

$$|I(0, T)/T - l_e(x_s^*, u_s^*)| \leq (1 - \tilde{T}/T)\varepsilon/2 + \tilde{T}\tilde{M}/T < \varepsilon \tag{6.85}$$

which proves that the asymptotic average economic cost under the auxiliary controller is  $l_e(x_s^*, u_s^*)$ .

Considering the left hand side of Eq. 6.80, the limit as  $K \rightarrow \infty$  may not exist owing to the possible time-varying system operation under the proposed two-layer LEMPC scheme. Therefore, an upper bound on the asymptotic average performance under the LEMPC scheme is considered. Since the limit superior is equal to the limit when the limit exists, the following is obtained:

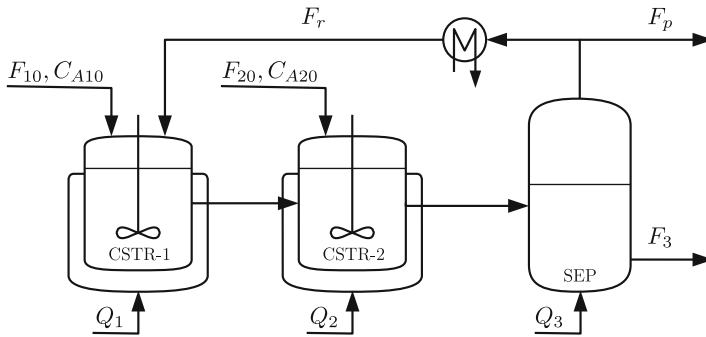
$$\begin{aligned}
 \limsup_{K \rightarrow \infty} \frac{1}{KN_E\Delta_E} \int_0^{KN_E\Delta_E} l_e(x(t), u^*(t)) dt \\
 \leq \limsup_{K \rightarrow \infty} \frac{1}{KN_E\Delta_E} \int_0^{KN_E\Delta_E} l_e(z(t), v(t)) dt = l_e(x_s^*, u_s^*)
 \end{aligned} \tag{6.86}$$

which is the desired result of Eq. 6.78.

*Remark 6.6* The finite-time result of Theorem 6.3 may be extended to any  $T > 0$  by, for instance, adjusting  $N_E$  and/or  $\Delta_E$  in the last operating window.

#### 6.4.4 Application to Chemical Process Example

Consider a three vessel chemical process network consisting of two continuously stirred tank reactors (CSTRs) in series followed by a flash tank separator. The process flow diagram of the process network is shown in Fig. 6.14. In each of the reactors, an irreversible second-order reaction of the form  $A \rightarrow B$  takes place in an inert solvent  $D$  ( $A$  is the reactant and  $B$  is the desired product). The bottom stream of the flash tank is the product stream of the process network. Part of the overhead



**Fig. 6.14** Process flow diagram of the reactor and separator process network

vapor stream from the flash tank is purged from the process while the remainder is fully condensed and recycled back to the first reactor. Each of the vessels have a heating/cooling jacket to supply/remove heat from the liquid contents of the vessel. The following indexes are used to refer to each vessel:  $i = 1$  denotes CSTR-1,  $i = 2$  denotes CSTR-2, and  $i = 3$  denotes SEP-1. The heat rate supplied/removed from the  $i$ th vessel is  $Q_i$ ,  $i = 1, 2, 3$ . Furthermore, each reactor is fed with fresh feedstock containing  $A$  in the solvent  $D$  with concentration  $C_{Ai0}$ , volumetric flow rate  $F_{i0}$ , and constant temperature  $T_{i0}$  where  $i = 1, 2$ . Applying first principles and standard modeling assumptions, a dynamic model of the reactor-separator process network may be obtained (neglecting the dynamics of the condenser) and is given by the following ODEs (see Table 6.4 for variable definitions and values):

$$\frac{dT_1}{dt} = \frac{F_{10}}{V_1}(T_{10} - T_1) + \frac{F_r - F_p}{V_1}(T_3 - T_1) - \frac{\Delta H k_0}{\rho_L C_p} e^{-E/RT_1} C_{A1}^2 + \frac{Q_1}{\rho_L C_p V_1} \quad (6.87a)$$

$$\frac{dC_{A1}}{dt} = \frac{F_{10}}{V_1}(C_{A10} - C_{A1}) + \frac{F_r - F_p}{V_1}(C_{Ar} - C_{A1}) - k_0 e^{-E/RT_1} C_{A1}^2 \quad (6.87b)$$

$$\frac{dC_{B1}}{dt} = -\frac{F_{10}}{V_1} C_{B1} + \frac{F_r - F_p}{V_1}(C_{Br} - C_{B1}) + k_0 e^{-E/RT_1} C_{A1}^2 \quad (6.87c)$$

$$\frac{dT_2}{dt} = \frac{F_{20}}{V_2}(T_{20} - T_2) + \frac{F_1}{V_2}(T_1 - T_2) - \frac{\Delta H k_0}{\rho_L C_p} e^{-E/RT_2} C_{A2}^2 + \frac{Q_2}{\rho_L C_p V_2} \quad (6.87d)$$

$$\frac{dC_{A2}}{dt} = \frac{F_{20}}{V_2}(C_{A20} - C_{A2}) + \frac{F_1}{V_2}(C_{A1} - C_{A2}) - k_0 e^{-E/RT_2} C_{A2}^2 \quad (6.87e)$$

$$\frac{dC_{B2}}{dt} = -\frac{F_{20}}{V_2} C_{B2} + \frac{F_1}{V_2}(C_{B1} - C_{B2}) + k_0 e^{-E/RT_2} C_{A2}^2 \quad (6.87f)$$

$$\frac{dT_3}{dt} = \frac{F_2}{V_3}(T_2 - T_3) - \frac{\Delta H_{\text{vap}} F_r}{\rho_L C_p V_3} + \frac{Q_3}{\rho_L C_p V_3} \quad (6.87g)$$

**Table 6.4** Process parameters of the reactor and separator process network

Symbol/Value	Description
$T_{10} = 300 \text{ K}$	Temp.: CSTR-1 inlet
$T_{20} = 300 \text{ K}$	Temp.: CSTR-2 inlet
$F_{10} = 5.0 \text{ m}^3 \text{ h}^{-1}$	Flow rate: CSTR-1 inlet
$F_{20} = 5.0 \text{ m}^3 \text{ h}^{-1}$	Flow rate: CSTR-2 inlet
$F_r = 3.0 \text{ m}^3 \text{ h}^{-1}$	Flow rate: SEP-1 vapor
$F_p = 0.5 \text{ m}^3 \text{ h}^{-1}$	Flow rate: purge stream
$V_1 = 1.5 \text{ m}^3$	Volume: CSTR-1
$V_2 = 1.0 \text{ m}^3$	Volume: CSTR-2
$V_3 = 1.0 \text{ m}^3$	Volume: SEP-1
$k_0 = 3.0 \times 10^6 \text{ m}^3 \text{ kmol}^{-1} \text{ h}^{-1}$	Pre-exponential factor
$E = 3.0 \times 10^4 \text{ kJ kmol}^{-1}$	Activation energy
$\Delta H = -5.0 \times 10^3 \text{ kJ kmol}^{-1}$	Heat of reaction
$\Delta H_{\text{vap}} = 5.0 \text{ kJ kmol}^{-1}$	Heat of vaporization
$C_p = 0.231 \text{ kJ kg}^{-1} \text{ K}^{-1}$	Heat capacity
$R = 8.314 \text{ kJ kmol}^{-1} \text{ K}^{-1}$	Gas constant
$\rho_L = 1000 \text{ kg m}^{-3}$	Density
$\alpha_A = 5.0$	Relative volatility: A
$\alpha_B = 0.5$	Relative volatility: B
$\alpha_D = 1.0$	Relative volatility: D
$MW_A = 18.0 \text{ kg kmol}^{-1}$	Molecular weight: A
$MW_B = 18.0 \text{ kg kmol}^{-1}$	Molecular weight: B
$MW_D = 40.0 \text{ kg kmol}^{-1}$	Molecular weight: D

$$\frac{dC_{A3}}{dt} = \frac{F_2}{V_3} C_{A2} - \frac{F_r}{V_3} C_{Ar} - \frac{F_3}{V_3} C_{A3} \quad (6.87h)$$

$$\frac{dC_{B3}}{dt} = \frac{F_2}{V_3} C_{B2} - \frac{F_r}{V_3} C_{Br} - \frac{F_3}{V_3} C_{B3} \quad (6.87i)$$

and the following algebraic equations:

$$K = \frac{1}{\rho_L} \sum_{i \in \{A, B, D\}} \alpha_i C_{i3} MW_i, \quad (6.88a)$$

$$C_{ir} = \alpha_i C_{i3} / K, \quad i = A, B, D, \quad (6.88b)$$

$$F_1 = F_r - F_p + F_{10}, \quad F_2 = F_1 + F_{20}, \quad (6.88c)$$

$$F_3 = F_2 - F_r. \quad (6.88d)$$

where  $C_{ir}$  is the concentration of the  $i$ th component ( $i = A, B, D$ ) in the flash separator overhead, purge, and recycle streams. The state variables of the process network include the temperatures and concentrations of  $A$  and  $B$  in each of the vessels:

$$x^T = [T_1 \ C_{A1} \ C_{B1} \ T_2 \ C_{A2} \ C_{B2} \ T_3 \ C_{A3} \ C_{B3}] .$$

The manipulated inputs are the heat inputs to the three vessels,  $Q_1$ ,  $Q_2$ , and  $Q_3$ , and the concentration of  $A$  in the inlet streams,  $C_{A10}$  and  $C_{A20}$ :

$$u^T = [Q_1 \ Q_2 \ Q_3 \ C_{A10} \ C_{A20}] .$$

The control objective is to regulate the process in an economically optimal time-varying fashion to maximize the average amount of product  $B$  in the product stream  $F_3$ . Continuously feeding in the maximum concentration of  $A$  into each reactor maximizes the production of  $B$  owing to the second-order reaction. However, this may not be practical from an economic stand-point. Instead, the average amount of reactant material that may be fed to each reactor is fixed motivating the use of EMPC to control the process network. In addition, supplying/removing heat to/from the vessels is considered undesirable. To accomplish these economic considerations, the two-layer LEMPC structure is applied and the upper and lower layer LEMPCs are formulated with the following economic stage cost function and constraint, respectively:

$$l_e(x, u) = -F_3 C_{B3} + p_1 Q_1^2 + p_2 Q_2^2 + p_3 Q_3^2 \quad (6.89)$$

$$\frac{1}{t'} \int_{\hat{t}_k}^{\hat{t}_{k+1}} (C_{A10} + C_{A20}) dt = 8.0 \text{ kmol m}^3 \quad (6.90)$$

where  $t' = 1.0$  h is the operating period length and  $p_i = 10^{-6}$ ,  $i = 1, 2, 3$  are the penalty weights for using energy. The value for the heat rate penalty has been chosen to account for the different numerical range of the heat rate and the first term in the economic cost (molar flow rate of  $B$  in the product stream). The economically optimal steady-state with respect to the economic cost function of Eq. 6.89 is open-loop asymptotically stable and is the only steady-state in the operating region of interest. An explicit characterization of  $\Omega_\rho$  is not needed for the LEMPC implementation.

The two-layer LEMPC structure, formulated with the cost function and reactant material constraint of Eqs. 6.89–6.90, respectively, is applied to the reactor-separator chemical process network. To numerically integrate the dynamic model of Eq. 6.87, explicit Euler method is used with an integration step of  $1.0 \times 10^{-3}$  h. The auxiliary controller is formulated as an auxiliary LMPC. The prediction horizon and sampling period of the auxiliary LMPC and upper layer LEMPC are  $N_E = 10$  and  $\Delta_E = 0.1$  h, respectively, while, the lower layer LEMPC is formulated with a prediction horizon of  $\bar{N} = 2$  and sampling period  $\Delta = 0.05$  h. Since the upper layer prediction horizon length is one hour, the reactant material constraint is enforced over each one hour operating period. However, the lower layer LEMPC prediction horizon does not

cover the entire one hour operating window. Instead of using the material constraint of Eq. 6.90 directly in the lower layer LEMPC, a constraint is formulated on the basis of the upper layer LEMPC solution. Namely, over the prediction horizon of the lower layer LEMPC, the lower layer LEMPC solution must use the same amount of reactant material as that of the upper layer LEMPC solution over the same time so that the material constraint is satisfied over the operating window. To solve the optimization problems, Ipopt [17] is used and the simulations were completed on a desktop PC with an Intel® Core™ 2 Quad 2.66 GHz processor and a Linux operating system.

#### 6.4.4.1 Effect of Horizon Length

In the first set of simulations, the length of the prediction horizon on closed-loop performance is considered. The closed-loop economic performance over 4.0h is defined by the total economic cost given by:

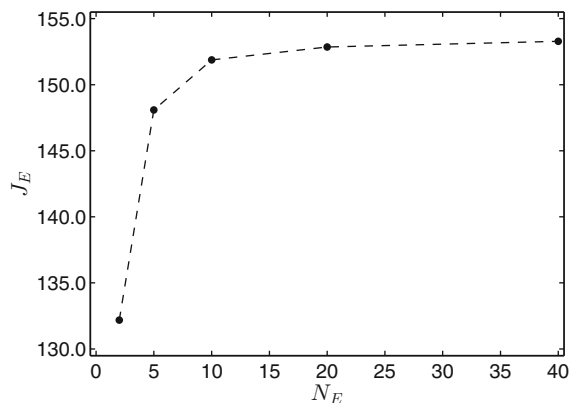
$$J_E = \int_0^{4.0} (F_3 C_{B3} - p_1 Q_1^2 - p_2 Q_2^2 - p_3 Q_3^2) dt . \quad (6.91)$$

In these simulations, only the upper layer LEMPC, formulated with a terminal constraint computed from the auxiliary LMPC, is considered. Figure 6.15 depicts the observed trend. As the prediction horizon increases, the closed-loop economic performance increases, which motivates the use of a long prediction horizon in EMPC.

#### 6.4.4.2 Effect of the Terminal Constraint

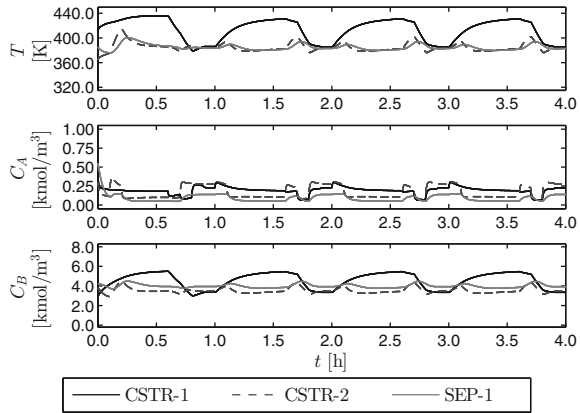
Since for any optimization problem, the addition of constraints may restrict the feasible region of the optimization problem, a reasonable consideration is the effect of the terminal constraint on closed-loop performance. To address this issue, consider

**Fig. 6.15** The closed-loop economic performance ( $J_E$ ) with the length of prediction horizon ( $N_E$ ) for the reactor-separator process under the upper layer LEMPC with a terminal constraint computed from an auxiliary LMPC

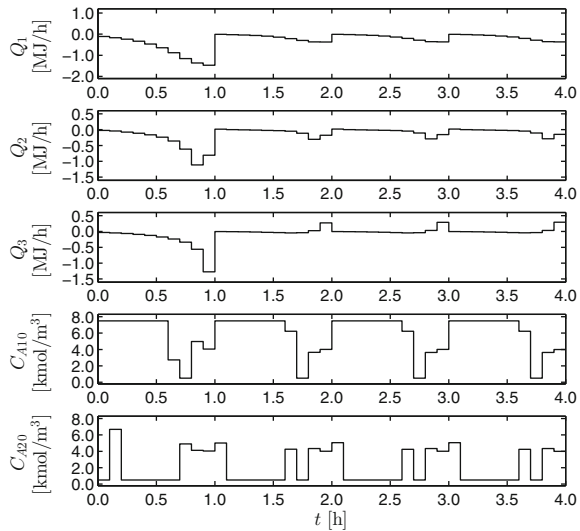


the closed-loop system under the upper layer LEMPC formulated with a terminal equality constraint computed by the auxiliary LMPC and under an LEMPC (mode 1 operation only) formulated with the economic cost of Eq. 6.89 and the material constraint of Eq. 6.90, but without terminal constraints. Both use a prediction horizon of  $N_E = 10$  and a sampling period of  $\Delta = 0.01$  h. Figures 6.16 and 6.17 display the closed-loop state and input trajectories of the reactor-separator process network with the upper layer LEMPC; while, Figs. 6.18 and 6.19 display the closed-loop trajectories under LEMPC with no terminal constraints.

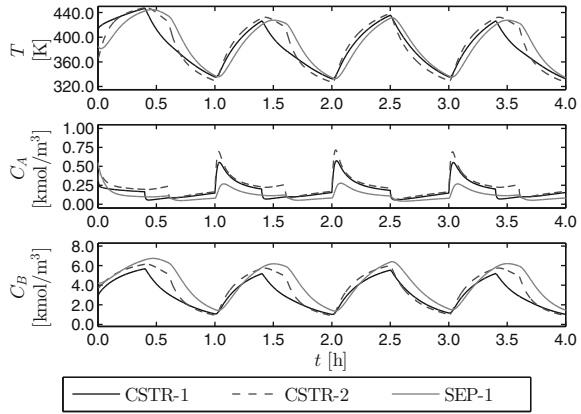
**Fig. 6.16** Closed-loop state trajectories of the reactor-separator process network with the upper layer LEMPC formulated with a terminal constraint computed by the auxiliary LMPC



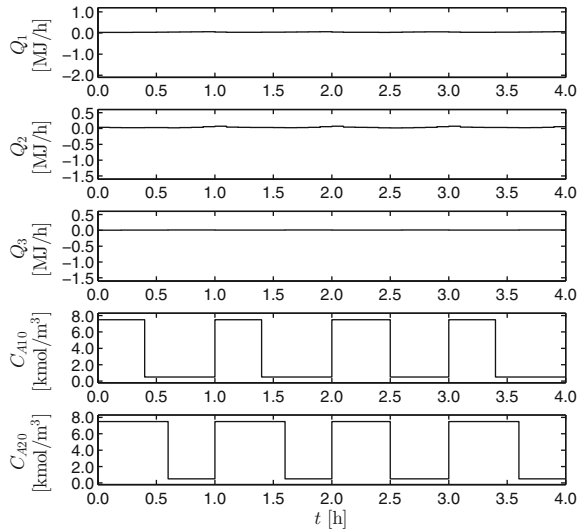
**Fig. 6.17** Input trajectories of the reactor-separator process network computed by the upper layer LEMPC formulated with a terminal constraint computed by the auxiliary LMPC



**Fig. 6.18** Closed-loop state trajectories of the reactor-separator process network with an LEMPC formulated without terminal constraints



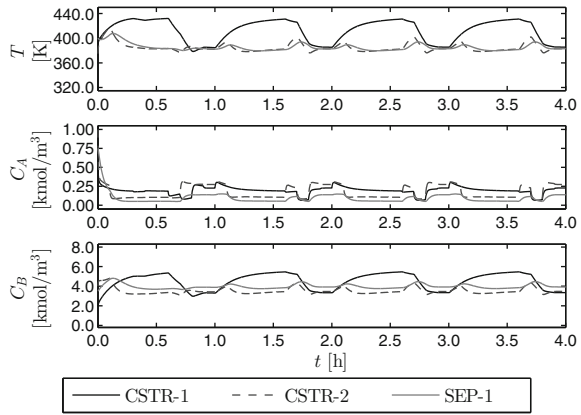
**Fig. 6.19** Input trajectories of the reactor-separator process network computed by an LEMPC formulated without terminal constraints



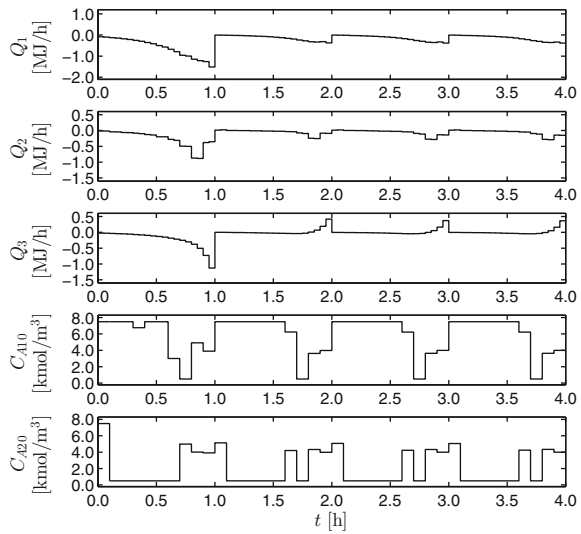
The reactor-separator process network under the LEMPC with the terminal constraint evolves in a smaller operating range (370–430 K) than the evolution under the LEMPC without the terminal constraint (325– 440 K). The total economic cost with the upper layer LEMPC (based on the auxiliary LMPC) is 151.2, while the total economic cost with LEMPC formulated without terminal constraints is 159.3. The terminal constraint imposed in the LEMPC problem affects the achievable performance. However, the key advantage of the addition of this constraint is that for any system and any prediction horizon the closed-loop economic performance under the two-layer LEMPC structure is guaranteed to be at least as good as a stabilizing controller for both finite-time and infinite-time operating intervals.



**Fig. 6.20** Closed-loop state trajectories of the reactor-separator process network with the two-layer LEMPC structure



**Fig. 6.21** Input trajectories of the reactor-separator process network computed by the two-layer LEMPC structure



**6.4.4.3 Two-Layer LEMPC Structure**

The two-layer LEMPC structure with a terminal constraint computed from an auxiliary LMPC is applied to the reactor-separator process network. Several closed-loop simulations over a 4.0h length of operation are completed. The closed-loop state and input trajectories of one of the simulations are shown in Figs. 6.20 and 6.21, respectively and demonstrate time-varying operation of the process network. The economic performance (Eq. 6.91) is compared to the economic performance with the auxiliary LMPC (Table 6.5). From this comparison, an average of 10% benefit with the two-layer LEMPC structure is realized over operation under the auxiliary LMPC, i.e., resulting in steady-state operation.

**Table 6.5** Total economic cost and average computational time in seconds per sampling period for several 4.0h simulations with: (a) the auxiliary LMPC, (b) the one-layer LEMPC and (c) the two-layer LEMPC structure

Sim.	LMPC	One-layer EMPC		Two-layer EMPC	
	Cost	CPU time	Cost	CPU time	Cost
1	140.1	5.68	151.5	1.10	151.1
2	150.3	4.24	153.9	1.05	153.4
3	142.0	4.65	152.4	0.98	152.0
4	130.7	6.45	152.3	1.24	151.9
5	126.0	4.67	151.9	1.11	151.5
6	140.2	4.63	151.6	1.33	151.2
7	144.6	4.60	150.6	1.08	150.2
8	138.1	5.01	152.5	1.06	152.1

Additionally, a comparison between the computational time required to solve the two-layer LEMPC system and that of a one-layer LEMPC system is completed. The one-layer LEMPC system consists of the upper layer LEMPC with a terminal constraint computed from the auxiliary LMPC. In the one-layer LEMPC system, the LEMPC applies its computed control actions directly to the process network, and there is no lower layer LEMPC. To make the comparison consistent, the one layer LEMPC is implemented with a prediction horizon of  $N_E = 20$  and a sampling period of  $\Delta_E = 0.05$  h, which are the same sampling period and horizon used in the lower layer LEMPC of the two-layer LEMPC system. Since the upper and lower layer controllers are sequentially computed, the computational time at the beginning of each operating window is measured as the sum of the computational time to solve the auxiliary LMPC, the upper layer LEMPC, and the lower layer LEMPC for the two-layer LEMPC system and as the sum of the time to solve the auxiliary LMPC and the LEMPC for the one-layer LEMPC system. From Table 6.5, the one-layer LEMPC achieves slightly better closed-loop economic performance because the one-layer LEMPC uses a smaller sampling period than the upper layer LEMPC in the two-layer LEMPC structure. However, the computational time required to solve the one-layer LEMPC structure is greater than the computational time of the two-layer LEMPC structure. The two-layer LEMPC structure is able to reduce the computational time by about 75% on average.

#### 6.4.4.4 Handling Disturbances

While the two-layer EMPC has been designed for nominal operation to guarantee finite-time and infinite-time closed-loop performance as is at least as good as that achieved under a stabilizing controller, it may be applied to the process model in the presence of disturbances, plant/model mismatch, and other uncertainties with

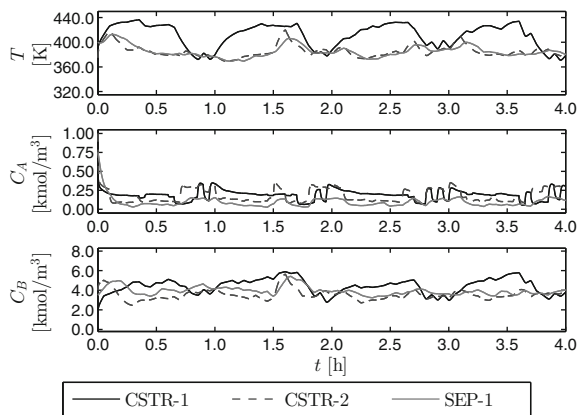
some modifications to improve recursive feasibility of the optimization problems and to ensure greater robustness of the controller to uncertainties. For instance, if the disturbances are relatively small, it may be sufficient to relax the terminal constraints or treat them as soft constraints. If one were to simply relax the terminal constraints, e.g., use a terminal region instead of a point-wise terminal constraint, it is difficult to guarantee recursive feasibility of the optimization problem. Another potential methodology is to treat the terminal state constraints as a soft constraint instead of imposing them as hard constraints. For example, use a cost functional in the lower layer LEMPC of the form:

$$\left( \int_{t_j}^{t_{j+N}} l_e(\tilde{x}(t), u(t)) dt \right) + |\tilde{x}(t_{j+N}) - x_E^*(t_j|t_k)|_Q \tag{6.92}$$

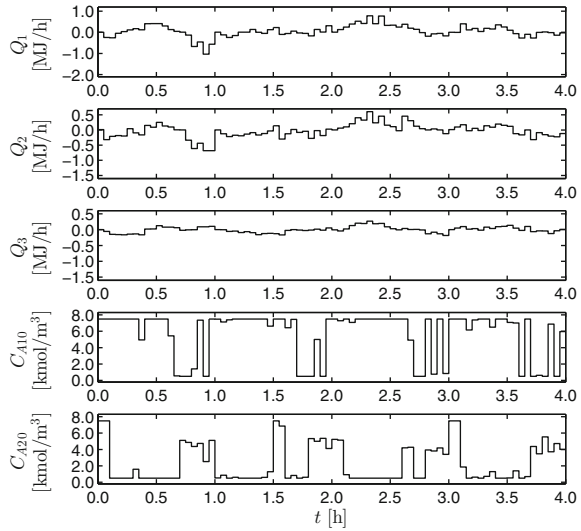
where  $Q$  is a positive definite weighting matrix. The cost functional works to optimize the economic performance while ensuring the predicted evolution is near the terminal state through the quadratic terminal cost. The resulting lower layer LEMPC has the same stability and robustness to bounded disturbances properties as the LEMPC (without terminal constraints), i.e., recursive feasibility and boundedness of the closed-loop state for all initial states starting in  $\Omega_\rho$ . While no provable performance guarantees may be made on closed-loop performance in the presence of disturbances, the closed-loop performance benefit may be evaluated through simulations.

The two-layer LEMPC with the lower layer LEMPC designed with the cost described above in Eq. 6.92 and without terminal constraints is applied to the example with significant process noise added. The noise is modeled as bounded Gaussian white noise and is introduced additively to each model state. The closed-loop state and input trajectories are shown in Figs. 6.22 and 6.23, respectively. The closed-loop system performance under the two-layer LEMPC is compared to the system under auxiliary LMPC with the same realization of the process noise. The LMPC is formu-

**Fig. 6.22** Closed-loop state trajectories of the reactor-separator process network with process noise added with the two-layer LEMPC structure



**Fig. 6.23** Input trajectories of the reactor-separator process network with process noise added computed by the two-layer LEMPC structure



lated with a prediction horizon of  $N = 2$  and sampling period  $\Delta = 0.05$  h which is the same horizon and sampling period as the lower layer LEMPC. The closed-loop performance under the two-layer LEMPC is 2.6 % better than that under the LMPC for this particular realization of the process noise.

## 6.5 Conclusions

In this chapter, several computationally-efficient two-layer frameworks for integrating dynamic economic optimization and control of nonlinear systems were presented. In the upper layer, EMPC was employed to compute economically optimal time-varying operating trajectories. Explicit control-oriented constraints were employed in the upper layer EMPC. In the lower layer, an MPC scheme was used to force the system to track the optimal time-varying trajectory computed by the upper layer EMPC. The properties, i.e., stability, performance, and robustness, of closed-loop systems under the two-layer EMPC methods were rigorously analyzed. The two-layer EMPC methods were applied to chemical process examples to demonstrate the closed-loop properties. In all the examples considered, closed-loop stability was achieved, the closed-loop economic performance under the two-layer EMPC framework was better than that achieved under conventional approaches to optimization and control, and the total on-line computational time was better with the two-layer EMPC methods compared to that under one-layer EMPC methods.

## References

1. Helbig A, Abel O, Marquardt W (2000) Structural concepts for optimization based control of transient processes. In: Allgöwer F, Zheng A (eds) *Nonlinear model predictive control. Progress in systems and control theory*, vol 26. Birkhäuser, Basel, pp 295–311
2. Kadam JV, Schlegel M, Marquardt W, Tousain RL, van Hessem DH, van den Berg J, Bosgra OH (2002) A two-level strategy of integrated dynamic optimization and control of industrial processes—a case study. In: Grievink J, van Schijndel J (eds) *Proceedings of the European symposium on computer aided process engineering-12, computer aided chemical engineering*, vol 10. The Netherlands, The Hague, pp 511–516
3. Marquardt W (2001) Nonlinear model reduction for optimization based control of transient chemical processes. In: *Proceedings of the Sixth international conference on chemical process control*, Tucson, AZ, , pp 12–42
4. Kadam JV, Marquardt W, Schlegel M, Backx T, Bosgra OH, Brouwer PJ, Dünnebieber G, van Hessem D, Tiagounov A, de Wolf S (2003) Towards integrated dynamic real-time optimization and control of industrial processes. In: *Proceedings of the 4th international conference of the foundations of computer-aided process operations*. Coral Springs, Florida, pp 593–596
5. Tosukhowong T, Lee JM, Lee JH, Lu J (2004) An introduction to a dynamic plant-wide optimization strategy for an integrated plant. *Comput Chem Eng* 29:199–208
6. Zhu X, Hong W, Wang S (2004) Implementation of advanced control for a heat-integrated distillation column system. In: *Proceedings of the 30th Conference of IEEE industrial electronics society*, pp 2006–2011, Busan, Korea
7. Kadam JV, Marquardt W (2007) Integration of economical optimization and control for intentionally transient process operation. In: Findeisen R, Allgöwer F, Biegler LT (eds) *Assessment and future directions of nonlinear model predictive control*, vol 358., *Lecture notes in control and information sciences*, Springer, Berlin, pp 419–434
8. Würth L, Hannemann R, Marquardt W (2009) Neighboring-extremal updates for nonlinear model-predictive control and dynamic real-time optimization. *J Process Control* 19:1277–1288
9. Würth L, Rawlings JB, Marquardt W (2007) Economic dynamic real-time optimization and nonlinear model-predictive control on the infinite horizon. In: *Proceedings of the 7th IFAC international symposium on advanced control of chemical processes*, pp 219–224, Istanbul, Turkey
10. Würth L, Hannemann R, Marquardt W (2011) A two-layer architecture for economically optimal process control and operation. *J Process Control* 21:311–321
11. Ellis M, Christofides PD (2014) Optimal time-varying operation of nonlinear process systems with economic model predictive control. *Ind Eng Chem Res* 53:4991–5001
12. Ellis M, Christofides PD (2014) Integrating dynamic economic optimization and model predictive control for optimal operation of nonlinear process systems. *Control Eng Pract* 22:242–251
13. Ellis M, Christofides PD (2014) On finite-time and infinite-time cost improvement of economic model predictive control for nonlinear systems. *Automatica* 50:2561–2569
14. Ellis M, Christofides PD (2014) Selection of control configurations for economic model predictive control systems. *AIChE J* 60:3230–3242
15. Khalil HK (2002) *Nonlinear systems*, 3rd edn. Prentice Hall, Upper Saddle River
16. Massera JL (1956) Contributions to stability theory. *Ann Math* 64:182–206
17. Wächter A, Biegler LT (2006) On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming. *Math Progr* 106:25–57
18. Christofides PD, El-Farra NH (2005) *Control of nonlinear and hybrid process systems: designs for uncertainty. Constraints and time-delays*. Springer, Berlin
19. Kokotović P, Arcak M (2001) *Constructive nonlinear control: a historical perspective*. *Automatica* 37:637–662
20. Sontag ED (1998) *Mathematical control theory: deterministic finite dimensional systems*, vol 6. Springer, Berlin

21. Heidarinejad M, Liu J, Christofides PD (2013) Algorithms for improved fixed-time performance of Lyapunov-based economic model predictive control of nonlinear systems. *J Process Control* 23:404–414