2009 American Control Conference
Hyatt Regency Riverfront, St. Louis, MO, USA
June 10-12, 2009

Distributed Model Predictive Control of Nonlinear Systems with Input Constraints

Jinfeng Liu, David Muñoz de la Peña and Panagiotis D. Christofides

Abstract— **In this work, we introduce a distributed Lyapunov-based model predictive control method for nonlinear systems with input constraints. The class of systems considered arises naturally when new sensors, actuators and controllers are added to already operating control loops to improve closed-loop performance, taking advantage from the latest advances in sensor/actuator network technology. Assuming that there exists a Lyapunov-based controller that stabilizes the closed-loop system using the pre-existing control loops, we propose to use Lyapunov-based model predictive control to design two separate predictive controllers that compute the optimal input trajectories in a distributed manner. The proposed distributed control scheme preserves the stability properties of the Lyapunov-based controller while satisfying input constraints and improving the closed-loop performance. The theoretical results are illustrated using a chemical process example.**

I. INTRODUCTION

Optimal process operation and management of abnormal situations during plant operation are major challenges in the process industries. This realization has motivated extensive research in the area of chemical process control to ensure safe and efficient process operation. From a control architecture standpoint, control systems traditionally utilize dedicated, point-to-point wired communication links to measurement sensors and control actuators to regulate process variables at desired values. While this paradigm to process control has been successful, we are currently witnessing an augmentation of the existing, dedicated local control networks, with additional networked (wired and/or wireless) actuator/sensor devices which have become cheap and easy-to-install the last few years. Such an augmentation in sensor information and networked-based availability of data has the potential [1], [2], [3], [4], [5] to be transformative in the sense of dramatically improving the ability of the control systems to optimize process performance (i.e., achieving control objectives that go well beyond the ones that can be achieved with dedicated, local control systems) and prevent or deal with abnormal situations more quickly and effectively (fault-tolerance). The addition of networked sensors and actuators allows for easy modification of the control strategy by rerouting signals, having redundant systems that can be activated automatically when failures occur, and in general, they allow having improved control over the entire plant.

However, augmenting dedicated, local control networks with real-time sensor and actuator networks gives rise to the need to redesign and coordinate separate control systems that operate on a process. There are several ways to deal with this problem. One is to design a centralized controller to decide the manipulated inputs of all the actuators, and another one is to design separate controllers that coordinate their actions and take into account interactions between subsystems (i.e., distributed control architecture). Model predictive control (MPC) is a natural control framework to deal with the design of distributed control systems because of its ability to handle input and state constraints, and also because it can account for the actions of other actuators in computing the control action of a given set of control actuators in real-time. In the context of centralized control design using MPC, a model of the overall process is used and the control actions are computed while minimizing an overall objective function. Usually, centralized MPC gives better closed-loop performance compared with distributed MPC schemes. However, with the increase of optimization (decision) variables, the computational complexity of a centralized MPC scheme grows significantly, which may prohibit certain online MPC applications with a large number of decision variables.

In the context of distributed MPC design, several distributed MPC schemes have been proposed in the literature that deal with the coordination of separate MPC controllers that communicate in order to obtain optimal input trajectories in a distributed manner; see [8], [9] for reviews of results in this area. In [10], the problem of distributed control of dynamically coupled nonlinear systems that are subject to decoupled constraints was considered. In [11], [12], the effect of the coupling was modeled as a bounded disturbance compensated using a robust MPC formulation. In [13], it was proven that through multiple communications between distributed controllers and using system-wide control objective functions, stability of the closed-loop system can be guaranteed. In [14] distributed MPC of decoupled systems (a class of systems of relevance in the framework of multi-agents systems) was studied. In our previous work [16], a distributed MPC scheme was proposed for nonlinear processes without explicit consideration of input constraints. Generally, the computational burden of these distributed MPC schemes is smaller compared to the one of the corresponding centralized MPC schemes because of smaller optimization problems. Within process control, important recent work on the subject

Financial support from NSF, CBET-0529295 and the European Commission, INFSOICT-223866, is gratefully acknowledged.

Jinfeng Liu and Panagiotis D. Christofides are with the Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095-1592, USA. Panagiotis D. Christofides is also with the Department of Electrical Engineering, University of California, Los Angeles, CA 90095-1592, USA, jinfeng@ucla.edu, pdc@seas.ucla.edu.

David Muñoz de la Peña is with the Departmento de Ingeniería de Sistemas y Automática Universidad de Sevilla, Sevilla 41092, Spain, dmunoz@us.es.

of networked process control includes the development of a quasi-decentralized control framework for multi-unit plants that achieves the desired closed-loop objectives with minimal cross communication between the plant units [15] and the development of a two-tier control architecture for nonlinear processes with heterogeneous measurements to improve the closed-loop performance by taking advantage of additional networked asynchronous, delayed measurements and control actuators [20].

In the present work, we consider a class of nonlinear control problems with input constraints that arises when networked control systems which use new sensors and/or actuators are added to already operating control loops to improve closed-loop performance. In this case, it is desirable to design the pre-exisitng control system and the new networked control system in a way that they coordinate their actions. To address this control problem, we introduce a distributed model predictive control architecture where both the pre-exisitng control system and the networked control system are designed via Lyapunov-based model predictive control (LMPC) theory. The proposed distributed MPC scheme coordinates the actions of the two LMPCs in an efficient fashion, preserves the stability properties of the Lyapunov-based controller while satisfying input constraints and improving the closed-loop performance, and it is computationally more efficient compared to the corresponding centralized LMPC. The theoretical results are illustrated using a chemical process example.

II. Preliminaries

A. Problem formulation

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

$$\dot{x}(t) = f(x(t))+g_1(x(t))u_1(t)+g_2(x(t))u_2(t)+k(x(t))w(t) \quad (1)$$

where $x(t) \in R^{n_x}$ denotes the state vector, $u_1(t) \in R^{n_{u_1}}$ and $u_2(t) \in R^{n_{u_2}}$ are two separate sets of possible control (manipulated) inputs and $w(t) \in R^{n_w}$ denotes the vector of disturbance. The two inputs are restricted to be in two nonempty convex sets $U_1 \subseteq R^{n_{u_1}}$ and $U_2 \subseteq R^{n_{u_2}}$ which are defined as follows:

$$U_i := \{u_i \in R^{n_{u_i}} : |u_i| \leq u_i^{\max}\}^1, \ i = 1, \ 2$$

where $u_i^{\max}, \ i = 1, \ 2$ are the magnitudes of the input constraints. The disturbance vector is bounded, i.e., $w(t) \in W$ where

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

We assume that $f, \ g_1, \ g_2, \ k$ are local Lipschitz vector functions and that the origin is an equilibrium of the unforced nominal system (i.e., system (1) with $u_1 = 0, \ u_2 = 0$ and $w = 0$) which implies $f(0) = 0$. We also assume that the state x of the system is sampled synchronously and continuously and the time instants that we have measurement samplings are indicated by the time sequence $\{t_{k\geq 0}\}$ with $t_k = t_0 + k\Delta, \ k = 0, 1, \ldots$ where t_0 is the initial time and Δ is the sampling time.

[1] $|\cdot|$ denotes Euclidean norm of a vector.

B. Lyapunov-based controller

We assume that there exists a Lyapunov function $V(x)$ for the nominal closed-loop system with $u_2 = 0$, which implies the nominal system can be stabilized using only u_1. Based on this assumption, the following continuous bounded control law [17], [18], [19] (i.e., $u_1 = h(x)$) can be constructed to stabilize the nominal system asymptotically satisfying the input constraint on u_1 for suitable initial conditions:

$$h(x) = -a(x)L_{g_1}V(x) \quad (2)$$

where

$$a(x) = \begin{cases} \dfrac{L_fV+\sqrt{(L_fV)^2+\left(u_1^{\max}L_{g_1}V\right)^4}}{(L_{g_1}V)^2\left[1+\sqrt{1+\left(u_1^{max}L_{g_1}V\right)^2}\right]}, & L_{g_1}V \neq 0 \\ 0 \hfill , & L_{g_1}V = 0 \end{cases}$$

with $L_fV = \frac{\partial V}{\partial x}|_x f(x)$ and $L_{g_1}V = \frac{\partial V}{\partial x}|_x g_1(x)$ being the Lie derivatives of the scalar function V with respect to the vector fields f and g_1 respectively.

We denote $\Omega_\rho \subseteq D \subseteq R^{n_x}$ as the stability region of the closed-loop nominal system ($w = 0$) of system (1) under the control $u_1 = h(x)$ and $u_2 = 0$ where D is an open neighborhood of the origin. Using converse Lyapunov theorems, it also implies that there exist class $\mathcal{K}^2$ functions $\alpha_i(\cdot), \ i = 1, 2, 3, 4$ that satisfy the following inequalities

$$\begin{array}{c} \alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|) \\ \frac{V(x)}{\partial x}|_x(f(x) + g_1(x)h(x)) \leq -\alpha_3(|x|) \\ |\frac{\partial V(x)}{\partial x}| \leq \alpha_4(|x|) \end{array} \quad (3)$$

for all $x \in \Omega_\rho$ and $w \in W$.

By continuity and the local Lipschitz property assumed for f, there exists a positive constant M such that

$$|f(x)| \leq M \quad (4)$$

for all $x \in \Omega_\rho$. In addition, by the continuous differentiable property of the Lyapunov function V and the Lipschitz property of $f, \ g_1, \ g_2$ and k, there exist positive constants $L_x, \ L_{u_1}, \ L_{u_2}$ and L_w such that

$$\begin{array}{c} |\frac{\partial V}{\partial x}|_x f(x) - \frac{\partial V}{\partial x}|_{x'} f(x')| \leq L_x|x - x'| \\ |\frac{\partial V}{\partial x}|_x g_1(x) - \frac{\partial V}{\partial x}|_{x'} g_1(x')| \leq L_{u_1}|x - x'| \\ |\frac{\partial V}{\partial x}|_x g_2(x) - \frac{\partial V}{\partial x}|_{x'} g_2(x')| \leq L_{u_2}|x - x'| \\ |\frac{\partial V}{\partial x}|_x k(x)| \leq L_w \end{array} \quad (5)$$

for all $x, x' \in \Omega_\rho$. These constants will be used in the proof of Theorem 1 in section III-B.

Remark 1: The assumption that there exists a controller $u_1 = h(x)$ which can stabilize the closed-loop system with $u_2 = 0$ implies that, in principle, it is not necessary to use the extra input u_2 in order to achieve closed-loop stability. However, one of the main objectives of the proposed distributed MPC scheme is to profit from the extra control effort to improve the closed-loop performance while maintaining the stability properties achieved by only implementing u_1.

[2] Class $\mathcal{K}$ functions are strictly increasing functions of their argument and satisfy $\alpha(0) = 0$.

C. Centralized LMPC

To take advantage of both sets of manipulated inputs u_1 and u_2, a possible approach is to design a centralized MPC. In order to guarantee robust stability of the closed-loop system, the MPC must include a set of stability constraints. To do this, we propose to use the LMPC scheme proposed in [21] which guarantees practical stability of the closed-loop system, allows for an explicit characterization of the stability region. LMPC is based on uniting receding horizon control with Lyapunov functions and computes the manipulated input trajectory solving a finite horizon constrained optimal control problem. The LMPC controller is based on the previously designed Lyapunov-based controller h. The controller h is used to define a contractive constraint for the LMPC scheme which guarantees that the LMPC inherits the stability and robustness properties of the Lyapunov-based controller.

The LMPC scheme introduced in [21] is based on the following optimization problem

$$\min_{u_{c1},u_{c2}\in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T Q_c \tilde{x} + u_{c1}^T R_{c1} u_{c1} + u_{c2}^T R_{c2} u_{c2}] d\tau \quad (6a)$$

$$\dot{\tilde{x}}(\tau) = f(\tilde{x}) + g_1(\tilde{x}) u_{c1}(\tau) + g_2(\tilde{x}) u_{i2}(\tau) \quad (6b)$$

$$\tilde{x}(0) = x(t_k) \quad (6c)$$

$$u_{c1}(\tau) \in U_1 \quad (6d)$$

$$u_{c2}(\tau) \in U_2 \quad (6e)$$

$$L_{g_1}V(x(t_k))u_{c1}(0) + L_{g_2}V(x(t_k))u_{c2}(0) \leq L_{g_1}V(x(t_k))h(x(t_k)) \quad (6f)$$

where $L_{g_1}V(x(t_k)) = \frac{\partial V}{\partial x}|_{x(t_k)} g_1(x(t_k))$ and $L_{g_2}V(x(t_k)) = \frac{\partial V}{\partial x}|_{x(t_k)} g_2(x(t_k))$ are the Lie derivatives of V with respect to the vector fields g_1 and g_2 respectively, $S(\Delta)$ is the family of piece-wise constant functions with sampling period Δ, Q_c, R_{c1} and R_{c2} are positive definite weight matrices that define the cost, $x(t_k)$ is the state measurement obtained at t_k, $\tilde{x}$ is the predicted trajectory of the nominal system for the input trajectory computed by the LMPC, N is the prediction horizon and V is the Lyapunov function corresponding to the controller $h(x)$.

The optimal solution to this optimization problem is denoted by $u_{c1}^*(\tau|t_k)$ and $u_{c2}^*(\tau|t_k)$. The LMPC controller is implemented with a receding horizon scheme; that is, at each sampling time t_k, the new state $x(t_k)$ is received from the sensors, the optimization problem (6) is solved, and $u_{c1}^*(0|t_k)$ and $u_{c2}^*(0|t_k)$ are applied to the closed-loop system for $t \in [t_k, t_{k+1})$. In what follows, we refer to this controller as the centralized LMPC. The manipulated inputs of the closed-loop system under the above centralized LMPC are defined as follows

$$\begin{aligned} u_1(t) &= u_{c1}^*(0|t_k),\ \forall t \in [t_k, t_{k+1}) \\ u_2(t) &= u_{c2}^*(0|t_k),\ \forall t \in [t_k, t_{k+1}). \end{aligned} \quad (7)$$

III. Distributed LMPC

A. Distributed LMPC formulations

For a nonlinear MPC scheme, the computational complexity is a very important issue. It is well known that the computational complexity of a nonlinear MPC optimization problem grows significantly with the increase of the iteration times and the number of optimization (decision) variables. High computational complexity may prohibit the application of an MPC scheme because of the real-time requirements of control systems. The computation delays introduced by an MPC to the controller-actuator link may deteriorate the closed-loop performance significantly. Using distributed MPC schemes instead of centralized MPC schemes is a possible approach to handle the high computational complexity problem of centralized MPC problems.

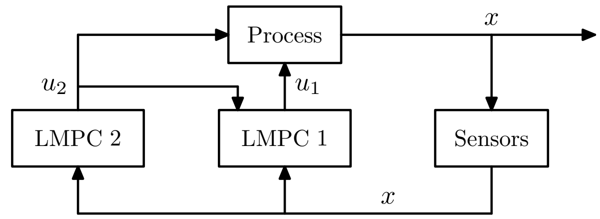

Fig. 1. Distributed LMPC control system.

The main objective of the proposed distributed LMPC scheme is to reduce the computational burden in the evaluation of the optimal manipulated inputs u_1 and u_2, while satisfying the input constraints and maintaining the performance of the closed-loop system at a very close level to the one attained when the centralized LMPC is used. Note that in general, the coordination of two controllers to regulate the same process is a difficult problem. In the present work, we design two separate LMPC schemes to compute u_1 and u_2, respectively. We refer to the LMPC schemes computing the trajectories of u_1 and u_2 as LMPC 1 and LMPC 2, respectively. Figure 1 shows a schematic of the proposed distributed scheme. We propose to use the following implementation strategy:

1) At each sampling instant t_k, both LMPC 1 and LMPC 2 receive the state measurement $x(t_k)$ from the sensors.
2) LMPC 2 evaluates the optimal input trajectory of u_2 based on the current state measurement and sends the optimal input trajectory to the corresponding actuators and to LMPC 1.
3) Once LMPC 1 receives the optimal input trajectory of u_2 from LMPC 2, it evaluates the optimal trajectory of u_1 based on the current state measurement and the optimal trajectory of u_2 decided by LMPC 2.
4) LMPC 1 sends the optimal input trajectory of u_1 to the corresponding actuators.

First we define the LMPC 2 optimization problem. This optimization problem depends on the latest state measurement $x(t_k)$, however, LMPC 2 does not have any information about the value that u_1 will take. In order to make a decision, LMPC 2 must assume a trajectory for u_1 along the prediction horizon. To this end, the Lyapunov-based controller $u_1 = h(x)$ is used. In order to inherit the stability properties of this controller, u_2 must satisfy a contractive constraint that guarantees a given minimum decrease rate of the Lyapunov function V. The LMPC 2 controller is based on the following optimization problem:

$$\min_{u_{i2}\in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T Q_c \tilde{x} + u_{i1}^T R_{c1} u_{i1} + u_{i2}^T R_{c2} u_{i2}] d\tau \quad (8a)$$

$$\dot{\tilde{x}}(\tau) = f(\tilde{x}) + g_1(\tilde{x}) u_{i1}(\tau) + g_2(\tilde{x}) u_{i2}(\tau) \quad (8b)$$

$$u_{i1}(\tau) = h(\tilde{x}(j\Delta)), \forall\, \tau \in [j\Delta, (j+1)\Delta] \quad (8c)$$

$$\tilde{x}(0) = x(t_k) \quad (8d)$$

$$u_{i2}(\tau) \in U_2 \quad (8e)$$

$$L_{g_2} V(x(t_k)) u_{i2}(0) \leq 0 \quad (8f)$$

where $\tilde{x}$ is the predicted trajectory of the nominal system with u_2 being the input trajectory computed by the LMPC and u_1 being the Lyapunov-based bounded control law $h(x)$ applied in a sample and hold fashion and $j = 0, 1, \cdots, N-1$.

The optimal solution to this optimization problem is denoted by $u_{i2}^*(\tau|t_k)$. This information is sent to LMPC 1. Constraint (8c) defines the manipulated input u_1 which is based on the bounded Lyapunov-based control law (2), constraint (8e) defines the constraint on the manipulated input u_2, constraint (8f) guarantees that the value of the time derivative of the Lyapunov function, $\dot{V}$, at the initial evaluation time, if $u_1 = h(x)$ and $u_2 = u_{i2}^*(0|t_k)$ are applied, is lower than or equal to the value obtained when $u_1 = h(x)$ and $u_2 = 0$ are applied.

The LMPC 1 optimization problem depends on the latest state measurement $x(t_k)$, and the decision taken by LMPC 2, $u_{i2}^*(\tau|t_k)$. This allows the LMPC 1 to compute an input u_1 such that the closed-loop performance is optimized, while guaranteeing that the stability properties of the Lyapunov-based controller are preserved. Specifically, the LMPC 1 controller is based on the following optimization problem:

$$\min_{u_{i1}\in S(\Delta)} \int_0^{N\Delta} [\tilde{x}^T Q_c \tilde{x} + u_{i1}^T R_{c1} u_{i1} + u_{i2}^{*T} R_{c2} u_{i2}^*] d\tau \quad (9a)$$

$$\dot{\tilde{x}} = f(\tilde{x}) + g_1(\tilde{x}) u_{i1}(\tau) + g_2(\tilde{x}) u_{i2}^*(\tau|t_k) \quad (9b)$$

$$\tilde{x}(0) = x(t_k) \quad (9c)$$

$$u_{i1} \in U_1 \quad (9d)$$

$$L_{g_1} V(x(t_k)) u_{i1}(0) \leq L_{g_1} V(x(t_k)) h(x(t_k)) \quad (9e)$$

where $\tilde{x}$ is the predicted trajectory of the nominal system with u_2 being the optimal input trajectory $u_{i2}^*(\tau|t_k)$ computed by LMPC 2 and u_1 being the input trajectory computed by LMPC 1. The optimal solution to this optimization problem is denoted by $u_{i1}^*(\tau|t_k)$. Constraint (9d) defines the constraint on the manipulated input u_1 and constraint (9e) guarantees that the value of the time derivative of the Lyapunov function, $\dot{V}$, at the initial evaluation time, if $u_1 = u_{i1}^*(0|t_k)$ and $u_2 = u_{i2}^*(0|t_k)$ are applied, is lower than or equal to the value obtained when $u_1 = h(x)$ and $u_2 = u_{i2}^*(0|t_k)$ are applied.

Once both optimization problems are solved, the manipulated inputs of the proposed distributed LMPC scheme with input constraints based on the LMPC 1 and LMPC 2 are defined as follows:

$$\begin{aligned} u_1(t) &= u_{i1}^*(0|t_k),\ \forall t \in [t_k, t_{k+1}) \\ u_2(t) &= u_{i2}^*(0|t_k),\ \forall t \in [t_k, t_{k+1}). \end{aligned} \quad (10)$$

The stability property of the distributed LMPC scheme (10) is presented in Theorem 1 in section III-B.

Remark 2: Since the computational burden of nonlinear MPC schemes is usually high, the proposed distributed LMPC scheme (10) only requires LMPC2 and LMPC 1 to "talk" once every sampling time (that is, LMPC 2 sends its optimal input trajectory to LMPC 1) to minimize the communication between the two LMPC controllers. This strategy is more robust when communication between the distributed LMPCs can be subject to disruptions.

Remark 3: Constraints (8f), (9b) and (9e) are a key element of the proposed distributed LMPC scheme (10). In general, guaranteeing closed-loop stability of a distributed control scheme is a difficult task because of the interactions between the separate controllers and can only be done under certain assumptions (see, for example, [22], [23], [24]). Constraint (9b) guarantees that LMPC 1 takes into account the effect of LMPC 2 to the applied inputs (recall that LMPC 2 is designed without taking LMPC 1 into account). Constraints (8f) and (9e) together with the hierarchical control strategy (i.e., LMPC 2 is solved first and LMPC 1 is solved second) guarantee that the value of the Lyapunov function of the closed-loop system is a decreasing sequence of time with a lower bound.

Remark 4: Note that the stability of the closed-loop system is inherited from the Lyapunov-based controller $u_1 = h(x)$. Once the contractive constraints (8f) and (9e) are satisfied, the closed-loop stability is guaranteed. The main purpose of LMPC 1 and LMPC 2 is to optimize the inputs u_1 and u_2. Thus, during the evaluation of the optimal solutions of LMPC 1 and LMPC 2 within a sampling period, we can terminate the optimization (i.e., limit the function evaluation times in the process of searching for the optimal solutions) to obtain sub-optimal input trajectories without loss of the closed-loop stability. An extreme application of this idea is when the optimization process is terminated at the beginning of every optimization process which gives the inputs: $u_1(t) = h(x(t_k))$ and $u_2(t) = 0$ for $t \in [t_k, t_{k+1})$.

Remark 5: In the proposed distributed LMPC scheme (10), LMPC 2 and LMPC 1 are evaluated in sequence, which implies that the minimum sampling time of the system should be greater than or equal to the sum of the evaluation times of LMPC 2 and LMPC 1. In order to make the two distributed LMPC optimization problems to be solved in parallel, LMPC 1 can use old input trajectories of LMPC 2, that is, at t_k, LMPC 1 uses $u_2^*(t|t_{k-1})$ to define its optimization problem. This strategy may introduce extra errors in the optimization problem, however, and may not guarantee closed-loop stability.

B. Distributed LMPC stability

In this subsection, we present the stability properties of the proposed distributed LMPC scheme (10). In order to guarantee the closed-loop stability under the proposed distributed LMPC scheme (10), we propose to follow a Lyapunov-based approach. The main idea, is that u_1 and u_2 have been computed in a way such that in the closed-loop system, the value of the Lyapunov function at time instant t_k (i.e., $V(x(t_k))$) is a decreasing sequence of values with a lower bound. Following Lyapunov arguments, this property guarantees practical stability of the closed-loop system. This is achieved due to the contractive constraints (8f) and (9e). This property is presented in Theorem 1 below.

Theorem 1: *Consider system (1) in closed-loop under the distributed LMPC scheme (10) with the bounded control law (2) designed using the Lyapunov function V. Let $\epsilon_w > 0$, $\Delta > 0$ and $\rho > \rho_s > 0$ satisfy the following constraint:*

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + L^* \le -\epsilon_w/\Delta. \quad (11)$$

with $L^* = L_x M\Delta + L_{u_1} M u_1^{\max} + L_{u_2} M u_2^{\max} + L_w\theta$. If $x(t_0) \in \Omega_\rho$ and if $\rho^* \le \rho$ where

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

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

Proof: The proof consists of two parts. We first prove that the optimization problems (8) and (9) are feasible for all states $x \in \Omega_\rho$. Then we prove that, under the proposed distributed LMPC scheme (10), the state of system (1) is ultimately bounded in a region that contains the origin.

Part 1: We prove the feasibility of LMPC 2 first, and then the feasibility of LMPC 1. All input trajectories of $u_2(\tau)$ such that $u_2(\tau) = 0$, $\forall\tau \in [0, N\Delta]$ satisfy the input constraint (8e) and the contractive constraint (8f), thus the feasibility of LMPC 2 is guaranteed. If $x(t_k) \in \Omega_\rho$, the feasibility of LMPC 1 follows. All input trajectories $u_1(\tau)$ such that $u_1(\tau) = h(x)$, $\forall\tau \in [0, N\Delta]$ are feasible solutions to the optimization problem of LMPC 1 since all such trajectories satisfy the input constraint (9d) which is guaranteed by the property of h and the contractive constraints (9e).

Part 2: From condition (3) and the constraints (8f) and (9e), if $x(t_k) \in \Omega_\rho$ it follows that

$$\begin{aligned} &\frac{\partial V(x)}{\partial x}|_{x(t_k)}(f(x(t_k)) + g_1(x(t_k))u_{i1}^*(0|t_k) \\ &\qquad\qquad + g_2(x(t_k))u_{i2}^*(0|t_k)) \\ &\le \frac{\partial V(x)}{\partial x}|_{x(t_k)}(f(x(t_k)) + g_1(x(t_k))h(x(t_k))) \\ &\le -\alpha_3(|x(t_k)|). \end{aligned} \quad (12)$$

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

$$\begin{aligned} \dot{V}(x(t)) = &\frac{\partial V}{\partial x}|_{x(t)}(f(x(t)) + g_1(x(t))u_{i1}^*(0|t_k) \\ &+ g_2(x(t))u_{i2}^*(0|t_k) + k(x(t))w(t)). \end{aligned}$$

Adding and subtracting $\frac{\partial V(x)}{\partial x}|_{x(t_k)}(f(x(t_k)) + g_1(x(t_k))u_{i1}^*(0|t_k) + g_2(x(t_k))u_{i2}^*(0|t_k))$ and taking into account (12) we obtain the following inequality

$$\begin{aligned} \dot{V}(x(t)) \le &-\alpha_3(|x(t_k)|) + \frac{\partial V}{\partial x}|_{x(t)}(f(x(t)) \\ &+ g_1(x(t))u_{i1}^*(0|t_k) + g_2(x(t))u_{i2}^*(0|t_k) \\ &+ k(x(t))w(t)) - \frac{\partial V(x)}{\partial x}|_{x(t_k)}(f(x(t_k)) \\ &+ g_1(x(t_k))u_{i1}^*(0|t_k) + g_2(x(t_k))u_{i2}^*(0|t_k)). \end{aligned} \quad (13)$$

From (3), (5) and (13), the following inequality is obtained for all $x(t_k) \in \Omega_\rho/\Omega_{\rho_s}$

$$\begin{aligned} \dot{V}(x(t)) \le &-\alpha_3(\alpha_2^{-1}(\rho_s)) + (L_x + L_{u_1}u_{i1}^*(0|t_k) \\ &+ L_{u_2}u_{i2}^*(0|t_k))|x(t) - x(t_k)| + L_w|w|. \end{aligned}$$

Taking into account (4) and the continuity of $x(t)$, the following bound can be written for all $t \in [t_k, t_{k+1}]$

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

Using this expression and the bounds on inputs u_1 and u_2, we obtain the following bound on the time derivative of the Lyapunov function for $t \in [t_k, t_{k+1}]$, for all initial states $x(t_k) \in \Omega_\rho/\Omega_{\rho_s}$

$$\begin{aligned} \dot{V}(x(t)) \le &-\alpha_3(\alpha_2^{-1}(\rho_s)) \\ &+ (L_x + L_{u_1}u_1^{\max} + L_{u_2}u_2^{\max})M + L_w\theta. \end{aligned}$$

If condition (11) is satisfied, then there exists $\epsilon_w > 0$ such that the following inequality holds for $x(t_k) \in \Omega_\rho/\Omega_{\rho_s}$

$$\dot{V}(x(t)) \le -\epsilon_w/\Delta$$

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

$$\begin{aligned} &V(x(t_{k+1})) \le V(x(t_k)) - \epsilon_w \\ &V(x(t)) \le V(x(t_k)),\ \forall t \in [t_k, t_{k+1}] \end{aligned} \quad (14)$$

for $x(t_k) \in \Omega_\rho/\Omega_{\rho_s}$. Using (14) recursively it is proved that, if $x(t_0) \in \Omega_\rho/\Omega_{\rho_s}$, the state converges to Ω_{ρ_s} in a finite number of sampling times without leaving the stability region. Once the state converges to $\Omega_{\rho_s} \subseteq \Omega_{\rho^*}$, it remains inside Ω_{ρ^*} for all times. This statement holds because of the definition of ρ^*. This proves that the closed-loop system under the proposed distributed LMPC scheme (10) is ultimately bounded in Ω_{ρ^*}.

Remark 6: Referring to Theorem 1, condition (11) guarantees that if the state of the closed-loop system at a sampling time t_k is outside the level set $V(x(t_k)) = \rho_s$ but inside the level set $V(x(t)k)) = \rho$, the derivative of the Lyapunov function of the state of the closed-loop system is negative under the proposed distributed LMPC scheme (10).

Remark 7: Referring to Theorem 1, ρ^* is the maximum value that the Lyapunov function can achieve in a time period of length Δ when $x(t_k) \in \Omega_{\rho_s}$. Ω_{ρ^*} defines an invariant set for the state $x(t)$ under sample-and-hold implementation of the inputs of the proposed distributed LMPC scheme (10).

IV. Application to a Reactor-Separator Process

The process considered in this example is a three vessel, reactor-separator process consisting of two continuously stirred tank reactors (CSTRs) and a flash tank separator (see Figure 2). A feed stream to the first CSTR F_{10} contains the reactant A which is converted into the desired product B. The desired product B can then further react into an undesired side-product C. The effluent of the first CSTR along with additional fresh feed F_{20} makes up the inlet to the second CSTR. The reactions $A \to B$ and $B \to C$ (referred to as 1 and 2, respectively) take place in the two CSTRs in series before the effluent from CSTR 2 is fed to a flash tank. The overhead vapor from the flash tank is condensed and recycled to the first CSTR and the bottom product stream is removed. A small portion of the overhead is purged before being recycled to the first CSTR. All the three vessels are assumed

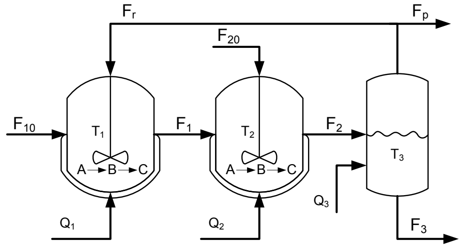

Fig. 2. Reactor-separator system with recycle.

TABLE I

NOISE PARAMETERS.

	x_{A1}	x_{B1}	T_1	x_{A2}	x_{B2}	T_2	x_{A3}	x_{B3}	T_3
σ_p	1	1	10	1	1	10	1	1	10
ϕ	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
θ_p	0.25	0.25	2.5	0.25	0.25	2.5	0.25	0.25	2.5

to have static holdup. The dynamic equations describing the behavior of the system, obtained through material and energy balances under standard modeling assumptions, are given below:

$$
\begin{aligned}
\frac{dx_{A1}}{dt} &= \frac{F_{10}}{V_1}(x_{A10}-x_{A1})+\frac{F_r}{V_1}(x_{Ar}-x_{A1})-k_1e^{\frac{-E_1}{RT_1}}x_{A1}\\
\frac{dx_{B1}}{dt} &= \frac{F_{10}}{V_1}(x_{B10}-x_{B1})+\frac{F_r}{V_1}(x_{Br}-x_{B1})+k_1e^{\frac{-E_1}{RT_1}}x_{A1}\\
&\quad -k_2e^{\frac{-E_2}{RT_1}}x_{B1}\\
\frac{dT_1}{dt} &= \frac{F_{10}}{V_1}(T_{10}-T_1)+\frac{F_r}{V_1}(T_3-T_1)+\frac{-\Delta H_1}{C_p}k_1e^{\frac{-E_1}{RT_1}}x_{A1}\\
&\quad +\frac{-\Delta H_2}{C_p}k_2e^{\frac{-E_2}{RT_1}}x_{B1}+\frac{Q_1}{\rho C_p V_1}\\
\frac{dx_{A2}}{dt} &= \frac{F_1}{V_2}(x_{A1}-x_{A2})+\frac{F_{20}}{V_2}(x_{A20}-x_{A2})-k_1e^{\frac{-E_1}{RT_2}}x_{A2}\\
\frac{dx_{B2}}{dt} &= \frac{F_1}{V_2}(x_{B1}-x_{B2})+\frac{F_{20}}{V_2}(x_{B20}-x_{B2})+k_1e^{\frac{-E_1}{RT_2}}x_{A2}\\
&\quad -k_2e^{\frac{-E_2}{RT_2}}x_{B2}\\
\frac{dT_2}{dt} &= \frac{F_1}{V_2}(T_1-T_2)+\frac{F_{20}}{V_2}(T_{20}-T_2)+\frac{-\Delta H_1}{C_p}k_1e^{\frac{-E_1}{RT_2}}x_{A2}\\
&\quad +\frac{-\Delta H_2}{C_p}k_2e^{\frac{-E_2}{RT_2}}x_{B2}+\frac{Q_2}{\rho C_p V_2}\\
\frac{dx_{A3}}{dt} &= \frac{F_2}{V_3}(x_{A2}-x_{A3})-\frac{F_r+F_p}{V_3}(x_{Ar}-x_{A3})\\
\frac{dx_{B3}}{dt} &= \frac{F_2}{V_3}(x_{B2}-x_{B3})-\frac{F_r+F_p}{V_3}(x_{Br}-x_{B3})\\
\frac{dT_3}{dt} &= \frac{F_2}{V_3}(T_2-T_3)+\frac{Q_3}{\rho C_p V_3}
\end{aligned}
\qquad (15)
$$

The model of the flash tank separator was derived under the assumption that the relative volatility for each of the species remains constant within the operating temperature range of the flash tank. This assumption allows calculating the mass fractions in the overhead based upon the mass fractions in the liquid portion of the vessel. It has also been assumed that there is a negligible amount of reaction taking place in the separator. The following algebraic equations model the composition of the overhead stream relative to

TABLE II

STEADY-STATE VALUES FOR u_{1s} AND u_{2s}.

Q_{1s}	$12.6\times10^5[KJ/hr]$	Q_{3s}	$11.88\times10^5[KJ/hr]$
Q_{2s}	$13.32\times10^5[KJ/hr]$	F_{20s}	$5.04[m^3/hr]$

TABLE III

STEADY-STATE VALUES FOR x_s.

x_{A1s}	0.605	x_{A2s}	0.605	x_{A3s}	0.346
x_{B1s}	0.386	x_{B2s}	0.386	x_{B3s}	0.630
T_{1s}	$425.9[K]$	T_{2s}	$422.6[K]$	T_{3s}	$427.3[K]$

the composition of the liquid holdup in the flash tank:

$$
\begin{aligned}
x_{Ar} &= \frac{\alpha_A x_{A3}}{\alpha_A x_{A3}+\alpha_B x_{B3}+\alpha_C x_{C3}}\\
x_{Br} &= \frac{\alpha_B x_{B3}}{\alpha_A x_{A3}+\alpha_B x_{B3}+\alpha_C x_{C3}}\\
x_{Cr} &= \frac{\alpha_C x_{C3}}{\alpha_A x_{A3}+\alpha_B x_{B3}+\alpha_C x_{C3}}
\end{aligned}
\qquad (16)
$$

Each of the tanks has an external heat input. The manipulated inputs to the system are the heat inputs, Q_1, Q_2 and Q_3, and the feed stream flow rate to vessel 2, F_{20}.

System (15) was numerically simulated using a standard Euler integration method. Process noise was added to the right-hand side of each equation in the system of ODEs of (15) to simulate disturbances/model uncertainty and it was generated as autocorrelated noise of the form $w_k = \phi w_{k-1} + \xi_k$ where $k = 0, 1, \ldots$ is the discrete time step of $0.001\ hr$, ξ_k is generated by a normally distributed random variable with standard deviation σ_p, and ϕ is the autocorrelation factor and w_k is bounded by θ_p, that is $|w_k| \leq \theta_p$. Table I contains the parameters used in generating the process noise.

We assume that the measurements of the temperatures T_1, T_2, T_3 and the measurements of mass fractions x_{A1}, x_{B1}, x_{A2}, x_{B2}, x_{A3}, x_{B3} are available synchronously and continuously at time instants $\{t_{k\geq 0}\}$ with $t_k = t_0 + k\Delta,\ k = 0, 1, \ldots$ where t_0 is the initial time and Δ is the sampling time. For the simulations carried out in this section, we pick the initial time to be $t_0 = 0$ and the sampling time to be $\Delta = 0.02\ hr = 1.2\ min$.

The control objective is to regulate the system to the steady state x_s corresponding to the operating point defined by Q_{1s}, Q_{2s}, Q_{3s} of u_{1s} and F_{20s} of u_{2s}. The steady-state values for u_{1s} and u_{2s} and the values of the steady-state are given in Table II and Table III, respectively. Taking this into account, the process model (15) belongs to the following class of nonlinear systems

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

where $x^T = [x_1\ x_2\ x_3\ x_4\ x_5\ x_6\ x_7\ x_8\ x_9] = [x_{A1} - x_{A1s}\ x_{B1} - x_{B1s}\ T_1 - T_{1s}\ x_{A2} - x_{A2s}\ x_{B2} - x_{B2s}\ T_2 - T_{2s}\ x_{A3} - x_{A3s}\ x_{B3} - x_{B3s}\ T_3 - T_{3s}]$ is the state, $u_1^T = [u_{11}\ u_{12}\ u_{13}] = [Q_1 - Q_{1s}\ Q_2 - Q_{2s}\ Q_3 - Q_{3s}]$ and $u_2 = F_{20} - F_{20s}$ are the manipulated inputs which are subject to the constraints $|u_{1i}| \leq 10^6\ KJ/hr$ $(i = 1, 2, 3)$ and $|u_2| \leq 3\ m^3/hr$, and $w = w_k$ is a time varying noise.

To illustrate the theoretical results, we consider a Lyapunov function $V(x) = x^T P x$ where P is the following weight matrix

TABLE IV

TOTAL PERFORMANCE COST.

sim.	Distr.	Centr.	sim.	Distr.	Centr.
1	65216	70868	9	79658	64342
2	70772	73112	10	65735	72819
3	57861	67723	11	62714	70951
4	62396	70914	12	76348	70547
5	60407	67109	13	49914	66869
6	83776	66637	14	89059	72431
7	61360	68897	15	78197	70257
8	47070	66818			

$$P = diag^3(5.2 \cdot 10^{12} P_v)$$

with $P_v = \begin{bmatrix} 4 & 4 & 10^{-4} & 4 & 4 & 10^{-4} & 4 & 4 & 10^{-4}) \end{bmatrix}$. The values of the weights in P have been chosen in such a way that the Lyapunov-based controller (2) stabilizes the closed-loop system and provides good closed-loop performance.

Based on the Lyapunov-based controller (2), we design the centralized and the proposed distributed LMPC schemes. In the simulations, the same parameters are used for both controllers. The prediction step is the same as the sampling time, that is $\Delta = 0.02\ hr = 1.2\ min$; the prediction horizon is chosen to be $N = 6$; and the weight matrices for the LMPC schemes are chosen as

$$Q_c = diag(10^3 Q_v)$$

with $Q_v = \begin{bmatrix} 2 & 2 & 0.0025 & 2 & 2 & 0.0025 & 2 & 2 & 0.0025 \end{bmatrix}$, and $R_{c1} = diag(\begin{bmatrix} 5 \cdot 10^{-12} & 5 \cdot 10^{-12} & 5 \cdot 10^{-12} \end{bmatrix})$ and $R_{c2} = 100$.

First, we have carried out a set of simulations to compare the distributed LMPC scheme (10) with the centralized LMPC scheme (6) with the same parameters from a performance index point of view. Table IV shows the total cost computed for 15 different closed-loop simulations under the distributed LMPC scheme (10) and the centralized LMPC scheme (6). To carry out this comparison, we have computed the total cost of each simulation with different operating conditions (different initial states and process noise) based on the index of the following form

$$\sum_{i=0}^{M} x(t_i)^T Q_c x(t_i) + u_1(t_i)^T R_{c1} u_1(t_i) + u_2(t_i)^T R_{c2} u_2(t_i)$$

where t_0 is the initial time of the simulations and $t_M = 1\ hr$ is the end of the simulations. As we can see in Table IV, the distributed LMPC scheme (10) has a cost lower than the centralized LMPC scheme (6) in 10 out of 15 simulations. This illustrates that in this example, the closed-loop performance of the distributed LMPC scheme (10) is comparable to the one of the centralized LMPC scheme (6).

Moreover, we have studied the importance of communicating optimal input trajectories of LMPC 2 to LMPC 1. Figures 3 and 4 show the results under the distributed LMPC scheme (10) with and without communication between the two controllers from the initial state

$$x(0)^T = [0.89\ 0.11\ 388.7\ 0.89\ 0.11\ 386.3\ 0.75\ 0.25\ 390.6].$$

[3] $diag(v)$ denotes a diagonal matrix with its diagonal elements being the elements of vector v.

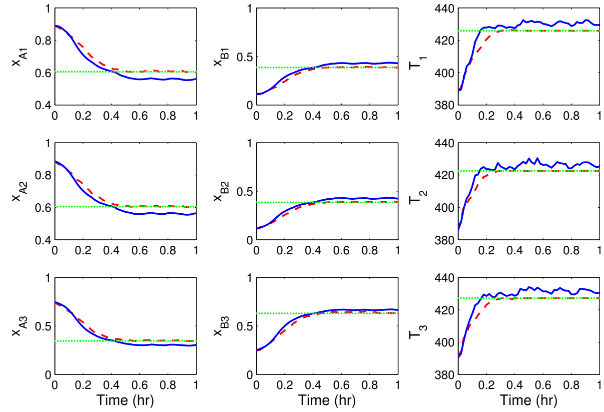

Fig. 3. State trajectories of system (15) under the distributed LMPC scheme (10) with (dashed) and without (solid) communication between the two controllers.

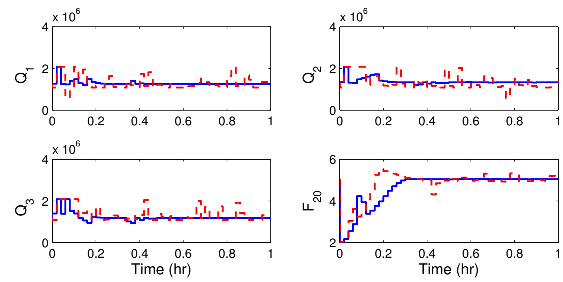

Fig. 4. Input trajectories of system (15) under the distributed LMPC scheme (10) with (dashed) and without (solid) communication between the two controllers.

When there is communication, LMPC 2 sends its optimal input trajectories to LMPC 1 as stated in the proposed implementation strategy in section III-A; when there is no communication, LMPC 2 does not send its optimal input trajectory to LMPC 1 at a sampling time. In order to make LMPC 1 work in the case of no communication, we use the nominal value of u_2 (i.e., $u_2 = F_{20s}$) to replace u_2^* in equation (9). From figure 3, we can see that the distributed LMPC scheme (10) without communication between the two controllers cannot stabilize the system at the required steady-state. The result is expected because when there is no communication between the two distributed controllers, they cannot coordinate their control actions and each controller views the input of the other controller as a disturbance and tries to counteract for it.

Furthermore, we have carried out a set of simulations to compare the computation time needed to evaluate the distributed LMPC scheme (10) with that of the centralized LMPC. The simulations have been carried out using Matlab in a Pentium 3.20 GHz. The optimization has been solved using the built-in function `fmincom` of Matlab. To solve the ODE model (15), an Euler method with a fixed integration time of $0.001\ hr$ has been implemented in a mex DLL using C programming language. For 50 evaluations, the mean time to solve the centralized LMPC is $9.40\ s$; the mean times to solve LMPC 1 and LMPC 2 are $3.19\ s$ and $4.53\ s$, respectively. From this set of simulations, we see that the computation time needed to solve the centralized LMPC is larger than the sum of the values for LMPC 1 and LMPC 2

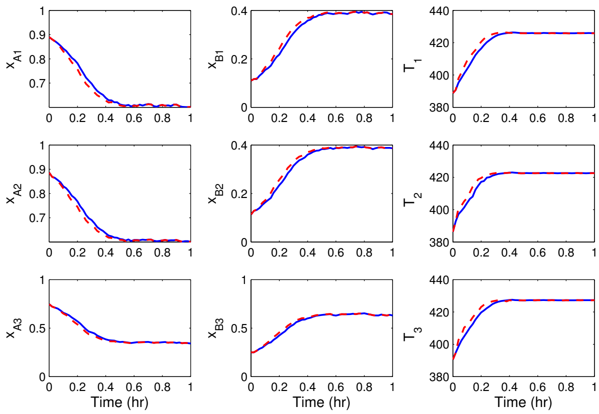

Fig. 5. State trajectories of system (15) under the distributed LMPC scheme (10) with limited (solid lines) and unconstrained (dashed lines) evaluation time.

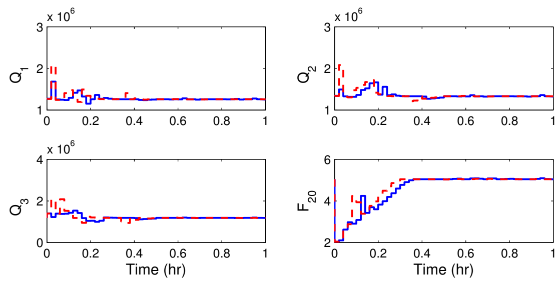

Fig. 6. Input trajectories of system (15) under the distributed LMPC scheme (10) with limited (solid lines) and unconstrained (dashed lines) evaluation time.

even though the closed-loop performance in terms of the total performance cost is comparable to the one of the distributed LMPC scheme (10). This is because the centralized LMPC has to optimize both the inputs u_1 and u_2 in one optimization problem and the distributed LMPC has to solve two smaller (in terms of decision variables) optimization problems.

Following Remark 4, we have also carried out a set of simulations to illustrate that the optimization processes of LMPC 1 and LMPC 2 can be terminated at any time to get sub-optimal solutions without loss of the closed-loop stability. In this set of simulations, we assume that the allowable evaluation times of LMPC 1 and LMPC 2 at each sampling time are 1 s and 2 s, and we terminate the optimization processes of LMPC 1 and LMPC 2 when they have been carried out for 1 s and 2 s, respectively. The closed-loop state and input trajectories under the distributed LMPC scheme (10) with limited and unconstrained computation time are shown in figures 5 and 6. From figure 5, we see that the distributed LMPC scheme (10) with limited evaluation time can stabilize the closed-loop system but the state responses are slower, leading to a higher cost (57778) compared with the one (47117) obtained under the distributed LMPC scheme (10) with unconstrained computation time.

References

[1] E. B. Ydstie, "New vistas for process control: Integrating physics and communication networks," *AIChE Journal*, vol. 48, pp. 422–426, 2002.

[2] J. F. Davis, "*Report from NSF Workshop on Cyberinfrastructure in Chemical and Biological Systems: Impact and Directions,* (see http://www.oit.ucla.edu/nsfci/NSFCIFullReport.pdf for the pdf file of this report)," 2007.

[3] P. Neumann, "Communication in industrial automation - what is going on?" *Control Engineering Practice*, vol. 15, pp. 1332–1347, 2007.

[4] P. D. Christofides, J. F. Davis, N. H. El-Farra, D. Clark, K. R. D. Harris, and J. N. Gipson, "Smart plant operations: Vision, progress and challenges," *AIChE Journal,* Perspective article, vol. 53, pp. 2734–2741, 2007.

[5] J. F. Davis and T. F. Edgar, "*Report from NSF Roadmap Development Workshop on Zero-Incident, Zero-Emission Smart Manufacturing,* (see http://www.oit.ucla.edu/nsf-evo-2008/program/SPM-Workshop-Report.pdf for the pdf file of this report)," 2008.

[6] E. F. Camacho and C. Bordóns, *Model Predictive Control, 2nd Edition*. Springer-Verlag, 2004.

[7] J. M. Maciejowski, *Predictive Control with Constraints*. Prentice Hall, 2002.

[8] E. Camponogara, D. Jia, B. H. Krogh, and S. Tulukdar, "Distributed model predictive control," *IEEE Control System Magazine*, vol. 22, pp. 44–52, 2002.

[9] J. B. Rawlings and B. T. Stewart, "Coordinating multiple optimization-based controllers: New opportunities and challenges," *J. Proc. Contr.*, vol. 18, pp. 839–845, 2008.

[10] W. B. Dunbar, "Distributed receding horizon control of dynamically coupled nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 52, pp. 1249–1263, 2007.

[11] A. Richards and J. P. How, "Robust distributed model predictive control," *International Journal of Control*, vol. 80, pp. 1517–1531, 2007.

[12] D. Jia and B. Krogh, "Min-max feedback model predictive control for distributed control with communication," in *Proceedings of the American Control Conference*, Anchorage, 2002, pp. 4507–4512.

[13] A. N. Venkat, J. B. Rawlings, and S. J. Wright, "Stability and optimality of distributed model predictive control," in *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference ECC 2005*, Seville, Spain, 2005, pp. 6680–6685.

[14] T. Keviczky, F. Borrelli, and G. J. Balas, "Decentralized receding horizon control for large scale dynamically decoupled systems," *Automatica*, vol. 42, pp. 2105–2115, 2006.

[15] Y. Sun and N. H. El-Farra, "Quasi-decentralized model-based networked control of process systems," *Computers and Chemical Engineering*, vol. 32, pp. 2016–2029, 2008.

[16] J. Liu, D. Muñoz de la Peña and P. D. Christofides, "Distributed control system design using Lyapunov-based model predictive control," In *Proc. Int. Workshop Assessment and Future Directions of Nonlinear Model Predictive Control*, 12 pages, 2008.

[17] Y. Lin and E. D. Sontag, "A universal formula for stabilization with bounded controls," *Syst. & Contr. Lett.*, vol. 16, pp. 393–397, 1991.

[18] N. H. El-Farra and P. D. Christofides, "Integrating robustness, optimality, and constraints in control of nonlinear processes," *Chem. Eng. Sci.*, vol. 56, pp. 1841–1868, 2001.

[19] P. D. Christofides and N. H. El-Farra, *Control of nonlinear and hybrid process systems: Designs for uncertainty, constraints and time-delays.* Berlin, German: Springer-Verlag, 2005.

[20] J. Liu, D. Muñoz de la Peña, B. Ohran, P. D. Christofides, and J. F. Davis, "A two-tier architecture for networked process control," *Chemical Engineering Science*, vol. 63, pp. 5349–5409, 2008.

[21] P. Mhaskar, N. H. El-Farra, and P. D. Christofides, "Predictive control of switched nonlinear systems with scheduled mode transitions," *IEEE Transactions on Automatic Control*, vol. 50, pp. 1670–1680, 2005.

[22] J. B. Rawlings and B. T. Stewart, "Coordinating multiple optimization-based controllers: New opportunities and chanllenges," in *Procedings of 8th IFAC Symposium on Dynamics and Control of Process*, vol. 1, Cancun, Mexico, 2007, pp. 19–28.

[23] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Systems Magazine*, vol. 22, pp. 44–52, 2002.

[24] P. Mhaskar, N. H. El-Farra and P. D. Christofides, "Stabilization of nonlinear systems with state and control constraints using Lyapunov-based predictive control," *Syst. & Contr. Lett.*, vol. 55, pp. 650–659, 2006.