Journal of Process Control 21 (2011) 173–181

Contents lists available at ScienceDirect

Journal of Process Control

journal homepage: www.elsevier.com/locate/jprocont

Handling communication disruptions in distributed model predictive control

Mohsen Heidarinejad[a], Jinfeng Liu[b], David Muñoz de la Peña[c], James F. Davis[b], Panagiotis D. Christofides[a,b,*]

[a] Department of Electrical Engineering, University of California, Los Angeles, CA 90095-1592, USA
[b] Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095-1592, USA
[c] Departamento de Ingeniería de Sistemas y Automática Universidad de Sevilla, Sevilla 41092, Spain

ARTICLE INFO

Article history:
Received 16 July 2010
Received in revised form 4 November 2010
Accepted 17 November 2010

Keywords:
Distributed control
Predictive control
Networked control

ABSTRACT

In this work, we study distributed model predictive control (DMPC) of nonlinear systems subject to communication disruptions – communication channel noise and data losses – between distributed controllers. Specifically, we focus on a DMPC architecture in which one of the distributed controllers is responsible for ensuring closed-loop stability while the rest of the distributed controllers communicate and cooperate with the stabilizing controller to further improve the closed-loop performance. To handle communication disruptions, feasibility problems are incorporated in the DMPC architecture to determine if the data transmitted through the communication channel is reliable or not. Based on the results of the feasibility problems, the transmitted information is accepted or rejected by the stabilizing MPC. In order to ensure the stability of the closed-loop system under communication disruptions, each model predictive controller utilizes a stability constraint which is based on a suitable Lyapunov-based controller. The theoretical results are demonstrated through a nonlinear chemical process example.

1. Introduction

The chemical process industry is a major sector of the US and global economy. Hence, the development of optimal process control and operation methodologies for chemical processes is a research subject of considerable importance. Advanced process control stands to benefit from the emergence of networked process control and operations, with the purpose of augmentation of traditional point-to-point local control systems with additional cheap, safe and easy-to-install networked sensors and actuators. Networked control systems (NCS) can substantially improve the efficiency, flexibility, robustness and fault tolerance of an industrial control system while reducing the installation, reconfiguration and maintenance expenses at the cost of coordination and design/redesign of different control systems in the new architecture [1–3]. Recent research efforts have led to important results on the design of networked control systems (e.g., [4–7]), employing a centralized control paradigm where all manipulated inputs are evaluated by a single control system.

Model predictive control (MPC) is a natural framework for dealing with the design and coordination of distributed control systems because it can account for the influence of other control systems on the computation of the control action for a certain set of actuators. MPC takes advantage of a process model to predict the future evolution of the process at each sampling time according to the current state along a given prediction horizon. These predictions are incorporated in an optimization problem to obtain an optimal input trajectory by minimizing a meaningful performance index. To reduce the computational complexity of the optimization problem, MPC obtains the optimal input solution over the family of piecewise constant trajectories with fixed sampling time and finite prediction horizon. Once the optimization problem is solved, only the first manipulated input value is implemented, discarding the rest of the trajectory and repeating the optimization in the next sampling step (e.g., [8]). In a centralized MPC paradigm, all the manipulated inputs of a given control system are coupled in a single optimization problem to obtain the optimal input trajectory. In the case of large number of state variables and manipulated inputs for a given control system, the computational complexity of the centralized MPC may increase significantly and consequently degrade closed-loop system performance, especially in the case of employing a nonlinear model in MPC. A computationally effective approach to overcome the above mentioned drawbacks of centralized MPC is to employ distributed MPC (DMPC) in which the optimal trajectory is obtained through solving a number of distributed optimization problems with lower dimensionality compared to the centralized design.

* Corresponding author at: Department of Chemical and Biomolecular Engineering, University of California, 5531 Boelter Hall, Los Angeles, CA 90095-1592, USA. Tel.: +1 310 794 1015; fax: +1 310 206 4107.

E-mail addresses: mohsen@ee.ucla.edu (M. Heidarinejad), jinfeng@ucla.edu (J. Liu), dmunoz@us.es (D. Muñoz de la Peña), jdavis@oit.ucla.edu (J.F. Davis), pdc@seas.ucla.edu (P.D. Christofides).

0959-1524/$ – see front matter
doi:10.1016/j.jprocont.2010.11.005

In the context of DMPC designs, a number of significant efforts have been recently made in the literature; please see [9–11] for reviews of available results in this direction. Specifically, the stability of the closed-loop linear system by considering multiple communications between distributed predictive controllers and using system-wide control objective functions was guaranteed in [12]. A distributed control method for weakly coupled nonlinear systems subject to decoupled constraints was proposed in [13]. A DMPC scheme for linear systems coupled through the inputs, based on a game theoretic approach, was proposed in [14]. A robust DMPC formulation for decoupled linear systems was studied in [15]. A DMPC architecture for decoupled nonlinear systems coupled through cost functions was studied in [16]. A DMPC framework for a class of nonlinear discrete-time systems subject to no exchange of information between local controllers was proposed in [17]. Furthermore, in [19], a quasi-decentralized control framework was developed for multi-unit plants that achieves the desired closed-loop objectives with minimal cross communication between the controllers.

In our previous work [20] (see also [21]), we proposed a DMPC architecture in which distributed MPCs are designed via Lyapunov-based MPC (LMPC) to coordinate their control actions using one-directional communication. Among the distributed LMPCs, one LMPC is responsible for closed-loop stability while the rest of the LMPCs communicate and cooperate with the stabilizing LMPC to improve the closed-loop performance. In [20], the communication between the distributed controllers was assumed to be perfect which is reasonable in applications where wired network communication links are utilized. Recently, wireless networks have received significant attention [22] and could play an increasingly important role in distributed control systems. In chemical process systems [3], there is an increasing trend toward developing industrial DMPC designs where individual MPCs operate through a shared wireless/wired communication network. However, the design of network-based DMPC system has to deal with the dynamics introduced by the communication network, which may include communication disruptions such as communication channel noise, data losses, bandwidth limitations, time-varying delays, and data quantization [23] which directly affect the closed-loop stability of NCS architectures. Thus, achieving closed-loop stability subject to communication disruptions in the context of DMPC is a subject of increasing importance.

Motivated by the above, in the present work, we consider DMPC of nonlinear systems subject to communication disruptions between the distributed controllers. Specifically, we focus on the design of DMPC architectures that take explicitly into account communication channel noise and data losses between the distributed controllers. In the proposed DMPC architecture, one of the distributed controllers is responsible for ensuring closed-loop stability while the rest of the distributed controllers communicate and cooperate with the stabilizing controller to further improve the closed-loop performance. The communication between the distributed controllers is prone to communication noise and data losses. We employ a specific channel model to consider a number of realistic data transmission scenarios. In order to determine if the data transmitted through the communication channel is reliable or not, feasibility problems are incorporated in the DMPC architecture and based on the result of these feasibility problems, the transmitted information is accepted or rejected by the stabilizing MPC. In order to ensure the stability of the closed-loop system under communication disruptions, each model predictive controller utilizes a stability constraint which is based on a suitable Lyapunov-based controller. The proposed DMPC system possesses an explicit characterization of the stability region of the closed-loop system and guarantees that the closed-loop system is ultimately bounded in an invariant set which contains the origin. The theoretical results are illustrated using a nonlinear chemical process example.

2. Preliminaries

2.1. Notation

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

2.2. Problem formulation

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

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

where $x(t) \in R^{n_x}$ denotes the vector of process state variables, $u_i(t) \in R^{m_{u_i}}$, $i = 1, \ldots, m$, are m sets of control (manipulated) inputs and $w(t) \in R^{n_w}$ denotes the vector of disturbance variables which is assumed to be bounded, that is, $w(t) \in W$ where

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

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

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

where $u_i^{\max}$, $i = 1, \ldots, m$, are the magnitudes of the input constraints. We will design m distributed controllers to compute the m sets of control inputs, respectively.

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

2.3. Model of the communication channel

We consider data losses and noise in communication between the m distributed controllers. For a given input $r \in R^{m_u}$ to the communication channel, the output $\tilde{r} \in R^{m_u}$ is characterized as

$$\tilde{r} = lr + n \tag{2}$$

where l is a Bernoulli random variable with parameter α and $n \in R^{m_u}$ is a vector whose elements are white gaussian noise with zero mean and the same variance σ^2. The random variable l is used to model data losses in the communication channel. The white noise, n, is used to model channel noise, quantization error or any other error to the transmitted signal, and it is independent of the data losses in a probabilistic sense. If the receiver determines that a successful transmission is made, then $l = 1$, otherwise $l = 0$. Furthermore, in order to obtain deterministic stability results, we assume that, when a successful transmission is made, the noise, n, attached to the input signal, r, is bounded by θ (that is $|n| \le \theta$) as shown in Fig. 1. Both

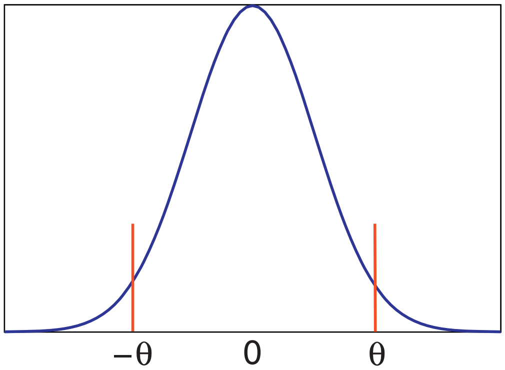

Fig. 1. Bounded communication channel noise.

assumptions are meaningful from a practical standpoint; please see the example in Section 5. We assume that the capacity of the communication channel [28] is high enough so that we can transmit data through it with a high rate.

Remark 1. Note that there are a variety of approaches to detect whether data loss has happened at the receiver side of a communication channel. One common approach is to measure the power of the received signal and compare it with a pre-configured signal transmission power level. If the power of the received signal is much smaller than the pre-configured signal transmission power level, then data loss is declared; and if the power of the received signal is close to the pre-configured signal transmission power level, then transmission is assumed to be successful.

2.4. Lyapunov-based controller

We assume that there exists a Lyapunov-based controller $h(x)$ which renders the origin of the nominal closed-loop system asymptotically stable with $u_1 = h(x)$ and $u_i = 0$ $(i = 2, \ldots, m)$, while satisfying the input constraint on u_1 for all the states x inside a given stability region. We note that this assumption is essentially equivalent to the assumption that the process is stabilizable or that the pair (A,B) in the case of linear systems is stabilizable. Using converse Lyapunov theorems [24–26], this assumption implies that there exist functions $\alpha_i(\cdot)$, $i = 1, 2, 3, 4$ of class $\mathcal{K}$ and a continuously differentiable Lyapunov function $V(x)$ for the nominal closed-loop system which is continuous and bounded in R^{n_x}, that satisfy the following inequalities:

$$\begin{aligned}
&\alpha_1(|x|) \le V(x) \le \alpha_2(|x|)\\
&\frac{\partial V(x)}{\partial x}(f(x) + g_1(x)h(x)) \le -\alpha_3(|x|)\\
&\left|\frac{\partial V(x)}{\partial x}\right| \le \alpha_4(|x|)\\
&h(x) \in U_1
\end{aligned} \tag{3}$$

for all $x \in D \subseteq R^{n_x}$ where D is an open neighborhood of the origin. We denote the region Ω_ρ as the stability region of the closed-loop system under the control inputs $u_1 = h(x)$ and $u_i = 0$ $(i = 2, \ldots, m)$. By continuity, the local Lipschitz property assumed for the functions $f(x)$, $g_i(x)$ where $i = 1, \ldots, m$ and $k(x)$ and the fact that the manipulated inputs u_i belong to the convex sets U_i, it can be concluded that there exists a positive constant M such that

$$\left| f(x(t)) + \sum_{i=1}^{m} g_i(x(t))u_i(t) + k(x(t))w(t) \right| \le M \tag{4}$$

for all $x \in \Omega_\rho$, $u_i \in U_i$ and $w \in W$. In addition, by the continuous differentiable property of the Lyapunov function V and the Lipschitz property assumed for the functions $f(x)$, $g_i(x)$ and $k(x)$, there exist positive constants L_x, L_{ui}, and L_w such that

$$\begin{aligned}
&\left|\frac{\partial V(x)}{\partial x}f(x) - \frac{\partial V(x')}{\partial x}f(x')\right| \le L_x|x - x'|\\
&\left|\frac{\partial V(x)}{\partial x}g_i(x) - \frac{\partial V(x')}{\partial x}g_i(x')\right| \le L_{ui}|x - x'|, \; i = 1, \ldots, m\\
&\left|\frac{\partial V(x)}{\partial x}k(x)\right| \le L_w
\end{aligned} \tag{5}$$

for all $x, x' \in \Omega_\rho$, $u_i \in U_i$ and $w \in W$. These constants will be employed in the proof of the stability of the closed-loop system (Theorem 1 in Section 4).

Remark 2. Note that while there are currently no general methods for constructing Lyapunov functions for general nonlinear systems, for broad classes of nonlinear systems arising in the context of chemical process control applications, quadratic Lyapunov functions are widely used and provide very good estimates of closed-loop stability regions; please see example in Section 5.

3. DMPC with communication disruptions

In our previous work [20], a DMPC architecture with flawless communication between controllers was introduced. In practice, however, there is communication disruption including channel noise and data loss between distributed controllers. The objective of this work is to propose a DMPC framework which deals with communication disruptions while maintaining closed-loop stability and improving closed-loop performance. In the sequel, we design m LMPCs to calculate the m sets of control inputs, respectively, and refer to the controller calculate u_i $(i = 1, \ldots, m)$ as LMPC i. In the proposed methodology, LMPC 1 is responsible for the stability of the closed-loop while the rest of LMPCs (i.e., LMPC 2 to LMPC m) communicate and cooperate with LMPC 1 to improve the closed-loop performance. The proposed DMPC design inherits the closed-loop stability from the Lyapunov-based controller $h(\cdot)$. A schematic diagram of the proposed DMPC design for systems subject to communication disruptions between distributed controllers is depicted in Fig. 2.

We propose to use the following implementation strategy:

1. All LMPCs receive the sensor measurements $x(t_k)$ at sampling time t_k.
2. For $i = 2, \ldots, m$

2.1. LMPC i evaluates the optimal input trajectory of u_i based on $x(t_k)$ and sends the first step input values of u_i to its corresponding actuators.
2.2. LMPC i sends the entire optimal input trajectory of u_i to LMPC 1 through a communication channel.

3. LMPC 1 solves a feasibility problem for each input trajectory it received to determine if the trajectory should be accepted or rejected.
4. LMPC 1 evaluates the future input trajectory of u_1 based on $x(t_k)$ and the results of the feasibility problems for the trajectories it received from LMPC i with $i = 2, \ldots, m$.
5. LMPC 1 sends the first step input value of u_1 to its corresponding actuators.

In the sequel, we describe the design of LMPC j $(j = 2, \ldots, m)$ and its corresponding feasibility problem and the design of LMPC 1.

Upon receiving the sensor measurement $x(t_k)$, LMPC j obtains its optimal input trajectory by solving the following optimization

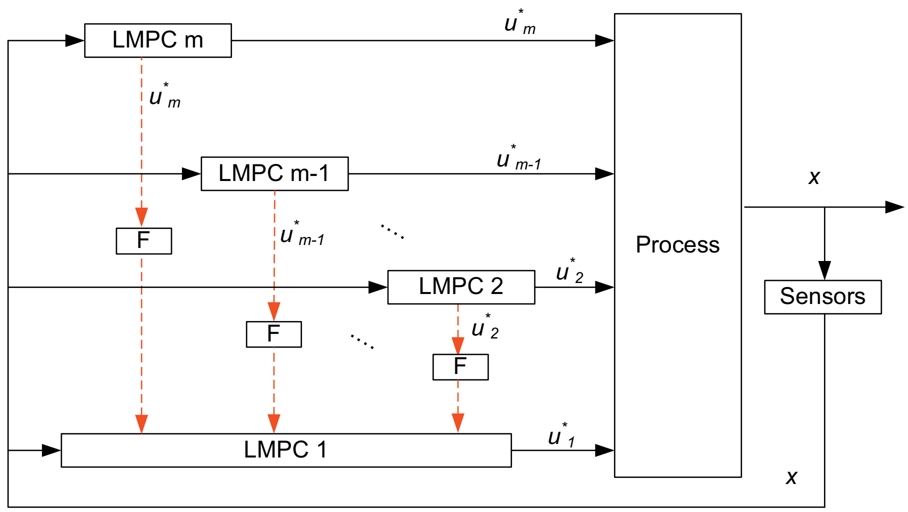

Fig. 2. Distributed LMPC control architecture (F means solving a feasibility problem).

problem:

$$\min_{u_j \in S(\Delta)} \int_0^{N\Delta} \left[\tilde{x}^j(\tau)^T Q_c \tilde{x}^j(\tau) + \sum_{i=1}^{m} u_i^T(\tau) R_{ci} u_i(\tau) \right] d\tau \quad (6a)$$

$$\dot{\tilde{x}}^j(\tau) = f(\tilde{x}^j(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}^j(\tau)) u_i(\tau) \quad (6b)$$

$$u_1(\tau) = h(\tilde{x}^j(q\Delta)), \forall \tau \in [q\Delta, (q+1)\Delta) \quad (6c)$$

$$u_i(\tau) = 0, \forall 2 \le i \le m \;\; \& \;\; i \ne j \quad (6d)$$

$$\tilde{x}^j(0) = x(t_k) \quad (6e)$$

$$u_j(\tau) \in U_j \quad (6f)$$

$$\frac{\partial V(x(t_k))}{\partial x} g_j(x(t_k)) u_j(0) \le 0 \quad (6g)$$

where $S(\Delta)$ is the family of piece-wise constant functions with sampling period Δ, Q_c and R_{ci} ($i=1,\ldots,m$) are positive definite weight matrices that define the cost, $q=0, \ldots, N-1$, $x(t_k)$ is the state measurement obtained at t_k, $\tilde{x}^j$ is the predicted trajectory of the nominal system for the input trajectory computed by the LMPC j, and N is the prediction horizon. We note that in order to simplify the notation, $\partial V(x(t_k))/\partial x$ is used to denote $\partial V(x(t))/\partial x|_{t=t_k}$. In the prediction of the future evolution of the system by LMPC j, it is assumed that LMPC 1 applies the explicit Lyapunov-based controller $h(\cdot)$ while the rest of the controllers apply zero. While this LMPC formulation intends to improve the closed-loop performance, Eq. (6g) ensures that the implemented control action contributes to further decrease of the value of the derivative of the Lyapunov function.

Let $u_j^*(\tau|t_k)$ denote the optimal solution of the optimization problem of Eq. (6). LMPC j sends the first step value of $u_j^*(\tau|t_k)$ to its actuators and transmits the whole optimal trajectory through the communication channel to LMPC 1. LMPC 1 receives a corrupted version of $u_j^*(\tau|t_k)$ which can be formulated as:

$$\tilde{u}_j(\tau|t_k) = l u_j^*(\tau|t_k) + n$$

If data losses occur during the transmission of the control input trajectory from LMPC j to LMPC 1, LMPC 1 assumes that LMPC j applies a zero input (i.e., $u_j=0$). Note that in this work, we do not consider explicitly the step of determining whether data losses occur or not in the transmission of input trajectories. Please see Remark 1 on approaches of determining transmission data losses.

When a transmission of the input trajectory $u_j^*(\tau|t_k)$ is successful, LMPC 1 receives $\tilde{u}_j(\tau|t_k)$ which is a noise-corrupted version of $u_j^*(\tau|t_k)$. To determine the reliability of the received information, LMPC 1 solves a feasibility problem. Based on the result of the feasibility problem, LMPC 1 determines if the received information should be accepted or rejected. The feasibility problem for the information received from LMPC j is as follows:

$$\text{find } z \in S(\Delta) \quad \tilde{u}_j(\tau|t_k) - \theta \le z(\tau) \le \tilde{u}_j(\tau|t_k) + \theta \quad (7a)$$

$$z(\tau) \in U_j \quad (7b)$$

$$\frac{\partial V(x(t_k))}{\partial x} g_j(x(t_k)) z(0) > 0 \quad (7c)$$

According to the bounded noise value and the received signal from the communication channel, LMPC 1 considers all the possibilities of noise effect on the optimal trajectory of LMPC j (i.e., constraint of Eq. (7a)) and checks whether in these cases LMPC j satisfies the constraint of Eq. (7c). Note that when the optimization problem of Eq. (7) is not feasible, it is guaranteed that the original signal $u_j^*(\tau|t_k)$ after transmission through the channel still satisfies the stability constraint of Eq. (6g). The feasibility of this problem is used to test whether there exists any possible value of the noise that could (due to corruption) end up making the implemented control action cause an increase in the Lyapunov function derivative, i.e., that $(\partial V(x(t_k))/\partial x) g_j(x(t_k)) u_j(0) > 0$. If the problem is infeasible, it is guaranteed that the noise cannot make the control action destabilizing, and hence, the control action is accepted. On the other hand, if the problem is feasible, it opens up the possibility of the noise rendering the control action destabilizing, and hence, it is discarded. We also note that there is no requirement that θ is sufficient small, however, larger values of θ increase the range of $z(\tau)$ and influence the feasibility of the problem of Eq. (7).

If the optimization problem of Eq. (7) is not feasible, then the trajectory information received by LMPC 1 (i.e., $\tilde{u}_j(\tau|t_k)$) is used in the evaluation of LMPC 1; and if the optimization problem of Eq. (7) is feasible, then $\tilde{u}_j(\tau|t_k)$ is discarded and a zero trajectory for u_j will be used in the evaluation of LMPC 1. If we define the trajectory of u_j that is used in the evaluation of LMPC 1 as $\tilde{u}_j^*(\tau|t_k)$, then it is defined as follows:

$$\tilde{u}_j^*(\tau|t_k) = \begin{cases} \tilde{u}_j(\tau|t_k) & \text{if (7) is not feasible and there is no data loss} \\ 0 & \text{if (7) is feasible or there exists data loss} \end{cases}$$

where $0 \in R^{m_{u_j}}$. Note that when data loss in the communication channel occurs, a zero trajectory of u_j is also used in the evaluation of LMPC 1. Note also that the above strategy on the use of the corrupted communication information is just one of many possible options to handle communication disruptions in the DMPC architecture.

Employing $\tilde{u}_j^*$ where $j = 2, \ldots, m$, LMPC 1 obtains its optimal trajectory according to the following optimization problem:

$$\min_{u_1 \in S(\Delta)} \int_0^{N\Delta} \left[\tilde{x}^1(\tau)^T Q_c \tilde{x}^1(\tau) + \sum_{i=1}^{m} u_i^T(\tau) R_{ci} u_i(\tau) \right] d\tau \tag{8a}$$

$$\dot{\tilde{x}}^1(\tau) = f(\tilde{x}^1(\tau)) + \sum_{i=1}^{m} g_i(\tilde{x}^1(\tau)) u_i(\tau)$$

$$u_1(\tau) \in U_1 \tag{8b}$$

$$u_j(\tau) = \tilde{u}_j^*(\tau|t_k), j = 2, \ldots, m \tag{8c}$$

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

$$\frac{\partial V(x(t_k))}{\partial x} g_1(x(t_k)) u_1(0) \leq \frac{\partial V(x(t_k))}{\partial x} g_1(x(t_k)) h(x(t_k)) \tag{8e}$$

In this formulation, LMPC 1 takes advantage of the knowledge of $m-1$ feasibility problems (i.e., $\tilde{u}_j^*, j = 2, \ldots, m$) and the Lyapunov-based controller $h(\cdot)$ to predict the future evolution of the system $\tilde{x}^1$. Let $u_1^*(\tau|t_k)$ denote the optimal solution of the optimization problem of Eq. (6).

Based on the solutions of the m LMPC optimization problems, the manipulated inputs of the proposed DMPC design are defined as follows:

$$u_i(t) = u_i^*(t - t_k|t_k), \forall t \in [t_k, t_{k+1})\, i = 1, \ldots, m. \tag{9}$$

Remark 3. It should be mentioned that the white gaussian noise considered in this work is the accumulation of thermal effects and quantization errors. We did not consider the effects of multi-path transmission, terrain blocking, interference, etc. Further, in this work, we assume that when packet loss happens, all of the information we want to transmit is lost; however, without loss of generality, we can extend this work to the case in which data loss happens only in some packets of information following a similar methodology like [7] to deal with this issue. The interested reader may refer to [28,29] for more details on communication channel modeling.

4. DMPC stability

As it will be proved in Theorem 1 below, the proposed DMPC framework takes advantage of the constraints of Eqs. (6g) and (8e) to compute the optimal trajectories $u_1, \ldots, u_m$ such that the Lyapunov function value $V(x(t_k))$ is a decreasing sequence with a lower bound and achieves the closed-loop stability of the system.

Theorem 1. *Consider the system of* Eq. (1) *in closed-loop under the DMPC design of* Eqs. (6)–(9) *based on a controller* $u_1 = h(x)$ *that satisfies the conditions of* Eq. (3). *Let* $\varepsilon_w > 0$, $\Delta > 0$ *and* $\rho > \rho_s > 0$ *satisfy the following constraint:*

$$-\alpha_3(\alpha_2^{-1}(\rho_s)) + \left(L_x + \sum_{i=1}^{m} L_{ui} u_i^{\max} \right) M\Delta + L_w \theta_w \leq \frac{-\varepsilon_w}{\Delta}. \tag{10}$$

If $x(t_0) \in \Omega_\rho$ *and if* $\rho^* \leq \rho$ *where* $\rho^* = \max\{V(x(t+\Delta)) : V(x(t)) \leq \rho_s\}$, *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 of Eqs. (6) and (8) are feasible for all states $x \in \Omega_\rho$. Subsequently, we prove that, under the DMPC design of Eqs. (6)–(9), the state of the system of Eq. (1) is ultimately bounded in a region that contains the origin.

Part 1: First, we consider the feasibility of LMPC j ($j = 2, \ldots, m$) and then focus on the feasibility of LMPC 1. All input trajectories of $u_j(\tau)$ such that $u_j(\tau) = 0, \forall \tau \in [0, N\Delta)$ satisfy all the constraints (including the input constraint of Eq. (6f) and the constraint of Eq. (6g)) of LMPC j, thus the feasibility of LMPC j is obtained. The feasibility of LMPC 1 follows because all input trajectories $u_1(\tau)$ such that $u_1(\tau) = h(x(t_k)), \forall \tau \in [0, \Delta)$ and $u_1(\tau) = 0, \forall \tau \in [\Delta, N\Delta)$ are feasible solutions to the optimization problem of LMPC 1 since all such trajectories satisfy the input constraint of Eq. (8b) and the constraint of Eq. (8e); this is guaranteed by the assumed property of the Lyapunov-based controller $h(\cdot)$.

Part 2: Considering the inequalities of Eq. (3), addition of inequalities of Eqs. (6g) and (8e) for $j = 2, \ldots, m$ implies that if $x(t_k) \in \Omega_\rho$, the following inequality holds:

$$\frac{\partial V(x(t_k))}{\partial x}\left(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k)) u_i^*(0|t_k)\right) \leq \frac{\partial V(x(t_k))}{\partial x}\left(f(x(t_k)) + g_1(x(t_k)) h(x(t_k))\right) \leq -\alpha_3(|x(t_k)|). \tag{11}$$

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

$$\dot{V}(x(t)) = \frac{\partial V(x)}{\partial x}\left(f(x(t)) + \sum_{i=1}^{m} g_i(x(t)) u_i^*(0|t_k) + k(x(t)) w(t)\right). \tag{12}$$

Adding and subtracting $(\partial V(x(t_k))/\partial x)(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k)) u_i^*(0|t_k))$ to the right-hand-side of Eq. (12) and taking Eq. (11) into account, we obtain the following inequality:

$$\dot{V}(x(t)) \leq -\alpha_3(|x(t_k)|) + \frac{\partial V(x)}{\partial x}\left(f(x(t)) + \sum_{i=1}^{m} g_i(x(t)) u_i^*(0|t_k) + k(x(t)) w(t)\right) - \frac{\partial V(x(t_k))}{\partial x}\left(f(x(t_k)) + \sum_{i=1}^{m} g_i(x(t_k)) u_i^*(0|t_k)\right). \tag{13}$$

From Eq. (5) and the inequality of Eq. (13), the following inequality is obtained for all $x(t_k) \in \Omega_\rho \backslash \Omega_{\rho_s}$:

$$\dot{V}(x(t)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + L_w |w(t)| + \left(L_x + \sum_{i=1}^{m} L_{ui} u_i^*(0|t_k) \right) |x(t) - x(t_k)|. \tag{14}$$

Taking into account Eq. (4) and the continuity of $x(t)$, the following bound can be written for all $t \in [t_k, t_{k+1})$, $|x(t) - x(t_k)| \leq M\Delta$. Using this expression, 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 $x(t_k) \in \Omega_\rho \backslash \Omega_{\rho_s}$:

$$\dot{V}(x(t)) \leq -\alpha_3(\alpha_2^{-1}(\rho_s)) + \left(L_x + \sum_{i=1}^{m} L_{ui} u_i^{\max} \right) M\Delta + L_w \theta_w.$$

If the condition of Eq. (10) is satisfied, then there exists $\varepsilon_w > 0$ such that the following inequality holds for $x(t_k) \in \Omega_\rho \backslash \Omega_{\rho_s}$:

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

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

$$\begin{aligned} V(x(t_{k+1})) &\leq V(x(t_k)) - \varepsilon_w \\ V(x(t)) &\leq V(x(t_k)), \forall t \in [t_k, t_{k+1}) \end{aligned} \tag{15}$$

for all $x(t_k) \in \Omega_\rho \backslash \Omega_{\rho_s}$. Using Eq. (15) recursively, it is proved that, if $x(t_0) \in \Omega_\rho \backslash \Omega_{\rho_s}$, 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 distributed LMPC design is ultimately bounded in Ω_{ρ^*}.

Remark 4. The condition of Eq. (10) 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 design.

Remark 5. For nonlinear systems under continuous control implementation, a sufficient condition for invariance is that the Lyapunov function is decreasing on the boundary of a set. For systems with continuous-time dynamics and sample-and-hold control implementation, this condition is not sufficient because the derivative may become positive during the sampling period and the system may leave the set before a new sample is obtained. Based on 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 control action.

Remark 6. Note that the closed-loop stability is guaranteed by the constraints of Eqs. (6g) and (8e). The use of the corrupted input trajectory information of u_j (i.e., $\tilde{u}_j$) where $j = 2,\ldots,m$ does not affect the feasibility of the optimization problems of Eqs. (6) and (8) as well as the stability of the closed-loop system; however, it does affect the closed-loop system performance. This is the reason for the introduction of the feasibility problem of Eq. (7) which is used to decide whether the corrupted information can be used to improve the closed-loop performance.

Remark 7. We have partitioned Ω_ρ into two regions ($\Omega_\rho \backslash \Omega_{\rho_s}$ and Ω_{ρ_s}). When $x(t_k) \in \Omega_\rho$, it follows that either $x(t_k) \in \Omega_\rho \backslash \Omega_{\rho_s}$ or $x(t_k) \in \Omega_{\rho_s}$. As we stated and proved in Theorem 1, according to definition of ρ^*, once the state converges to $\Omega_{\rho_s} \subseteq \Omega_{\rho^*}$, it remains inside Ω_{ρ^*} for all times. If $x(t_k) \in \Omega_\rho \backslash \Omega_{\rho_s}$ and the condition in Eq. (10) is satisfied, the state converges to Ω_{ρ_s} in a finite number of sampling times without leaving the stability region. In both cases, the state will be bounded in Ω_{ρ^*}.

Remark 8. In the present work, we deal with communication disruptions and do not address issues arising due to faults in the control actuators or in the control system (e.g., [18,27]). Also, we assume that all the distributed controllers have access to the full system state. In future work, we will address the scenario in which each controller has access only to partial state information and utilizes an observer to estimate the full system state subject to bounded process noise (disturbance). It should be mentioned that due to the effect of disturbances and model errors, the controllers should be updated at every several sampling time with full system state information in order to provide deterministic closed-loop stability properties.

5. Application to a chemical process

The process considered in this study is a three vessel, reactor-separator system consisting of two continuously stirred tank reactors (CSTRs) and a flash tank separator shown in Fig. 3 [27]. 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 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:

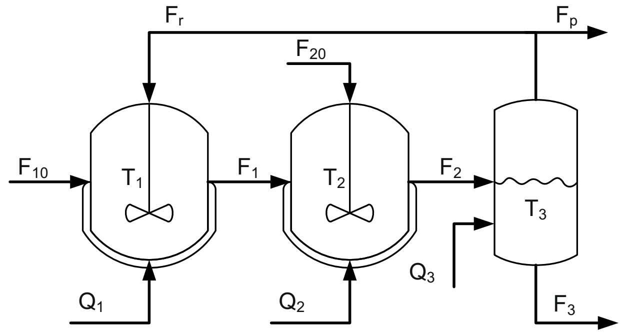

Fig. 3. Two CSTRs and a flash tank with recycle stream.

$$\frac{dT_1}{dt} = \frac{F_{10}}{V_1}(T_{10} - T_1) + \frac{F_r}{V_1}(T_3 - T_1) + \frac{-\Delta H_1}{\rho C_p} k_1 e^{-E_1/RT_1} C_{A1} + \frac{-\Delta H_2}{\rho C_p} k_2 e^{-E_2/RT_1} C_{A1} + \frac{Q_1}{\rho C_p V_1} \quad (16a)$$

$$\frac{dC_{A1}}{dt} = \frac{F_{10}}{V_1}(C_{A10} - C_{A1}) + \frac{F_r}{V_1}(C_{Ar} - C_{A1}) - k_1 e^{-E_1/RT_1} C_{A1} - k_2 e^{-E_2/RT_1} C_{A1} \quad (16b)$$

$$\frac{dC_{B1}}{dt} = \frac{-F_{10}}{V_1} C_{B1} + \frac{F_r}{V_1}(C_{Br} - C_{B1}) + k_1 e^{-E_1/RT_1} C_{A1} \quad (16c)$$

$$\frac{dC_{C1}}{dt} = \frac{-F_{10}}{V_1} C_{C1} + \frac{F_r}{V_1}(C_{Cr} - C_{C1}) + k_2 e^{-E_2/RT_1} C_{A1} \quad (16d)$$

$$\frac{dT_2}{dt} = \frac{F_1}{V_2}(T_1 - T_2) + \frac{(F_{20} + \Delta F_{20})}{V_2}(T_{20} - T_2) + \frac{-\Delta H_1}{\rho C_p} k_1 e^{-E_1/RT_2} C_{A2} + \frac{-\Delta H_2}{\rho C_p} k_2 e^{-E_2/RT_2} C_{A2} + \frac{Q_2}{\rho C_p V_2} \quad (16e)$$

$$\frac{dC_{A2}}{dt} = \frac{F_1}{V_2}(C_{A1} - C_{A2}) + \frac{(F_{20} + \Delta F_{20})}{V_2}(C_{A20} - C_{A2}) - k_1 e^{-E_1/RT_2} C_{A2} - k_2 e^{-E_2/RT_2} C_{A2} \quad (16f)$$

$$\frac{dC_{B2}}{dt} = \frac{F_1}{V_2}(C_{B1} - C_{B2}) - \frac{(F_{20} + \Delta F_{20})}{V_2} C_{B2} + k_1 e^{-E_1/RT_2} C_{A2} \quad (16g)$$

$$\frac{dC_{C2}}{dt} = \frac{F_1}{V_2}(C_{C1} - C_{C2}) - \frac{(F_{20} + \Delta F_{20})}{V_2} C_{C2} + k_2 e^{-E_2/RT_2} C_{A2} \quad (16h)$$

$$\frac{dT_3}{dt} = \frac{F_2}{V_3}(T_2 - T_3) - \frac{H_{vap} F_r}{\rho C_p V_3} + \frac{Q_3}{\rho C_p V_3} \quad (16i)$$

$$\frac{dC_{A3}}{dt} = \frac{F_2}{V_3}(C_{A2} - C_{A3}) - \frac{F_r}{V_3}(C_{Ar} - C_{A3}) \quad (16j)$$

$$\frac{dC_{B3}}{dt} = \frac{F_2}{V_3}(C_{B2} - C_{B3}) - \frac{F_r}{V_3}(C_{Br} - C_{B3}) \quad (16k)$$

Table 1
Process variables.

C_{A1}, C_{A2}, C_{A3}	Concentrations of A in vessels 1, 2, 3
C_{B1}, C_{B2}, C_{B3}	Concentrations of B in vessels 1, 2, 3
C_{C1}, C_{C2}, C_{C3}	Concentrations of C in vessels 1, 2, 3
C_{Ar}, C_{Br}, C_{Cr}	Concentrations of A, B, C in the recycle
T_1, T_2, T_3	Temperatures in vessels 1, 2, 3
T_{10}, T_{20}	Feed stream temperatures to vessels 1, 2
F_1, F_2, F_3	Effluent flow rates from vessels 1, 2, 3
F_{10}, F_{20}	Feed stream flow rates to vessels 1, 2
C_{A10}, C_{A20}	Concentrations of A in the feed stream to vessels 1, 2
F_r	Recycle flow rate
V_1, V_2, V_3	Volumes of vessels 1, 2, 3
u_1, u_2, u_3, u_4	Manipulated inputs
E_1, E_2	Activation energy for reactions 1, 2
k_1, k_2	Pre-exponential values for reactions 1, 2
$\Delta H_1, \Delta H_2$	Heats of reaction for reactions 1, 2
H_{vap}	Heat of vaporization
$\alpha_A, \alpha_B, \alpha_C, \alpha_D$	Relative volatilities of A, B, C, D
MW_A, MW_B, MW_C	Molecular weights of A, B, and C
Q_1, Q_2, Q_3	Heat inputs into vessels 1, 2, 3
C_p, R, ρ	Heat capacity, gas constant and solution density

Table 2
Parameter values.

$T_{10} = 300$, $T_{20} = 300$	K
$F_{10} = 5$, $F_{20} = 5$, $F_r = 1.9$	m^3/h
$C_{A10} = 4$, $C_{A20} = 3$	$kmol/m^3$
$V_1 = 1.0$, $V_2 = 0.5$, $V_3 = 1.0$	m^3
$E_1 = 5E4$, $E_2 = 5.5E4$	kJ/kmol
$k_1 = 3E6$, $k_2 = 3E6$	1/h
$\Delta H_1 = -5E4$, $\Delta H_2 = -5.3E4$	kJ/kmol
$H_{\text{vap}} = 5$	kJ/mol
$C_p = 0.231$	kJ/kg K
$R = 8.314$	kJ/kmol K
$\rho = 1000$	kg/m^3
$\alpha_A = 2$, $\alpha_B = 1$, $\alpha_C = 1.5$, $\alpha_D = 3$	unitless
$MW_A = 50$, $MW_B = 50$, $MW_C = 50$	kg/kmol

$$\frac{dC_{C3}}{dt} = \frac{F_2}{V_3}(C_{C2} - C_{C3}) - \frac{F_r}{V_3}(C_{Cr} - C_{C3}) \tag{16l}$$

Each of the tanks has an external heat input/removal actuator. The model of the flash tank separator is 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 the composition of the liquid holdup in the flash tank:

$$C_{Ar} = \frac{\alpha_A C_{A3}}{K}, \quad C_{Br} = \frac{\alpha_B C_{B3}}{K}, \quad C_{Cr} = \frac{\alpha_C C_{C3}}{K},$$
$$K = \alpha_A C_{A3}\frac{MW_A}{\rho} + \alpha_B C_{B3}\frac{MW_B}{\rho} + \alpha_C C_{C3}\frac{MW_C}{\rho} + \alpha_D x_D \rho \tag{17}$$

where x_D is the mass fraction of the solvent in the flash tank liquid holdup and is found from a mass balance. The definitions for the variables used in Eqs. (16) and (17) can be found in Table 1, with the parameter values given in Table 2.

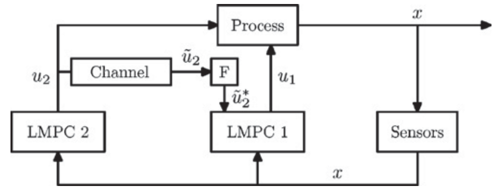

Fig. 4. Distributed LMPC control architecture for chemical process example (F means solving a feasibility problem).

The system of Eqs. (16) and (17) is numerically simulated using a standard Euler integration method. Process noise was added to the right-hand side of each equation in the process of Eq. (16) to simulate disturbances/model uncertainty and it is 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 h, 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 3 contains the parameters used in generating the process noise.

We assume that the state measurements which include the temperatures and species concentrations in the three vessels 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 $\Delta = 0.01$ h = 36 s (Tables 4 and 5).

The first set of manipulated inputs is the heat injected to or removed from the three vessels, that is $u_1 = [\,Q_1 - Q_{1s} \quad Q_2 - Q_{2s} \quad Q_3 - Q_{3s}\,]^T$; the second set of manipulated inputs is the deviated inlet flow rate to vessel 2, that is $u_2 = \Delta F_{20} = F_{20} - F_{20s}$. The open-loop system has one unstable and two stable steady states. The control objective is to regulate the system to the unstable steady-state x_s corresponding to the operating point defined by Q_{1s}, Q_{2s}, Q_{3s} and F_{20s}. The steady-state values for u_1 and u_2 are zero. Taking this control objective into account, the process model belongs to the following class of nonlinear systems: $\dot{x}(t) = f(x(t)) + g_1(x(t))u_1(t) + g_2(x(t))u_2(t) + w(t)$ where $x^T = [\,x_1 \quad x_2 \quad x_3 \quad x_4 \quad x_5 \quad x_6 \quad x_7 \quad x_8 \quad x_9 \quad x_{10} \quad x_{11} \quad x_{12}\,] = [\,T_1 - T_{1s} \quad C_{A1} - C_{A1s} \quad C_{B1} - C_{B1s} \quad C_{C1} - C_{C1s} \quad T_2 - T_{2s} \quad C_{A2} - C_{A2s} \quad C_{B2} - C_{B2s} \quad C_{C2} - C_{C2s} \quad T_3 - T_{3s} \quad C_{A3} - C_{A3s} \quad C_{B3} - C_{B3s} \quad C_{C3} - C_{C3s}\,]$ is the state, $u_1^T = [\,u_{11} \;\; u_{12} \;\; u_{13}\,] = [\,Q_1 - Q_{1s} \quad Q_2 - Q_{2s} \quad Q_3 - Q_{3s}\,]$ and $u_2 = \Delta F_{20} = F_{20} - F_{20s}$ are the manipulated inputs which are deviation variables and are subject to the constraints $|u_{1i}| \leq 10^4$ kJ/h ($i = 1,2,3$) and $|u_2| \leq 5\ m^3/h$, and w is a bounded noise.

We consider a quadratic Lyapunov function $V(x) = x^T P x$ with $P = \text{diag}([\,10 \;\; 10^4 \;\; 10^4 \;\; 10^4 \;\; 10 \;\; 10^4 \;\; 10^4 \;\; 10^4 \;\; 10 \;\; 10^4 \;\; 10^4 \;\; 10^4\,])$ and design the controller $h(x)$ as three PI controllers with proportional gains $K_{p1} = K_{p2} = K_{p3} = 8000$ and integral time constants $\tau_{I1} = \tau_{I2} = \tau_{I3} = 10$ based on the measurements of T_1, T_2 and T_3, respectively. The values of the weights in P have been chosen in a way such that the Lyapunov-based controller $h(x)$ satisfies the input constraints, stabilizes the closed-loop system and provides good closed-loop performance. Note that, in the absence of process and measurement noise, this design of $h(x)$ manipulating

Table 3
Disturbance parameters.

	σ_p	ϕ	θ_p		σ_p	ϕ	θ_p		σ_p	ϕ	θ_p
C_{A1}	0.1	0.7	0.09	C_{A2}	0.1	0.7	0.09	C_{A3}	0.1	0.7	0.09
C_{B1}	0.02	0.7	0.01	C_{B2}	0.1	0.7	0.03	C_{B3}	0.1	0.7	0.02
C_{C1}	0.02	0.7	0.01	C_{C2}	0.1	0.7	0.01	C_{C3}	0.02	0.7	0.01
T_1	10	0.7	1.17	T_2	10	0.7	1.35	T_3	10	0.7	1.35

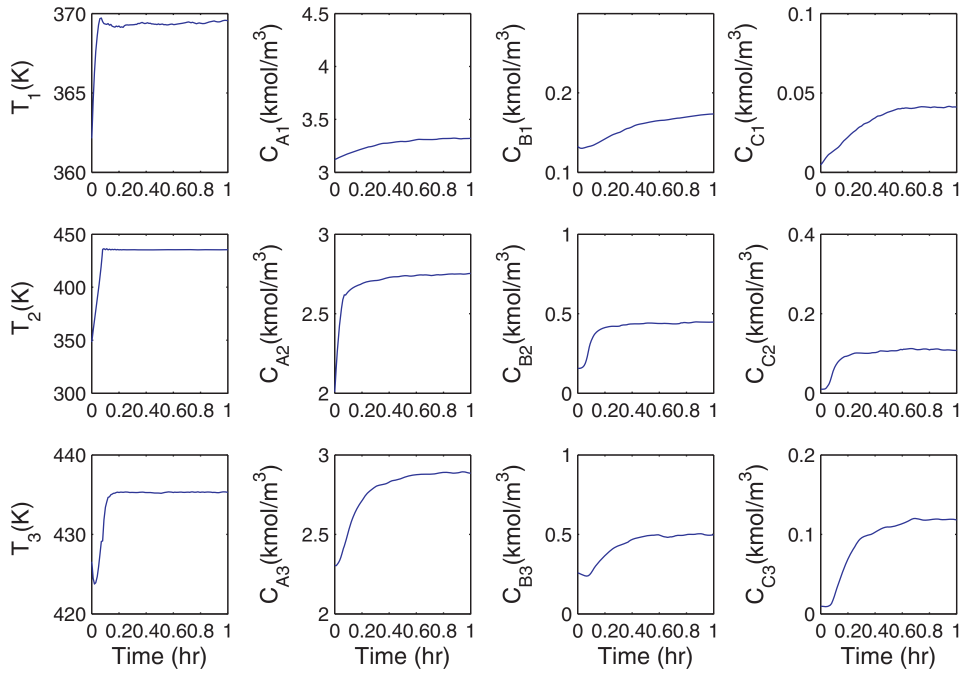

Fig. 5. State trajectories of the process under the proposed DMPC design.

Table 4
Steady-state values for Q_{1s}, Q_{2s} and Q_{3s}.

Q_{1s}	0 [kJ/h]	Q_{2s}	0 [kJ/h]	Q_{3s}	0 [kJ/h]

$u_1^T = [\,Q_1 \quad Q_2 \quad Q_3\,]$ can stabilize the closed-loop system asymptotically without the help of u_2. Based on $h(x)$ and $V(x)$, we design LMPC 1 to determine u_1 and LMPC 2 to determine u_2 following the forms given in Eqs. (6) and (8), respectively. In the design of the LMPC controllers, the weighting matrices are chosen to be $Q_c = \mathrm{diag}([\,10 \;\; 10^4 \;\; 10^4 \;\; 10^4 \;\; 9 \;\; 10^4 \;\; 10^4 \;\; 10^4 \;\; 10 \;\; 10^4 \;\; 10^4 \;\; 10^4\,])$, $R_1 = \mathrm{diag}([(\,5 \;\; 5 \;\; 5\,) \cdot 10^{-4}])$ and $R_2 = 10^4$. The prediction horizon for the optimization problem is $N = 5$ with a time step of $\Delta = 0.01$ h. The initial condition which is utilized to carry out the simulations is $x(0)^T = [362.14 \;\; 3.1191 \;\; 0.13 \;\; 0.01 \;\; 348.21 \;\; 2.01 \;\; 0.16 \;\; 0.01 \;\; 462.55 \;\; 2.31 \;\; 0.26 \;\; 0.01]$.
We set the communication channel noise power (σ^2), the data loss probability α and the noise bound θ to 0.01, 0.1 and 0.25, respectively. Fig. 4 depicts the proposed control design for the chemical process example which is composed of two LMPCs.

The state trajectory of the process under the proposed DMPC design from the initial state are shown in Fig. 5. These figures show that the proposed control design drive the temperatures and the concentrations in the closed-loop system close to the desired steady-state and achieves closed-loop stability.

To emphasize the importance of solving the feasibility problem in LMPC 1 during obtaining its optimal input trajectory, we have carried out a set of simulations to compare the proposed design with our previous control scheme [20] in which LMPC 1 incorporates the received channel signal in its optimization problem without any pre-processing. In other words, in this case LMPC 1 ignores the fact that whether communication channel noise and data loss effects violate the feasibility constraints of LMPC 2 optimization problem. We have carried out a number of simulations to compare the proposed DMPC design with our previous DMPC design with the same parameters and initial condition from a performance index point of view. Table 6 shows the total cost computed for 10 different closed-loop simulations under the proposed DMPC design and our previous control scheme. To carry out this comparison, we have computed the total cost of each simulation with different operating conditions (different initial states and process disturbances) based on the index of the following form:

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

where t_0 is the initial time of the simulations and $t_G = 1$ h is the final time of the simulations. As we can see in Table 6, the proposed distributed LMPC design has a cost lower than the previous DMPC design in all 10 simulations. This illustrates that in this example, the proposed distributed LMPC design improves our previous design from a closed-loop performance point of view.

Table 6
Total performance cost ($\times 10^7$) along the closed-loop system trajectories.

sim.	Prop.	Prev.	sim.	Prop.	Prev.
1	5.486	5.488	6	2.549	2.559
2	2.497	2.519	7	1.691	1.697
3	1.771	1.785	8	6.688	6.695
4	1.203	1.215	9	6.632	6.633
5	3.163	3.181	10	2.498	2.515

Table 5
Steady-state values for x_s.

C_{A1s}	3.31 [kmol/m^3]	C_{A2s}	2.75 [kmol/m^3]	C_{A3s}	2.88 [kmol/m^3]
C_{B1s}	0.17 [kmol/m^3]	C_{B2s}	0.45 [kmol/m^3]	C_{B3s}	0.50 [kmol/m^3]
C_{C1s}	0.04 [kmol/m^3]	C_{C2s}	0.11 [kmol/m^3]	C_{C3s}	0.12 [kmol/m^3]
T_{1s}	369.53 [K]	T_{2s}	435.25 [K]	T_{3s}	435.25 [K]

Table 7
Total performance cost ($\times 10^7$) along the closed-loop system trajectories for different data loss probabilities and $\sigma^2 = 0.01$.

α	Prop.	Prev.	α	Prop.	Prev.
0.05	6.803	6.900	0.30	6.808	6.901
0.10	6.779	6.908	0.35	6.818	6.906
0.15	6.821	6.897	0.40	6.779	6.901
0.20	6.821	6.905	0.45	6.793	6.893
0.25	6.801	6.899	0.50	6.744	6.895

Table 8
Total performance cost ($\times 10^7$) along the closed-loop system trajectories for different channel noise power values and $\alpha = 0.1$.

σ^2	Prop.	Prev.	σ^2	Prop.	Prev.
0.005	6.787	6.907	0.030	6.802	6.899
0.010	6.762	6.894	0.035	6.809	6.894
0.015	6.820	6.895	0.040	6.769	6.939
0.020	6.744	6.898	0.045	6.835	6.909
0.025	6.841	6.893	0.050	6.756	6.892

Finally, we have carried out a set of simulations to evaluate the performance of the proposed DMPC design over the one in [20] from a closed-loop performance index point of view under different communication channel noise powers and data loss probabilities. Tables 7 and 8 show the total cost computed for 10 different data loss probabilities and noise powers compared to our previous DMPC design, respectively. As it can be seen from these tables, the proposed DMPC design is superior from a closed-loop performance point of view for different noise power and data loss probability values. It should be mentioned that the number of feasible and infeasible solutions of the optimization problem of Eq. (7) depends on the bound on the communication channel noise; as this bound increases, the number of feasible solutions increases. For the simulation results corresponding to Fig. 5, LMPC 1 utilizes the received signal about 8% of the total number of transmissions.

Remark 9. Note that the DMPC design in [20] can still guarantee the closed-loop system stability in the presence of communication disruptions; however, the closed-loop performance may be degraded. In this work, we propose a practical approach to deal with communication disruptions to improve the closed-loop performance while maintaining the stability properties of the closed-loop system. In all simulations, the proposed DMPC design accounting for disruptions yields reduced performance costs compared to the previous DMPC design, even though this benefit cannot be proved to hold in general.

6. Conclusions

In this work, we proposed a DMPC design for nonlinear systems taking into account explicitly communication disruptions (i.e., data losses and channel noise) between the distributed controllers. In the proposed DMPC architecture, one of the distributed controllers is responsible for ensuring closed-loop stability while the rest of the distributed controllers communicate and cooperate with the stabilizing controller to further improve the closed-loop performance. To determine if the data transmitted through the communication channel is reliable or not, feasibility problems were incorporated in the DMPC design and based on the result of these feasibility problems, the transmitted information was accepted or rejected by the stabilizing MPC. In order to ensure the stability of the closed-loop system under communication disruptions, each distributed controller utilized a stability constraint which is based on a suitable Lyapunov-based controller. The proposed DMPC system possesses an explicit characterization of the closed-loop system stability region and guarantees that the closed-loop system is ultimately bounded in an invariant set which contains the origin. The theoretical results were demonstrated through a nonlinear chemical process example.

Acknowledgement

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

References

[1] E.B. Ydstie, New vistas for process control: integrating physics and communication networks, AIChE Journal 48 (2002) 422–426.
[2] P. Neumann, Communication in industrial automation: what is going on? Control Engineering Practice 15 (2007) 1332–1347.
[3] P.D. Christofides, J.F. Davis, N.H. El-Farra, D. Clark, K.R.D. Harris, J.N. Gipson, Smart plant operations: vision, progress and challenges, AIChE Journal 53 (2007) 2734–2741.
[4] L.A. Montestruque, P.J. Antsaklis, On the model-based control of networked systems, Automatica 39 (2003) 1837–1843.
[5] L.A. Montestruque, P.J. Antsaklis, Stability of model-based networked control systems with time-varying transmission times, IEEE Transactions on Automatica Control 49 (2004) 1562–1572.
[6] D. Ne, A.R. Teelšic′, Input-to-state stability of networked control systems, Automatica 40 (2004) 2121–2128.
[7] J. Liu, D. Muñoz de la Peña, B.J. Ohran, P.D. Christofides, J.F. Davis, A two-tier control architecture for nonlinear process systems with continuous/asynchronous feedback, International Journal of Control 83 (2010) 257–272.
[8] J.B. Rawlings, Tutorial overview of model predictive control, IEEE Control Systems Magazine 20 (June (3)) (2000) 38–52.
[9] E. Camponogara, D. Jia, B.H. Krogh, S. Talukdar, Distributed model predictive control, IEEE Control Systems Magazine 22 (2002) 44–52.
[10] J.B. Rawlings, B.T. Stewart, Coordinating multiple optimization-based controllers: new opportunities and challenges, Journal of Process Control 18 (2008) 839–845.
[11] R. Scattolini, Architectures for distributed and hierarchical model predictive control—a review, Journal of Process Control 19 (2009) 723–731.
[12] A.N. Venkat, J.B. Rawlings, 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.
[13] W.B. Dunbar, Distributed receding horizon control of dynamically coupled nonlinear systems, IEEE Transactions on Automatic Control 52 (2007) 1249–1263.
[14] J.M. Maestre, D. Muñoz de la Peña, E.F. Camacho, A distributed MPC scheme with low communication requirements, in: Proceedings of 2009 American Control Conference, Saint Louis, MO, USA, 2009, pp. 2797–2802.
[15] A. Richards, J.P. How, Robust distributed model predictive control, International Journal of Control 80 (2007) 1517–1531.
[16] T. Keviczky, F. Borrelli, G.J. Balas, Decentralized receding horizon control for large scale dynamically decoupled systems, Automatica 42 (2006) 2105–2115.
[17] L. Magni, R. Scattolini, Stabilizing decentralized model predictive control of nonlinear systems, Automatica 42 (2006) 1231–1236.
[18] P. Mhaskar, A. Gani, P.D. Christofides, Fault-tolerant control of nonlinear processes: performance-based reconfiguration and robustness, International Journal of Robust and Nonlinear Control 16 (2006) 91–111.
[19] Y. Sun, N.H. El-Farra, Quasi-decentralized model-based networked control of process systems, Computers and Chemical Engineering 32 (2008) 2016–2029.
[20] J. Liu, D. Muñoz de la Peña, P.D. Christofides, Distributed model predictive control of nonlinear process systems, AIChE Journal 55 (2009) 1171–1184.
[21] J. Liu, D. Muñoz de la Peña, P.D. Christofides, Distributed model predictive control of nonlinear systems subject to asynchronous and delayed measurements, Automatica 46 (2010) 52–61.
[22] M. Tabbara, D. Ne, A.R. Teelšic′, Stability of wireless and wireline networked control systems, IEEE Transactions on Automatic Control 52 (2007) 1615–1630.
[23] D. Muñoz de la Peña, P.D. Christofides, Lyapunov-based model predictive control of nonlinear systems subject to data losses, IEEE Transactions on Automatic Control 53 (2008) 2076–2089.
[24] J.L. Massera, Contributions to stability theory, Annals of Mathematics 64 (1956) 182–206.
[25] Y. Lin, E.D. Sontag, Y. Wang, A smooth converse Lyapunov theorem for robust stability, SIAM Journal on Control and Optimization 34 (1996) 124–160.
[26] P.D. Christofides, N.H. El-Farra, Control of Nonlinear and Hybrid Process Systems: Designs for Uncertainty, Constraints and Time-delays, Springer-Verlag, Berlin, Germany, 2005.
[27] D. Chilin, J. Liu, D. Muñoz de la Peña, P.D. Christofides, J.F. Davis, Detection, isolation and handling of actuator faults in distributed model predictive control systems, Journal of Process Control 20 (2010) 1059–1075.
[28] T.M. Cover, J.A. Thomas, J. Wiley, Elements of Information Theory, Wiley, New York, 1991.
[29] J.G. Salehi, M. Salehi, Digital Communications, McGraw-Hill, New York, 2001.