Sequential and Iterative Architectures for Distributed Model Predictive Control of Nonlinear Process Systems. Part I: Theory

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

Abstract—In this work, we focus on distributed model predictive control (DMPC) of large scale nonlinear process systems in which several distinct sets of manipulated inputs are used to regulate the process. For each set of manipulated inputs, a different model predictive controller is used to compute the control actions. The controllers are able to communicate with the rest of the controllers in making its decisions. Under the assumption that the feedback of the states of the process is available to all the distributed controllers at each sampling time and a model of the plant is available, we propose two different DMPC architectures. In the first one, the distributed controllers use a one-directional communication network, are evaluated in sequence, and each controller is evaluated only once at each sampling time; in the second one, the distributed controllers utilize a bi-directional communication network, are evaluated in parallel and iterate to improve closed-loop performance. In the design of the distributed controllers, Lyapunov-based model predictive control (LMPC) techniques are used. To ensure the stability of the closed-loop system, each controller in both architectures incorporates a stability constraint which is based on a suitable Lyapunov-based controller. We prove that the proposed DMPC architectures enforce practical stability in the closed-loop system and ensure optimal performance.

I. INTRODUCTION

Model predictive control (MPC) is a popular control strategy based on a model of the process to predict the future evolution of the system at each sampling time along a given prediction horizon. Using these predictions, the manipulated input trajectory that minimizes a given performance index is computed by solving a suitable optimization problem. To obtain finite dimensional optimization problems, MPC optimizes over a family of piecewise constant trajectories with a fixed sampling time and a finite prediction horizon. Once the optimization problem is solved, only the first manipulated input value is implemented, discarding the rest of the trajectory and repeating the optimization in the next sampling step [1], [2]. Typically, MPC is studied from a centralized control point of view in which all the manipulated inputs of a control system are optimized with respect to an objective function in a single optimization problem. When the number of the state variable and manipulated inputs of the process, however, becomes large, the computational burden of the centralized optimization problem may increase significantly and may impede the applicability of a centralized MPC, especially in the case where nonlinear process models are used in the MPC. One feasible alternative to overcome this problem is to utilize a distributed MPC (DMPC) architecture in which the manipulated inputs are computed by more than one optimization problems in a coordinated fashion. The objective of the present work is to propose two DMPC architectures for nonlinear process systems to reduce the computational burden with respect to centralized MPC and to coordinate the distributed controllers in a suitable fashion to achieve stability and optimal performance of the closed-loop system.

With respect to available results on DMPC architectures, several DMPC methods have been proposed in the literature that deal with the coordination of separate MPC controllers that communicate in order to obtain optimal input trajectories in a distributed manner; see [3], [4], [5] for reviews of results in this area. More specifically, in [6], the problem of distributed control of dynamically coupled nonlinear systems that are subject to decoupled constraints was considered. In [7], [8], the effect of the coupling was modeled as a bounded disturbance compensated using a robust MPC formulation. In [9], 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 for linear systems. In [10], DMPC of decoupled systems (a class of systems of relevance in the context of multi-agents systems) was studied. In [11], a DMPC algorithm was proposed under the main condition that the system is nonlinear, discrete-time and no information is exchanged between local controllers, and in [12], DMPC for nonlinear systems was studied from an input-to-state stability point of view. In [13], [14], [15], a game theory based DMPC scheme for linear systems coupled through the inputs was proposed. In a recent work [16], we proposed a DMPC architecture with one-directional communication for general nonlinear process systems. In this architecture, two separate MPC controllers designed via Lyapunov-based MPC (LMPC) were considered, in which one LMPC was used to guarantee the stability of the closed-loop system and the other LMPC was used to improve the closed-loop performance. Generally, the computational burden of these DMPC methods is smaller compared to the one of the corresponding centralized MPC because of the formulation of optimization problems with a smaller number of decision variables.
In this work, we focus on the design of DMPC architectures for large scale nonlinear process systems in which several distinct sets of manipulated inputs are used to regulate the process. For each set of manipulated inputs, a different model predictive controller is used to compute the control actions. The controllers are able to communicate with the rest of the controllers in making its decisions. Under the assumption that the feedback of the states of the process is available to all the distributed controllers at each sampling time and a model of the plant is available, we propose two different DMPC architectures designed via LMPM techniques. In the first one, the distributed controllers use a one-directional communication network, are evaluated in sequence and each controller is evaluated only once at each sampling time; in the second one, the distributed controllers utilize a bi-directional communication network, are evaluated in parallel and iterate to improve closed-loop performance. To ensure the stability of the closed-loop system, each controller in both architectures incorporates a stability constraint which is based on a suitable Lyapunov-based controller. We prove that the proposed DMPC architectures enforce practical stability in the closed-loop system and ensure optimal performance. The proposed control architectures have also been applied to a catalytic alkylation of benzene process and compared with a centralized control design from stability, computational time and performance points of view [17].

II. PRELIMINARIES

A. Problem formulation

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

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

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

\[
U_i := \{u_i \in R^{n_{u_i}} : |u_i| \leq u_i^{\text{max}}\}, i = 1, \ldots, m
\]

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

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

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

**Remark 1:** In general, distributed control systems are formulated based on the assumption that the controlled process systems consist of decoupled or partially coupled subsystems. However, we consider a fully coupled process model; this is a very common occurrence in chemical process control as we illustrate in [17]. In our future work, we will extend the proposed distributed control systems to the case in which only local state information is available to each distributed controller based on distributed state estimation.

**Remark 2:** Note that the assumption that \(f, g_i, i = 1, \ldots, m\), and \(k\) are locally Lipschitz vector functions is a reasonable assumption for most of chemical processes. Note also that the assumption that the state \(x\) of the system is sampled synchronously is a widely used assumption in the research of process control. The proposed control system designs can be extended to the case where only part of the state \(x\) is measurable by designing an observer to estimate the whole state vector from output measurements and by designing the control system based on the measured and estimated states. In this case, the stability properties of the resulting output feedback control systems are affected by the convergency of the observer and need to be carefully studied.

B. Lyapunov-based controller

We assume that there exists a Lyapunov-based controller \(h(x) = [h_1(x) \ldots h_m(x)]^T\) with \(u_i = h_i(x), i = 1, \ldots, m\), which renders the origin of the nominal closed-loop system asymptotically stable while satisfying the input constraints for all the states \(x\) inside a given stability region. We note that this assumption is essentially equivalent to the assumption that the process is stabilizable or that the pair \((A,B)\) in the case of linear systems is stabilizable. Using converse Lyapunov theorems [18], [19], [20], this assumption implies that there exist functions \(\alpha_i(\cdot), i = 1, 2, 3, 4\) of class \(K\) and a continuously differentiable Lyapunov function \(V(x)\) for the nominal closed-loop system which is continuous and bounded in \(R^{n_x}\), that satisfy the following inequalities:

\[
\alpha_1(|x|) \leq V(x) \leq \alpha_2(|x|), \quad \frac{\partial V(x)}{\partial x} \leq \alpha_3(|x|)
\]

\[
\frac{\partial V(x)}{\partial x}(f(x) + \sum_{i=1}^{m} g_i(x)h_i(x)) \leq -\alpha_3(|x|)
\]

\[
h_i(x) \in U_i, \quad i = 1, \ldots, m
\]

for all \(x \in D \subseteq R^{n_x}\) where \(D\) is an open neighborhood of the origin. We denote the region \(\Omega_{\rho} \overset{1}{=} D\) as the stability region of the closed-loop system under the Lyapunov-based controller \(h(x)\). The construction of \(V(x)\) can be carried out in a number of ways using systematic techniques like, for example, sum-of-squares methods.

By continuity, the local Lipschitz property assumed for the vector fields \(f, g_i, i = 1, \ldots, m\), and \(k\) and taking into

\[\overset{1}{=}\]

\[\overset{2}{=}\]

\[\overset{3}{=}\]

\[\overset{4}{=}\]
account that the manipulated inputs $u_i$, $i = 1, \ldots, m$, and
the disturbance $w$ are bounded in convex sets, there exists a
positive constant $M$ such that

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

(3)

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

$$\left| \frac{\partial V}{\partial x} f(x) - \frac{\partial V}{\partial x} f(x') \right| \leq L_x |x - x'|$$

$$\left| \frac{\partial V}{\partial x} g_i(x) - \frac{\partial V}{\partial x} g_i(x') \right| \leq L_{u_i} |x - x'|, \ i = 1, \ldots, m$$

$$\left| \frac{\partial V}{\partial x} k(x) \right| \leq L_w$$

(4)

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

Remark 3: Different state feedback control laws for non-
linear systems have been developed using Lyapunov tech-
niques; the reader may refer to [21], [22], [20], [23], [24]
for results in this area including results on the design of bounded
Lyapunov-based controllers by taking explicitly into account
input constraints for broad classes of nonlinear systems.

C. Centralized LMPC

To take advantage of all the sets of manipulated inputs,
one option is to design a centralized MPC controller. In order
to guarantee robust stability of the closed-loop system, the
MPC controller must include a set of stability constraints.
To do this, we propose to use the LMPC controller proposed in
[25], [26] which guarantees practical stability of the
closed-loop system, allows for an explicit characterization of the
stability region and yields a reduced complexity optimization problem. LMPC is based on uniting receding horizon control with Lyapunov functions and computes the manipulated input trajectory solving a finite horizon con-
strained optimal control problem. The LMPC controller is based on the Lyapunov-based controller $h(x)$. The controller $h(x)$ is used to define a stability constraint for the LMPC
controller which guarantees that the LMPC controller inherits
the stability and robustness properties of the Lyapunov-based
controller $h(x)$. The LMPC controller introduced in [25],
[26] is based on the following optimization problem:

$$\min_{u_{c1} \ldots u_{cm} \in \mathcal{S}(\Delta)} \int_{0}^{N\Delta} \left[ \dot{x}(\tau)Q \dot{x}(\tau) + \sum_{i=1}^{m} u_{c1}^{T}(\tau) R_{ci} u_{ci}(\tau) \right] d\tau$$

(5a)

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

(5b)

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

(5c)

$$\dot{x}(0) = x(t_k)$$

(5d)

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

(5e)

where $S(\Delta)$ is the family of piece-wise constant functions
with sampling period $\Delta$, $N$ is the prediction horizon, $Q_i$
and $R_{ci}$, $i = 1, \ldots, m$, are positive definite weight matrices
that define the cost, $x(t_k)$ is the state measurement obtained
at $t_k$, $\dot{x}$ is the predicted trajectory of the nominal system
with $u_i$, $i = 1, \ldots, m$, the input trajectory computed by the
LMPC of Eq. 5.

The optimal solution to this optimization problem is denoted by $u_{ci}^*(\tau|t_k)$, $i = 1, \ldots, m$, which is defined for
$\tau \in [0, N\Delta)$. The LMPC controller is implemented with a receding horizon method; that is, at each sampling time $t_k$, the new state $x(t_k)$ is received from the sensors, the optimization problem of Eq. 5 is solved, and $u_{ci}^*(t-t_k|t_k)$, $i = 1, \ldots, m$ are applied to the closed-loop system for $t \in [t_k, t_{k+1})$.

The constraint of Eq. 5e guarantees that the value of the
time derivative of the Lyapunov function at the initial
evaluation time of the centralized LMPC controller is lower
or equal to the value obtained if only the Lyapunov-based
controller $h(x)$ is implemented in the closed-loop system in
a sample-and-hold fashion. This is the constraint that allows
proving that the centralized LMPC controller inherits the
stability and robustness properties of the Lyapunov-based
controller.

The manipulated inputs of the closed-loop system under
the above centralized LMPC controller are defined as follows

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

(6)

In what follows, we refer to this controller as the centralized
LMPC. The main property of this controller is that the origin
of the closed-loop system is practically stable for all initial
states inside the stability region $\Omega_p$ for a sufficient small sam-
ping time $\Delta$ and disturbance upper bound $\theta$. This property
is also guaranteed by the Lyapunov-based controller $h(x)$
when the controller is implemented in a sample-and-hold
fashion (see [27], [28] for results on sampled-data systems).
The main advantage of LMPC approaches with respect to the
Lyapunov-based controller is that optimality considerations
can be taken explicitly into account (as well as constraints
on the inputs and the states [26]) in the calculation of the
control actions within an online optimization framework.

III. DMPC ARCHITECTURES

In our previous work [16], we introduced a DMPC archi-
tecture for nonlinear process systems based on the scheme
shown in Fig. 1. In this DMPC architecture, two controllers
designed via LMPC technique were considered. One of the
two LMPC controllers (LMPC 1) was designed to guarantee
the stability of the closed-loop system and the other LMPC
controller (LMPC 2) was designed to improve the closed-
loop performance while maintaining the closed-loop stability
achieved by LMPC 1. This DMPC architecture required one-
directional communication between the two distributed con-
rolled and was proved that it guarantees practical stability of
the closed-loop system and has the potential to maintain
the closed-loop stability and performance in the face of new or
failing controllers or actuators and to reduce computational
burden in the evaluation of the optimal manipulated inputs compared with a fully centralized LMPC controller of the same input/output-space dimension.

In the present work, our objective is to extend our results in [16] and propose DMPC architectures including multiple MPCs for large scale nonlinear process systems. Specifically, we propose two different DMPC architectures. The first DMPC architecture is a direct extension of our previous work in [16] in which different MPC controllers are evaluated in sequence, only once at each sampling time, and require one-directional communication between consecutive distributed controllers (i.e., the distributed controllers are connected by pairs). In the second architecture, different MPC controllers are evaluated in parallel, once or more than once at each sampling time, and require bi-directional communication among all the distributed controllers (i.e., the distributed controllers are all interconnected) is used.

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

A. Sequential DMPC

In this subsection, we will discuss the direct extension of the results in [16] to include multiple MPC controllers, in which different controllers are evaluated in sequence, once at each sampling time, and one-directional communication between consecutive distributed controllers (i.e., the distributed controllers are connected by pairs) is used. A schematic of this architecture is shown in Fig. 2. We first present the proposed implementation strategy of this DMPC architecture, and then describe how to design the corresponding LMPC controllers. The proposed implementation strategy of this DMPC architecture is as follows:

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

2) For $j = m$ to 1

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

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

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

$$\min_{u_{s,i} \in \mathbb{S}(\Delta)} \int_0^N \left[\dot{x}^T(\tau)Q_x \dot{x}(\tau) + \sum_{i=1}^m u_{s,i}(\tau)^TR_{u,i}u_{s,i}(\tau)\right]d\tau$$

$$\dot{x}(\tau) = f(\dot{x}(\tau)) + \sum_{i=1}^m g_i(\dot{x}(\tau))u_{s,i}$$

$$u_{s,i}(\tau) = h_i(\dot{x}(\tau)), \quad i = 1, \ldots, j - 1,$$

$$\forall \tau \in [\Delta_l, (l + 1)\Delta], \quad l = 0, \ldots, N - 1$$

$$u_{s,i}(\tau) = u_{s,i}(\tau|t_k), \quad i = j + 1, \ldots, m$$

$$u_{s,i}(\tau) \in U_i$$

$$\dot{x}(0) = x(t_k)$$

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

where $\dot{x}$ is the predicted trajectory of the nominal system with $u_i$, $i = j + 1, \ldots, m$, the input trajectory computed by the LMPC controllers of Eq. 7 evaluated before LMPC $j$, $u_i$, $i = j + 1, \ldots, m - 1$, the corresponding elements of $h(x)$ applied in a sample-and-hold fashion and $u_{s,i}^*(\tau|t_k)$ denotes the future input trajectory of $u_i$ obtained by LMPC $i$ of the form of Eq. 7. The optimal solution to the optimization problem of Eq. 7 is denoted $u_{s,j}^*(\tau|t_k)$ which is defined for $\tau \in [0, N\Delta]$

The constraint of Eq. 7b is the nominal model of the system of Eq. 1, which is used to predict the future evolution of the system; the constraints of Eq. 7c define the value of the inputs evaluated after $u_j$ (i.e., $u_i$ with $i = j + 1, \ldots, m$); the constraints of Eq. 7d define the value of the inputs evaluated before $u_j$ (i.e., $u_i$ with $i = j + 1, \ldots, m$); the constraint of Eq. 7e is the constraint on the manipulated input $u_j$; the constraint of Eq. 7f sets the initial state for the opti-
mization problem; the constraint of Eq. 7g guarantees that
the contribution of input $u_j$ to the decrease rate of the time
derivative of the Lyapunov function at the initial evaluation
time, if $u_j = u_{s,j}^*(0|t_k)$ is applied, is bigger or equal to
the value obtained when $u_j = h_j(x(t_k))$ is applied. This
constraint allows proving the closed-loop stability properties
of the proposed controller.

The manipulated inputs of the proposed control design of
Eq. 7 are defined as follows:

$$u_i(t) = u_{s,i}^*(t - t_k|t_k), \quad i = 1, \ldots, m, \forall t \in [t_k, t_{k+1}). \quad (8)$$

In what follows, we refer to this DMPC architecture as the
decisional DMPC.

Remark 4: Note that, in order to simplify the description
of the implementation strategy proposed above, we do not
distinguish LMPC $m$ and LMPC 1 from the others. We
note that LMPC $m$ does not receive any information from
the other controllers and LMPC 1 does not have to send
information to any other controller.

The proposed DMPC architecture of Eqs. 7-8 computes the
inputs $u_i$, $i = 1, \ldots, m$, applied to the system of Eq. 1
in a way such that in the closed-loop system, the value of
the Lyapunov function at time instant $t_k$ (i.e., $V(x(t_k))$)
is a decreasing sequence of values with a lower bound.
Following Lyapunov arguments, this property guarantees
practical stability of the closed-loop system. This is achieved
due to the constraint of Eq. 7g. This property is presented
in Theorem 1 below.

Theorem 1: Consider the system of Eq. 1 in closed-loop
under the DMPC of Eqs. 7-8 based on the controller $h(x)$
that satisfies the condition of Eq. 2. Let $\epsilon_w > 0$, $\Delta > 0$ and
$\rho > \rho_s > 0$ satisfy the following constraint:

$$-\alpha_3(\rho_s^{-1}(\rho_s)) + L^* \leq -\epsilon_w/\Delta \quad (9)$$

with $L^* = (L_x + \sum_{i=1}^m L_{u_i} u_i^\text{max}) \Delta + L_w \theta$ with $M$, $L_x$, $L_{u_i}$
($i = 1, \ldots, m$) and $L_w$ being defined in Eqs. 3-4. For any
$N \geq 1$, if $x(t_0) \in \Omega_p$ and if $\rho^* \leq \rho$ where

$$\rho^* = \max\{V(x(t + \Delta)) : V(x(t)) \leq \rho_s\}, \quad (10)$$

then the state $x(t)$ of the closed-loop system is ultimately
bounded in $\Omega_p$.

Proof: The proof consists of two parts. We first prove that
the optimization problem of Eq. 7 is feasible for all
$j = 1, \ldots, m$ and $x \in \Omega_p$. Then we prove that, under
the proposed DMPC of Eqs. 7-8, the state of the system of Eq. 1
is ultimately bounded in $\Omega_p$. Note that the constraint of
Eq. 7g of each distributed controller is independent from the
decisions that the rest of the distributed controllers make.

Part 1: In order to prove the feasibility of the optimization
problem of Eq. 7, we only have to prove that there exists a
$u_{s,j}(0)$ which satisfies the input constraint of Eq. 7e and the
constraint of Eq. 7g. This is because the constraint of Eq. 7g is
only enforced on the first prediction step of $u_{s,j}(\tau)$ and in
the prediction time $\tau \in [\Delta, N\Delta)$, the input constraint of
Eq. 8 can be easily satisfied with $u_{s,j}(\tau)$ being any value in
the convex set $U_j$.

We assume that $x(t_k) \in \Omega_p$ ($x(t)$ is bounded in $\Omega_p$
which will be proved in Part 2). It is easy to verify that
the value of $u_{s,j}(0)$ such that $u_{s,j}(0) = h_j(x(t_k))$ satisfies the
input constraint of Eq. 7e (assumed property of $h(x)$ for
$x \in \Omega_p$) and the constraint of Eq. 7g, thus, the feasibility
of the optimization problem of LMPC $j$, $j = 1, \ldots, m$, is
guaranteed.

Part 2: From the condition of Eq. 2 and the constraint of
Eq. 7g, if $x(t_k) \in \Omega_p$, it follows that

$$\frac{\partial V}{\partial x}(f(x(t_k))) + \sum_{i=1}^m g_i(x(t_k))u_{s,i}^*(0|t_k))$$

$$\leq \frac{\partial V}{\partial x}(f(x(t_k))) + \sum_{i=1}^m g_i(x(t_k))h_i(x(t_k))$$

$$\leq -\alpha_3(|x(t_k)|). \quad (11)$$

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

$$\dot{V}(x(t)) = \frac{\partial V}{\partial x}(f(x(t))) + \sum_{i=1}^m g_i(x(t))u_{s,i}^*(0|t_k) + k(x(t))w(t). \quad (12)$$

Adding and subtracting $\frac{\partial V}{\partial x}(f(x(t_k))) + \sum_{i=1}^m g_i(x(t_k))u_{s,i}^*(0|t_k))$ and taking into account Eq. 11, we obtain the following inequality

$$\dot{V}(x(t)) \leq -\alpha_3(|x(t_k)|) + \frac{\partial V}{\partial x}(f(x(t)))$$

$$+ \sum_{i=1}^m g_i(x(t))u_{s,i}^*(0|t_k) + k(x(t))w(t)$$

$$- \frac{\partial V}{\partial x}(f(x(t_k))) + \sum_{i=1}^m g_i(x(t_k))u_{s,i}^*(0|t_k)). \quad (13)$$

Taking into account Eqs. 2 and 3, the following inequality
is obtained for all $x(t_k) \in \Omega_p/\Omega_p$, from Eq. 12

$$\dot{V}(x(t)) \leq -\alpha_3(\rho_s^{-1}(\rho_s)) + (L_x + \sum_{i=1}^m L_{u_i} u_i^\text{max})|x(t) - x(t_k)|$$

$$+ L_w |w(t)|. \quad (13)$$

Taking into account Eq. 3 and the continuity of $x(t)$, the
following bound can be written for all $t \in [t_k, t_{k+1}]

$$|x(t) - x(t_k)| \leq M\Delta. \quad (14)$$

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

3152
If the condition of Eq. 9 is satisfied, then there exists $\epsilon_w > 0$ such that the following inequality holds for $x(t_k) \in \Omega_p/\Omega_{p^*}$

$$V(x(t)) \leq -\epsilon_w/\Delta$$

for $t \in [t_k, t_{k+1})$. Integrating the inequality of Eq. 15 on $t \in [t_k, t_{k+1})$, we obtain that

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

and

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

for all $x(t_k) \in \Omega_p/\Omega_{p^*}$. Using Eqs. 16 and 17 recursively it can be proved that, if $x(t_0) \in \Omega_p/\Omega_{p^*}$, the state converges to $\Omega_{p^*}$ in a finite number of sampling times without leaving the stability region. Once the state converges to $\Omega_{p^*} \subseteq \Omega_{p^*}$, it remains inside $\Omega_{p^*}$ for all times. This statement holds because of the definition of $p^*$. This proves that the closed-loop system under the proposed DMPC of Eqs. 7-8 is ultimately bounded in $\Omega_{p^*}$.

### B. Iterative DMPC

An alternative approach to the sequential DMPC architecture presented in the previous subsection is to evaluate all the distributed controllers in parallel and iterate to improve closed-loop performance. A schematic of this control architecture is shown in Fig. 3. In this architecture, each distributed controller must be able to communicate with all the other controllers (i.e., the distributed controllers are all interconnected). More specifically, when a new state measurement is available at a sampling time, each controller evaluates and obtains its future input trajectory; and then each LMPC controller broadcasts its latest obtained future input trajectory to all the other controllers. Based on the newly received input trajectories, each LMPC controller evaluates its future input trajectory again, and this process is repeated until a termination condition is satisfied. Specifically, we propose to use the following implementation strategy:

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

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

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

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

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

Note that at the initial iteration, all controllers use $h(x)$ to approximate the input trajectories of all the other controllers. Note also that the number of iterations $c$ can be variable and it does not affect the closed-loop stability of the proposed DMPC architecture; a point that will be made clear below.

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

In order to proceed, we define $\hat{x}(\tau|t_k)$ for $\tau \in [0, N\Delta)$ as the nominal sampled trajectory of the system of Eq. 1 associated with the feedback control law $h(x)$ and sampling time $\Delta$ starting from $x(t_k)$. This nominal sampled trajectory is obtained by integrating recursively the following differential equation:

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

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

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

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

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

$$\begin{align}
\min_{u_{p, j}} & \int_{0}^{N\Delta} \left[\hat{x}(\tau)^T Q_{\hat{x}}(\tau) + \sum_{i=1}^{m} u_{p, i}(\tau)^T R_{p, i} u_{p, i}(\tau)\right] d\tau \\
\text{subject to} & \quad \dot{\hat{x}}(\tau) = f(\hat{x}(\tau)), \forall \tau \in [(l\Delta, (l+1)\Delta)), l = 0, \ldots, N-1 \\
& \quad u_{p, j}(\tau) = u^*_p, j(\tau|t_k), \forall \tau \neq j \\
& \quad \hat{x}(0) = x(t_k) \\
& \quad \frac{\partial V(x)}{\partial x} g_j(x(t_k))u_{p, j}(0) \leq \frac{\partial V(x)}{\partial x} g_j(x(t_k))h_j(x(t_k)) \quad (18f)
\end{align}$$

where $\hat{x}$ is the predicted trajectory of the nominal system with $u_k$, the input trajectory computed by the LMPC of Eq. 18 and all the other inputs are the optimal input trajectories at iteration $c - 1$ of the rest of distributed controllers. The optimal solution to the optimization problem of Eq. 18 is denoted $u^*_{p, j}(\tau|t_k)$ which is defined for $\tau \in [0, N\Delta)$.
Accordingly, we define the final optimal input trajectory of LMPC $j$ (that is, the optimal trajectories computed at the last iteration) as $u_{p,j}^*(\tau|t_k)$ which is also defined for $\tau \in [0, N\Delta)$.

The manipulated inputs of the proposed control design of Eq. 18 are defined as follows:

$$u_i(t) = u_{p,i}^*(t-t_k|t_k), \quad i = 1, \ldots, m, \quad \forall t \in [t_k, t_{k+1}). \quad (19)$$

In what follows, we refer to this DMPC architecture as the iterative DMPC. The stability property of the iterative DMPC is stated in the following Theorem 2.

**Theorem 2:** Consider the system of Eq. 1 in closed-loop under the DMPC of Eqs. 18-19 based on the controller $h(x)$ that satisfies the condition of Eq. 2. Let $\epsilon_w > 0$, $\Delta > 0$ and $\rho > \rho_s > 0$ satisfy the constraint of Eq. 9. For any $N \geq 1$ and $c \geq 1$, if $x(t_0) \in \Omega_\rho$ and if $\rho^* \leq \rho$ where $\rho^*$ is defined as in Eq. 10, then the state $x(t)$ of the closed-loop system is ultimately bounded in $\Omega_{\rho^*}$.

**Proof:** Similar to the proof of Theorem 1, the proof of Theorem 2 also consists of two parts. We first prove that the optimization problem of Eq. 18 is feasible for each iteration $c$ and $x \in \Omega_\rho$. Then we prove that, under the proposed DMPC scheme of Eqs. 18-19, the state of the system of Eq. 1 is ultimately bounded in $\Omega_{\rho^*}$.

**Part 1:** In order to prove the feasibility of the optimization problem of Eq. 18, we only have to prove that there exists a $u_{p,j}(0)$ which satisfies the input constraint of Eq. 18d and the constraint of Eq. 18f. This is because the constraint of Eq. 18f is only enforced on the first prediction step of $u_{p,j}(\tau)$ and in the prediction time $\tau \in [\Delta, N\Delta)$, the input constraint of Eq. 19 can be easily satisfied with $u_{p,j}(\tau)$ being any value in the convex set $U_j$.

We assume that $x(t_k) \in \Omega_\rho$ ($x(t)$ is bounded in $\Omega_\rho$ which will be proved in Part 2). It is easy to verify that the value of $u_{p,j}$ such that $u_{p,j}(0) = h_j(x(t_k))$ satisfies the input constraint of Eq. 18d (assumed property of $h(x)$ for $x \in \Omega_\rho$) and the constraint of Eq. 18f for all possible $c$, thus, the feasibility of LMPC $j$, $j = 1, \ldots, m$, is guaranteed.

**Part 2:** By adding the constraint of Eq. 18f of each LMPC together, we have

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

It follows from the above inequality and condition of Eq. 2 that

$$\frac{\partial V}{\partial x}(f(x(t_k))) + \sum_{j=1}^m g_j(x(t_k)) u_{p,j}^*(0|t_k)) \leq \frac{\partial V}{\partial x}(f(x(t_k))) + \sum_{j=1}^m g_j(x(t_k)) h_j(x(t_k))) \leq -\alpha_3(|x(t_k)|). \quad (20)$$

Following the same approach as in the proof of Theorem 1, we know that if condition of Eq. 9 is satisfied, then the state of the closed-loop system can be proved to be maintained in $\Omega_{\rho^*}$ under the proposed DMPC architecture of Eqs. 18-19.

**Remark 5:** Note that the DMPC designs have the same stability region $\Omega_\rho$ as the one of the Lyapunov-based controller $h(x)$. When the stability of the Lyapunov-based controller $h(x)$ is global (i.e., the stability region is the entire state space), then the stability of the DMPC designs is also global. Note also that any initial condition in $\Omega_\rho$, the DMPC designs are proved to be feasible.

**Remark 6:** We do not consider delays introduced into the system by the communication network or by the time needed to solve the optimization problems. In future works, these delays will be taken into account in the formulation of the controllers. In this work, state constraints have also not been considered but the proposed designs can be extended to handle state constraints by restricting the closed-loop stability region further to satisfy the state constraints.

**Remark 7:** The choice of the horizon of the DMPC designs does not affect the stability of the closed-loop system. For any horizon length $N \geq 1$, the closed-loop stability is guaranteed by the constraints of Eqs. 7g and 18f. However, the choice of the horizon does affect the performance of the DMPC designs.

**Remark 8:** Note that because the manipulated inputs enter the dynamics of the system of Eq. 1 in an affine manner, the constraints designed in the LMPC optimization problems of Eqs. 7 and 18 that guarantee the closed-loop stability can be decoupled for different distributed controllers as in Eqs. 7g and 18f.

**Remark 9:** In the sequential DMPC architecture presented in Section III-A, the distributed controllers are evaluated in sequence, which implies that the minimal time to obtain a set of solutions to all the LMPC controllers is the sum of the evaluation times of all the LMPC controllers; whereas in the iterative DMPC architecture proposed in Section III-B, the distributed controllers are evaluated in parallel, which implies that the minimal time to obtain a set of solutions to all the LMPC controllers in each iteration is the largest evaluation time among all the LMPCs.

**Remark 10:** Note that the sequential (or iterative) DMPC is not a direct decomposition of the centralized LMPC because the set of constraints of Eq. 7g (or Eq. 18f) for $j = 1, \ldots, m$ in the distributed LMPC formulation of Eq. 7 (or Eq. 18) imposes a different feasibility region from the one of the centralized LMPC of Eq. 5 which has a single constraint (Eq. 5e).

**Remark 11:** In general, there is no guaranteed convergence of the optimal cost or solution of an iterated DMPC (e.g., the DMPC architecture discussed in Section III-B) to the optimal cost or solution of a centralized MPC for general nonlinear constrained systems because of the non-convexity of the MPC optimization problems. The reader may refer to [3], [7] for discussions on the conditions under which convergence of the solution of a distributed linear or convex MPC design to the solution of a centralized MPC or a Pareto optimal solution is ensured in the context of linear systems.

**Remark 12:** Note also that in general there is no guarantee...
that the closed-loop performance of one (centralized or distributed) MPC architecture discussed in this work should be superior than the others since the solutions provided by these MPC architectures are proved to be feasible and stabilizing but the superiority of the performance of one MPC architecture over another is not established. This is because the MPC designs are implemented in a receding horizon scheme and the prediction horizon is finite; and also because the different MPC designs are not equivalent as we discussed in Remark 10 and the non-convexity property as we discussed in Remark 11. In applications of these MPC architectures, especially for chemical process control in which non-convex problems is a very common occurrence, simulations should be conducted before making decisions as to which architecture should be used. Please see [17] for the application of the proposed DMPC architectures to a catalytic alkylation of benzene process example.

IV. CONCLUSIONS

In the present work, we presented two different DMPC architectures for nonlinear process systems: sequential DMPC and iterative DMPC. In both architectures, the controllers were designed via LMPC techniques. In the sequential DMPC architecture, the controllers adopt a one-directional communication network and are evaluated in sequence and once at each sampling time; in the iterative DMPC architecture, the controllers utilize a bi-directional communication network, are evaluated in parallel and iterate to improve closed-loop performance. Each LMPC controller in both architectures incorporates a suitable stability constraint which ensures that the state of the closed-loop system under the proposed DMPC architectures is ultimately bounded in an invariant set.

REFERENCES