Networked Coordination-Based Distributed Model Predictive Control for Large-Scale System

Yi Zheng, Shaoyuan Li, and Hai Qiu

***Abstract*— A class of large-scale systems, which is naturally divided into many smaller interacting subsystems, is usually controlled by a distributed or decentralized control framework. In this case, how to improve the performance of the entire system is a problem. A novel distributed model predictive control (MPC) is proposed for improving the global performance. Each subsystem is controlled by a subsystem-based MPC. These controllers coordinate with each other through global performance optimization index, and take the interactions among subsystems into account when predicting states evolution. The stability analysis for the unconstrained distributed MPC is given for guiding the control parameters tuning. Numeric results show that the proposed architecture could guarantee satisfactory global performance under even strong interactions among subsystems.**

***Index Terms*— Distributed control system, distributed model predictive control (DMPC), model predictive control (MPC), networked model predictive control.**

I. Introduction

THERE are a class of complex large-scale systems which are composed of many physically or geographically divided subsystems. Each subsystem interacts with some so-called neighboring subsystems by their states and inputs. The technical target is to achieve a specific global performance of the entire system.

The classical centralized control solution, which could obtain a good global performance, is often impractical to be applied to a large-scale system for computational reason and lack of error tolerance. When the centralized controller fails or a control component fails, the entire system is out of control and the control integrity cannot be guaranteed.

The distributed (or decentralized) framework, where each subsystem is controlled by an independent controller, has the advantages of error-tolerance, less computational effort, and being flexible to system structure. Thus the distributed control framework is usually adopted in this class of system [1]–[5], in spite of the fact that the dynamic performance of centralized framework is better than it. Thus, how to improve global performance under distributed control framework is a valuable problem.

Model predictive control (MPC), as a highly practical control technology with high performance [6], has been successfully applied to various linear [7]–[9] and nonlinear [10]–[12] systems in the process industries, and is becoming more widespread [13]–[15]. The distributed framework of MPC, distributed MPC (DMPC), is also gradually developing for the control of large-scale systems with the development of communication network technologies in process industries which allow the control technologies and methodologies to utilize their potentials for improving control.

Some DMPC formulations are available in the literatures [1], [2], [16]–[25]. Among them, [17] proposed a DMPC scheme for large-scale linear time-invariant systems, in which the subsystems interact with each other with inputs, and each local controller uses the so-called local performance index. A DMPC for systems with polytopic description is presented in [25]. As pointed out by the authors of [18]–[20], the performance of the DMPC is, in most cases, worse than that of centralized MPC. An iterative algorithm for DMPC based on Nash optimality to improve the efficiency of DMPC solution was developed in [18]. Then, an extended scheme based on a so-called neighborhood optimization is proposed to improve the global performance [19]. A DMPC scheme, in which the optimization index of each subsystem-based MPC is the performance of whole system is also presented in [20], to guarantee the feasibility properties in iteration and to improve the global performance. However, these methods are based on an iterative algorithm, and they do not focus on the system in which each subsystem interacts with each other by both their states and their inputs with time delay existing in communication network.

Fortunately, [1] presents a coordination strategy for this class of system, which is a noniterative method. However, the control strategy of this method pursues the performance of local subsystem. Thus, [2] proposed a method based on the so-called neighborhood optimization to improve the global performance under the condition of limited network resources.

As an extension, this brief proposed another method, called "networked coordination-based DMPC (NC-DMPC)," for improving the global performance of this class of system when global information is available. In this method, the subsystem-based MPCs coordinate with each other through global performance optimization index. The interactions among subsystems

Manuscript received January 30, 2011; revised October 25, 2011; accepted April 6, 2012. Manuscript received in final form April 16, 2012. Date of publication May 14, 2012; date of current version April 17, 2013. This work was supported in part by the National Nature Science Foundation of China under Grant 60825302 and Grant 60934007, the Program of Shanghai Subject Chief Scientist, and Shu Guang Project supported by Shanghai Municipal Education Commission and Shanghai Education Development Foundation and the Key Project of Shanghai Science and Technology Commission 10JC1403400. Recommended by Associate Editor L. Xie.

Y. Zheng is with the Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China, and also with Department of Electrical Engineering, Shanghai Jiao Tong University, Shanghai 200240, China (e-mail: zhengyi@sjtu.org).

S. Li is with the Department of Automation, Shanghai Jiao Tong University, Shanghai 200240, China, and also with the Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai 200240, China (e-mail: syli@sjtu.edu.cn).

H. Qiu is with GE Global Research, Shanghai 201203, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2012.2196280

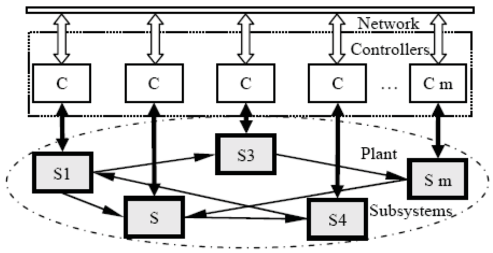

Fig. 1. Structures of distributed system and distributed control framework.

are estimated and considered according to the predictions of future states and inputs of each subsystem. A stability analysis of unconstrained NC-DMPC is provided for tuning the controllers and the performance of closed-loop system using proposed NC-DMPC is analyzed. Simulations are given to validate the efficiency of this method.

The contents are organized as follows. Section II describes the problem. Section III proposes the NC-DMPC and gives the closed-form solution of NC-DMPC. The stability and performance analysis is provided in Section IV. Section V gives the simulation results. Finally, a brief conclusion is drawn to summarize this brief.

II. Problem Description

A. System Description

For a class of large-scale system with hundreds (or thousands) of input and output variables, since centralized control is forbidden for the less flexibility and the large cost of computation, the distributed framework is usually adopted regardless of losing global performance. As shown in Fig. 1 the whole system is natural in view of the process layout partitioned properly into interconnected subsystems [26]. Each subsystem is controlled by a subsystem-based controller and these controllers are interconnected via network.

Without loss of generality, suppose that the whole system $\mathcal{S}$ is composed of m linear, discrete-time subsystems $\mathcal{S}_i, i = 1, \ldots, m$, and each subsystem interacts with others by both inputs and states. Then, according to [2] the state-space representation of $\mathcal{S}_i$ can be expressed as

$$\mathbf{x}_i(k+1) = \mathbf{A}_{ii}\mathbf{x}_i(k) + \mathbf{B}_{ii}\mathbf{u}_i(k) + \sum_{j=1,\ldots,m;\, j\neq i} \mathbf{A}_{ij}\mathbf{x}_j(k) + \sum_{j=1,\ldots,m;\, j\neq i} \mathbf{B}_{ij}\mathbf{u}_j(k) \tag{1a}$$

$$\mathbf{y}_i(k) = \mathbf{C}_{ii}\mathbf{x}_i(k) + \sum_{j=1,\ldots,m,\, j\neq i} \mathbf{C}_{ij}\mathbf{x}_j(k) \tag{1b}$$

where $\mathbf{x}_i \in \mathrm{R}^{n_{x_i}}$, $\mathbf{u}_i \in \mathrm{R}^{n_{u_i}}$, and $\mathbf{y}_i \in \mathrm{R}^{n_{y_i}}$, are the local state, input, and output vectors, respectively. The model of $\mathcal{S}$ can be expressed as

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}u(k) \tag{2a}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{2b}$$

where $\mathbf{x} \in \mathrm{R}^{n_x}$, $\mathbf{u} \in \mathrm{R}^{n_u}$, and $\mathbf{y} \in \mathrm{R}^{n_y}$, are state, input, and output vectors of $\mathcal{S}$, respectively. $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are system matrices.

The control objective minimizes the following global performance index:

$$J(k) = \sum_{i=1}^{m}\left(\sum_{l=1}^{P}\left\|\mathbf{y}_i(k+l) - \mathbf{y}_i^d(k+l)\right\|_{\mathbf{Q}_i}^2 + \sum_{l=1}^{M}\|\Delta\mathbf{u}_i(k+l-1)\|_{\mathbf{R}_i}^2\right) \tag{3}$$

where $\mathbf{y}_i^d$ and $\Delta\mathbf{u}_i(k)$ are output set point and input increment of $\mathcal{S}_j$, and $\Delta\mathbf{u}_i(k) = \mathbf{u}_i(k) - \mathbf{u}_i(k-1)$. $\mathbf{Q}_i$ and $\mathbf{R}_i$ are weight matrices, $P, M \in \mathrm{N}, P \geq M$, are predictive horizon and control horizon, respectively.

The problem is to design a coordination strategy for improving the global performance of the closed-loop system in the distributed framework.

B. Existing Methods

There are two DMPCs that appear in literature for the large-scale systems described above.

Distributed algorithms where each local controller minimizes local optimization objective [1]

$$J_i(k) = \sum_{l=1}^{P}\left\|\mathbf{y}_i(k+l) - \mathbf{y}_i^d(k+l)\right\|_{\mathbf{Q}_i}^2 + \sum_{l=1}^{M}\|\Delta\mathbf{u}_i(k+l-1)\|_{\mathbf{R}_i}^2. \tag{4}$$

In computing the optimal solution, each local controller exchanges estimation states with its neighbors, thereby improving the performance of local subsystem. However, the whole system's performance is not considered in this method.

Distributed algorithms are those in which each local controller minimizes a neighborhood cost function [2]

$$J_i'(k) = \sum_{i,j\in(\text{neighbor of } \mathrm{S_i})} J_j(k). \tag{5}$$

This strategy could achieve a better performance than that using local performance optimization index. However, since it balanced the communication resource, the global information is not fully used in each subsystem-based MPC.

In this brief, a method based on global performance optimization is proposed for large-scale systems in which the subsystems interact with each other by both inputs and states. The goal is to achieve a significantly improved performance of the whole system with full use of network resources.

III. NC-DMPC

A. NC-DMPC Formulation

The proposed control architecture is based on a set of independent MPC controllers $\mathcal{C}_i$ for each $\mathcal{S}_i, i = 1, 2, \ldots, m$. These MPCs could exchange information with its neighbors through a network. To clearly discuss the proposed control methodology, the simplifying Assumption 1 and the notations defined in Table I are adopted in this brief.

Assumption 1: 1) Controllers are synchronous, since the sampling interval is usually rather long compared with the computational time in process control; 2) communication channel introduces a delay of a single sampling time interval,

TABLE I
NOTATIONS DEFINITION[1]

Notations	Explanations
$\mathrm{diag}_a\{\mathbf{A}\}$	Diagonal block matrix made by a blocks equal to $\mathbf{A}$;
$\lambda_j\{\mathbf{A}\}$	The j^{th} eigenvalue of a square matrix $\mathbf{A}$;
$\mathcal{O}(a)$	To be proportional to a;
$\mathbf{0}_{a\times b}$	An $a \times b$ null matrix;
$\mathbf{0}_a$	An $a \times a$ null matrix;
$\mathbf{I}_a$	An $a \times a$ identity matrix;
$\hat{\mathbf{x}}_i(l \mid h)$	Predictions of $\mathbf{x}_i(l)$ computed by $\mathcal{C}_i$ at time h;
$\hat{\mathbf{y}}_i(l \mid h)$	Predictions of $\mathbf{y}_i(l)$ computed by $\mathcal{C}_i$ at time h;
$\mathbf{u}_i(l \mid h)$	Input $\mathbf{u}_i(l)$ computed by $\mathcal{C}_i$ at time h;
$\Delta\mathbf{u}_i(l \mid h)$	Input increment $\Delta\mathbf{u}_i(l)$ computed by $\mathcal{C}_i$ at time h;
$\mathbf{y}_i^d(l \mid h)$	Set point of $\mathbf{y}_i(l \mid h)$;
$\mathbf{y}^d(l \mid h)$	Set point of $\mathbf{y}(l \mid h)$;
$\hat{\mathbf{x}}^i(l \mid h)$	Predictions of $\mathbf{x}(l)$ computed by $\mathcal{C}_i$ at time h;
$\hat{\mathbf{y}}^i(l \mid h)$	Predictions of $\mathbf{y}(l)$ computed by $\mathcal{C}_i$ at time h;
$\mathbf{U}_i(l, p \mid h)$	A complete input vector $\mathbf{U}_i(l, p \mid h) =$ $[\,\mathbf{u}_i^T(l \mid h)\ \mathbf{u}_i^T(l+1 \mid h)\ \cdots\ \mathbf{u}_i^T(l+p \mid h)\,]^T$;
$\Delta\mathbf{U}_i(l, p \mid h)$	Input increment sequence vector, $\Delta\mathbf{U}_i(l, p \mid h) =$ $[\Delta\,\mathbf{u}_i^T(l \mid h)\ \Delta\mathbf{u}_i^T(l+1 \mid h)\ \cdots\ \Delta\mathbf{u}_i^T(l+p \mid h)\,]^T$
$\mathbf{U}(l, p \mid h)$	Complete stacked input vector, $\mathbf{U}(l, p \mid h) =$ $[\,\mathbf{u}^T(l \mid h)\ \mathbf{u}^T(l+p \mid h)\ \cdots\ \mathbf{u}^T(l+p \mid h)\,]^T$;
$\hat{\mathbf{X}}^i(l, p \mid h)$	Sacked distributed state vector, $\hat{\mathbf{X}}^i(l, p \mid h) =$ $[\,\hat{\mathbf{x}}^{iT}(l \mid h)\ \hat{\mathbf{x}}^{iT}(l+1 \mid h)\ \cdots\ \hat{\mathbf{x}}^{iT}(l+p \mid h)\,]^T$;
$\hat{\mathbf{X}}_i(l, p \mid h)$	Stacked distributed state vector, $\hat{\mathbf{X}}_i(l, p \mid h) =$ $[\,\hat{\mathbf{x}}_i^T(l \mid h)\ \hat{\mathbf{x}}_i^T(l+1 \mid h)\ \cdots\ \hat{\mathbf{x}}_i^T(l+p \mid h)\,]^T$;
$\hat{\mathbf{X}}(l, p \mid h)$	Complete stacked state vector, $\hat{\mathbf{X}}(l, p \mid h) =$ $[\,\hat{\mathbf{x}}^T(l \mid h)\ \hat{\mathbf{x}}^T(l+1 \mid h)\ \cdots\ \hat{\mathbf{x}}^T(l+p \mid h)\,]^T$;
$\hat{\mathbf{Y}}^i(l, p \mid h)$	Stacked distributed state vector, $\hat{\mathbf{Y}}^i(l, p \mid h) =$ $[\,\hat{\mathbf{y}}^{iT}(l \mid h)\ \hat{\mathbf{y}}^{iT}(l+1 \mid h)\ \cdots\ \hat{\mathbf{y}}^{iT}(l+p \mid h)\,]^T$;
$\mathbf{Y}(l, p \mid h)$	Complete stacked state vector, $\mathbf{Y}(l, p \mid h) =$ $[\,\mathbf{y}^T(l \mid h)\ \mathbf{y}^T(l+1 \mid h)\ \cdots\ \mathbf{y}^T(l+p \mid h)\,]^T$;
$\mathbf{Y}^d(l, p \mid h)$	Set point of $\mathbf{Y}(l, p \mid h)$;
$\hat{\mathrm{X}}(l, p \mid h)$	Complete stacked state vector, $\hat{\mathrm{X}}(l, p \mid h) =$ $[\,\hat{\mathbf{X}}_1^T(l, p \mid h)\ \cdots\ \hat{\mathbf{X}}_m^T(l, p \mid h)\,]^T$;
$\mathrm{Y}^d(l, p \mid h)$	$\mathrm{Y}^d(l, p \mid h) = \mathrm{diag}_m(\mathbf{Y}^d)$;
$\mathrm{U}(l, p \mid h)$	Complete stacked state vector, $\mathrm{U}(l, p \mid h) =$ $[\,\mathbf{U}_1^T(l, p \mid h)\ \cdots\ \mathbf{U}_m^T(l, p \mid h)\,]^T$.

[1] a and b are constants; p, l, h are positive integers, and $h < l$; $\mathbf{A}$ is a matrix.

since an instantaneous data transfer is not possible in real situations; 3) controllers communicate only once within a sampling time interval; and 4) local states $\mathbf{x}_i(k), i = 1, 2, \ldots, m$, are accessible.

1) Performance Index: Since the optimal control decision of $\mathcal{S}_i$ affects, or even destroys, the performance of other subsystems, the performance of other subsystems should be considered in finding the optimal of solution of $\mathcal{S}_i$. To improve the global performance of whole closed-loop system, the following so-called global performance index is adopted in each $\mathcal{C}_i$, $i = 1, \ldots, m$:

$$\bar{J}_i(k) = \sum_{l=1}^{P} \left\| \hat{\mathbf{y}}^i(k+l \mid k) - \mathbf{y}^d(k+l \mid k) \right\|_{\mathbf{Q}}^2 + \sum_{l=1}^{M} \left\| \Delta\mathbf{u}_i(k+l-1 \mid k) \right\|_{\mathbf{R}_i}^2 \tag{6}$$

where $\mathbf{Q} = \mathrm{diag}\{\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_m\}$. It should be noted that $\Delta\mathbf{u}_j(k+l-1 \mid k)$ is excluded in the performance index, since it is independent of the future inputs sequence of $\mathcal{S}_j$.

2) Prediction Model: Since the state evolution of other subsystems is affected by $\mathbf{u}_i(k)$ after one or several control periods, to improve the prediction precision, this influence is considered in $\mathcal{C}_i$ when predicting the future states of all subsystems. In addition, due to the unit delay introduced by the network, the information of other subsystems is available only after one sampling time interval. Therefore, in $\mathcal{C}_i$, the states and outputs of all subsystems in l-step ahead are predicted by

$$\hat{\mathbf{x}}^i(k+l+1 \mid k) = \mathbf{A}^l\mathbf{L}_i\mathbf{x}(k) + \mathbf{A}^l\mathbf{L}'_i\hat{\mathbf{x}}(k \mid k-1) + \sum_{s=1}^{l}\mathbf{A}^{s-1}\mathbf{B}_i\mathbf{u}_i(k+l \mid k) + \sum_{\substack{j \in \{1,\ldots,m\} \\ j \neq i}} \times \sum_{s=1}^{l}\mathbf{A}^{s-1}\mathbf{B}_j\mathbf{u}_j(k+l \mid k-1) \tag{7a}$$

$$\hat{\mathbf{y}}^i(k+l+1 \mid k) = \hat{\mathbf{C}}\mathbf{x}^i(k+l+1 \mid k) \tag{7b}$$

where $\mathbf{L}_i = [\mathbf{0}_{n_{xi}\times\sum_{j=1}^{i-1}n_{xj}}\ \mathbf{I}_{n_{xi}}\ \mathbf{0}_{n_{xi}\times\sum_{j=i+1}^{m}n_{xj}}\,]$, $\mathbf{L}'_i = \mathrm{diag}\{\mathbf{I}_{\sum_{j=1}^{i-1}n_{xj}}, \mathbf{0}_{n_{xi}}, \mathbf{I}_{\sum_{j=i+1}^{m}n_{xj}}\}$, $\mathbf{B}_i = [\,\mathbf{B}_{1i}^{\mathbf{T}}\ \mathbf{B}_{2i}^{\mathbf{T}}\ \cdots\ \mathbf{B}_{mi}^{\mathbf{T}}\,]^T$.

Remark 1: It should be noted that the input of this neighborhood model is still the input of $\mathcal{S}_i$, and the inputs and states of other subsystem are regarded as disturbances. The estimations of future states and outputs of all subsystems (except $\mathcal{S}_i$) are only used in controller $\mathcal{C}_i$, and these estimations are different from that estimated by the controller $\mathcal{C}_i$ itself.

3) Optimization Problem:

Definition 2: For each independent controller $\mathcal{C}_i, i = 1, \ldots, m$, the unconstrained NC-DMPC problem with prediction horizon P and control horizon $M, M < P$, at time k is to minimize the performance index (6) with system equation constraint (7)

$$\min_{\Delta\mathbf{U}_i(k,M \mid k)} \sum_{l=1}^{P} \left\| \hat{\mathbf{y}}^i(k+l \mid k) - \mathbf{y}^d(k+l \mid k) \right\|_{\mathbf{Q}}^2 + \sum_{l=1}^{M} \left\| \Delta\mathbf{u}_i(k+l-1 \mid k) \right\|_{\mathbf{R}_i}^2$$
$$\text{s.t. (7).} \tag{8}$$

At time k, based on the exchanged information $\hat{\mathbf{x}}_j(k \mid k-1)$, $\mathbf{U}_j(k+l \mid k-1)$, together with $\mathbf{x}(k)$, the optimization problem (8) is solved in each independent $\mathcal{C}_i$. The first element of the optimal solution is selected and $\mathbf{u}_i(k) = \mathbf{u}_i(k-1) + \Delta\mathbf{u}_i(k \mid k)$ is applied to $\mathcal{S}_j$. Then, by (7), each local controller estimates the future state at time $k+1$ and broadcasts it in network together with the optimal control sequence over the control horizon. At time $k+1$, each local controller uses this information to repeat the whole procedure.

B. Closed-Form Solution

The main result of this subsection is the computation of a closed-form solution to the NC-DMPC problem. For this purpose, the NC-DMPC (8) is first transformed into a quadratic program (QP) problem which has to be locally solved online at each sampling instant.

Define

$$\tilde{\mathbf{T}}_i = \operatorname{diag}\left\{\mathbf{I}_{\sum_{j=1}^{i-1} n_{uj}}, \mathbf{0}_{n_{ui}}, \mathbf{I}_{\sum_{j=i+1}^{M} n_{uj}}\right\} \tag{9a}$$

$$\tilde{\mathbf{B}}_i = \left[\begin{array}{c|c} \mathbf{0}_{(M-1)n_x\times n_u} & \operatorname{diag}_{M-1}\{\mathbf{B}\tilde{\mathbf{T}}_i\} \\ \hline \mathbf{0}_{n_x\times(M-1)n_u} & \mathbf{B}\tilde{\mathbf{T}}_i \\ \vdots & \vdots \\ \mathbf{0}_{n_x\times(M-1)n_u} & \mathbf{B}\tilde{\mathbf{T}}_i \end{array}\right] \tag{9b}$$

$$\bar{\mathbf{S}} = \begin{bmatrix} \mathbf{A}^0 & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}^1 & \mathbf{A}^0 & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{A}^{P-1} & \cdots & \mathbf{A}^1 & \mathbf{A}^0 \end{bmatrix}$$

$$\bar{\mathbf{A}}_a = \begin{bmatrix} \mathbf{A} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix} \tag{9c}$$

$$\bar{\mathbf{B}}_i = \left[\begin{array}{c|c} \mathbf{0}_{(M-1)n_x\times n_{ui}} & \operatorname{diag}_{M-1}\{\mathbf{B}_i\} \\ \hline \mathbf{0}_{n_x\times(M-1)n_{ui}} & \mathbf{B}_i \\ \vdots & \vdots \\ \mathbf{0}_{n_x\times(M-1)n_{ui}} & \mathbf{B}_i \end{array}\right]$$

$$\bar{\mathbf{C}}_a = \operatorname{diag}_P\{\mathbf{C}\} \tag{9d}$$

$$\bar{\mathbf{\Gamma}}_i = \begin{bmatrix} \mathbf{I}_{n_{ui}} & \mathbf{0}_{n_{ui}} & \ldots & \mathbf{0}_{n_{ui}} \\ \mathbf{I}_{n_{ui}} & \mathbf{I}_{n_{ui}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0}_{n_{ui}} \\ \mathbf{I}_{n_{ui}} & \cdots & \mathbf{I}_{n_{ui}} & \mathbf{I}_{n_{ui}} \end{bmatrix}$$

$$\mathbf{\Gamma}'_i = [\overbrace{\mathbf{I}_{n_{ui}} \cdots \mathbf{I}_{n_{ui}}}^{M}]^T \tag{9e}$$

$$\mathbf{N}_i = \bar{\mathbf{C}}_a\bar{\mathbf{S}}\bar{\mathbf{B}}_i\bar{\mathbf{\Gamma}}_i, \quad \bar{\mathbf{Q}} = \operatorname{diag}_P\{\mathbf{Q}\}, \quad \bar{\mathbf{R}}_i = \operatorname{diag}_M\{\mathbf{R}_i\}. \tag{9f}$$

The following Lemma can be obtained based on (7)–(9).

Lemma 3 (QP): Under Assumptions 1, each independent controller $\mathcal{C}_i, i = 1, \ldots, m$ has to solve at time k the following optimization problem:

$$\min_{\Delta\mathbf{U}_i(k,M|k)} [\Delta\mathbf{U}_i^T(k, M|k)\mathbf{H}_i\Delta\mathbf{U}_i(k, M|k) - \mathbf{G}_i(k+1, P|k)\Delta\mathbf{U}_i(k, M|k)] \tag{10}$$

where the positive definite matrix $\mathbf{H}_i$ has the form

$$\mathbf{H}_i = \mathbf{N}_i^T\bar{\mathbf{Q}}\mathbf{N}_i + \bar{\mathbf{R}}_i \tag{11a}$$

and

$$\mathbf{G}_i(k+1, P|k) = 2\mathbf{N}_i^T\bar{\mathbf{Q}}[\mathbf{Y}^d(k+1, P|k) - \hat{\mathbf{Z}}_i(k+1, P|k)] \tag{11b}$$

with

$$\begin{aligned}\hat{\mathbf{Z}}_i(k+1, P|k) = \bar{\mathbf{C}}_a\bar{\mathbf{S}}[&\bar{\mathbf{B}}_i\mathbf{\Gamma}'_i\mathbf{u}_i(k-1) + \bar{\mathbf{A}}_a\mathbf{L}_i x_i(k|k) \\ &+ \bar{\mathbf{A}}_a\mathbf{L}'_i\hat{\mathbf{x}}(k|k-1) \\ &+ \tilde{\mathbf{B}}_i\mathbf{U}(k-1, M|k-1)].\end{aligned} \tag{11c}$$

The proof can be found in Appendix A.

According to (10), the solution to (8) can be deduced as $\Delta\mathbf{U}_i(k, M|k) = (1/2)\cdot\mathbf{H}_i^{-1}\mathbf{G}_i(k+1, P|k)$. By noting that only the first element of the optimal sequence is actually applied to the process, Theorem 4 is obtained.

Theorem 4 (Closed-Form Solution): Under Assumptions 1, for each controller $\mathcal{C}_i, i = 1, \ldots, m$, the closed-form control law applied at time k *to subsystem* $\mathcal{S}_i$ is given by

$$\mathbf{u}_i(k) = \mathbf{u}_i(k-1) + \mathbf{K}_i[\mathbf{Y}^d(k+1, P|k) - \hat{\mathbf{Z}}_i(k+1, P|k)] \tag{12}$$

where

$$\mathbf{K}_i = \mathbf{\Gamma}_i\bar{\mathbf{K}}_i, \mathbf{\Gamma}_i = [\,\mathbf{I}_{n_{u_i}}\ \ 0_{n_{u_i}\times(M-1)n_{u_i}}\,], \quad \bar{\mathbf{K}}_i = \mathbf{H}_i^{-1}\mathbf{N}_i^T\bar{\mathbf{Q}}. \tag{13}$$

Remark 2: In $\mathcal{C}_i$, the complexity to obtain the closed-form solution is mainly incurred by the inversion of $\mathbf{H}_i$. By using the Gauss–Jordan algorithm for this task and considering that the size of $\mathbf{H}_i$ equals $M\cdot n_{u_i}$, the complexity of inversion algorithm is $\mathcal{O}(M^3\cdot n_{u_i}^3)$. Therefore, the total computational complexity of solving NC-DMPC is only $\mathcal{O}(M^3\cdot\sum_{i=1}^n n_{u_i}^3)$ while the computational complexity of the centralized control strategy equals $\mathcal{O}(M^3\cdot(\sum_{i=1}^n n_{u_i})^3)$.

IV. Stability and Performance Analysis

A. Stability Analysis

On the basis of the closed-form solution stated by Theorem 4, the closed-loop dynamics can be specified and the stability condition can be verified by analyzing the closed-loop dynamic matrix. Define

$$\begin{aligned}\Omega &= [\,\Omega_1^T \cdots \Omega_P^T\,]^T, \quad \Omega_l = \operatorname{diag}\{\Omega_{1l}, \ldots, \Omega_{ml}\} \\ \Omega_{il} &= [\,\mathbf{0}_{n_{x_i}\times(l-1)n_{x_i}}\ \ \mathbf{I}_{n_{x_i}}\ \ \mathbf{0}_{n_{x_i}\times(P-l)n_{x_i}}\,] \\ &\quad (i = 1, \ldots, m, l = 1, \ldots, P)\end{aligned} \tag{14a}$$

$$\begin{aligned}\Pi &= [\,\Pi_1^T \cdots \Pi_M^T\,]^T, \quad \Pi_l = \operatorname{diag}\{\Pi_{1l}, \ldots, \Pi_{ml}\} \\ \Pi_{il} &= [\,\mathbf{0}_{n_{u_i}\times(l-1)n_{u_i}}\ \ \mathbf{I}_{n_{u_i}}\ \ \mathbf{0}_{n_{u_i}\times(M-l)n_{u_i}}\,] \\ &\quad (i = 1, \ldots, m, l = 1, \ldots, M)\end{aligned} \tag{14b}$$

$$\bar{\mathbf{A}} = \operatorname{diag}_m\{\bar{\mathbf{A}}_a\}, \quad \bar{\mathbf{B}} = \operatorname{diag}\{\bar{\mathbf{B}}_1, \ldots, \bar{\mathbf{B}}_m\} \tag{15a}$$

$$\bar{\mathbf{C}} = \operatorname{diag}_m\{\bar{\mathbf{C}}_a\}, \quad \bar{\mathbf{L}}_i = \operatorname{diag}_P\{\mathbf{L}_i^T\} \tag{15b}$$

$$\mathbf{L} = \operatorname{diag}\{\mathbf{L}_1, \ldots, \mathbf{L}_m\}, \quad \bar{\mathbf{L}} = \operatorname{diag}\{\bar{\mathbf{L}}_1, \ldots, \bar{\mathbf{L}}_m\} \tag{15c}$$

$$\mathbf{\Gamma}' = \operatorname{diag}\{\mathbf{\Gamma}'_1, \ldots, \mathbf{\Gamma}'_m\}, \quad \mathbf{\Gamma} = \operatorname{diag}\{\mathbf{\Gamma}_1, \ldots, \mathbf{\Gamma}_m\} \tag{15d}$$

$$\tilde{\mathbf{L}} = \mathbf{L}'[\,\mathbf{I}_{n_x}\ \ \mathbf{0}_{n_x\times(P-1)n_x}\,], \mathbf{L}' = [\,\mathbf{L}'^T_1 \cdots \mathbf{L}'^T_m\,]^T \tag{15e}$$

$$\tilde{\mathbf{B}} = [\,\tilde{\mathbf{B}}_1^T \cdots \tilde{\mathbf{B}}_m^T\,]^T, \quad \mathbf{S} = \operatorname{diag}_m\{\bar{\mathbf{S}}\} \tag{15f}$$

$$\Xi = \operatorname{diag}\{\bar{\mathbf{\Gamma}}_1\bar{\mathbf{K}}_1, \ldots, \bar{\mathbf{\Gamma}}_m\bar{\mathbf{K}}_m\} \tag{15g}$$

$$\Theta = -\Xi\bar{\mathbf{C}}\mathbf{S}\bar{\mathbf{A}}\mathbf{L} \tag{16a}$$

$$\Phi = -\Xi\bar{\mathbf{C}}\mathbf{S}\bar{\mathbf{A}}\tilde{\mathbf{L}}\mathbf{\Omega} \tag{16b}$$

$$\mathbf{\Psi} = \mathbf{\Gamma}'\mathbf{\Gamma} - \mathbf{\Gamma}'\mathbf{\Gamma}(\bar{\mathbf{B}}\mathbf{\Gamma}'\mathbf{\Gamma} + \tilde{\mathbf{B}}\mathbf{\Pi}). \tag{16c}$$

Then, the Theorem 5 can be deduced.

Theorem 5 (NC-DMPC Stability): The closed-loop system given by the feedback connection of plant $\mathcal{S}$ with the set of independent controller $\mathcal{C}_i, i = 1, \ldots, m$, whose closed-form control laws are given by (12), is asymptotically stable if and only if

$$\left|\lambda_j\{\mathbf{A}_N\}\right| < 1, \forall j = 1, \ldots, n_N \tag{17}$$

where

$$\mathbf{A}_N = \begin{bmatrix} \mathbf{A} & 0 & \mathbf{B\Gamma} & 0 \\ \bar{\mathbf{L}}\mathbf{S}\bar{\mathbf{A}}\mathbf{L} & \bar{\mathbf{L}}\mathbf{S}\bar{\mathbf{A}}\tilde{\mathbf{L}}\mathbf{\Omega} & \bar{\mathbf{L}}\mathbf{S}\bar{\mathbf{B}} & \bar{\mathbf{L}}\mathbf{S}\tilde{\mathbf{B}}\mathbf{\Pi} \\ \mathbf{\Theta A} + \mathbf{\Omega}\bar{\mathbf{L}}\mathbf{S}\bar{\mathbf{A}}\mathbf{L} & \mathbf{\Phi}\bar{\mathbf{L}}\mathbf{S}\bar{\mathbf{A}}\tilde{\mathbf{L}}\mathbf{\Omega} & \mathbf{\Theta B\Gamma} + \mathbf{\Phi}\bar{\mathbf{L}}\mathbf{S}\bar{\mathbf{B}} + \mathbf{\Psi} & \mathbf{\Phi}\bar{\mathbf{L}}\mathbf{S}\tilde{\mathbf{B}}\mathbf{\Pi} \\ 0 & 0 & \mathbf{I}_{Mn_u} & 0 \end{bmatrix}$$

$n_N = Pn_x + n_x + 2Mn_u$ is the order of the whole closed-loop system.

The proof can be found in Appendix B.

Remark 3: It should be noted that the first two block rows of dynamic matrix $\mathbf{A}_N$ included in (17) depend on the elements of matrix $\mathbf{A}$ and the elements of matrix $\mathbf{B}$, while the third block row depends on $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{Q}$, $\mathbf{R}_i$, P, and M. The degree of freedom available to the designer are on the choices of $\mathbf{Q}$, $\mathbf{R}_i$, P, and M, in the NC-DMPC design phase, which introduces significant modifications only on the third block row of $\mathbf{A}_N$ used for the stability test.

B. Performance Analysis

To explain the essential difference between the NC-MPC and the centralized MPC, for each $\mathcal{C}_i$, $i = 1, \ldots, m$, the optimization problem of NC-DMPC (8) is rewritten as

$$\begin{aligned}
&\min_{\Delta\mathbf{U}_i(k,M|k)} \sum_{l=1}^{P} \left\|\hat{\mathbf{y}}^i(k+l\,|k) - \mathbf{y}^d(k+l\,|k)\right\|_{\mathbf{Q}}^2 \\
&+ \sum_{l=1}^{M} \|\Delta\mathbf{u}_i(k+l-1\,|k)\|_{\mathbf{R}_i}^2 \\
&\text{s.t.} \begin{bmatrix} \hat{\mathbf{x}}_i^i(k+l+1\,|k) \\ \vdots \\ \hat{\mathbf{x}}_{i-1}^i(k+l+1\,|k) \\ \hat{\mathbf{x}}_i(k+l+1\,|k) \\ \hat{\mathbf{x}}_{i+1}^i(k+l+1\,|k) \\ \vdots \\ \hat{\mathbf{x}}_m^i(k+l+1\,|k) \end{bmatrix} = \mathbf{A}^l \begin{bmatrix} \hat{\mathbf{x}}_1(k\,|k-1) \\ \vdots \\ \hat{\mathbf{x}}_{i-1}(k\,|k-1) \\ \mathbf{x}_i(k\,|k) \\ \hat{\mathbf{x}}_{i+1}(k\,|k-1) \\ \vdots \\ \hat{\mathbf{x}}_m(k\,|k-1) \end{bmatrix} \\
&+ \sum_{s=1}^{l} \mathbf{A}^{s-1}\mathbf{B}\tilde{\mathbf{U}}(k,l\,|k) \\
&\tilde{\mathbf{U}}(k,M\,|k) = [\,\mathbf{u}_1^T(k\,|k-1)\ \ldots\ \mathbf{u}_{i-1}^T(k\,|k-1)\ \mathbf{u}_i(k\,|k) \\
&\mathbf{u}_{i+1}^T(k\,|k-1)\ \cdots\ \mathbf{u}_m^T(k\,|k-1)\ \cdots \\
&\mathbf{u}_1^T(k+l\,|k-1)\ \ldots\ \mathbf{u}_{i-1}^T(k+l\,|k-1)\ \mathbf{u}_i(k+l\,|k) \\
&\mathbf{u}_{i+1}^T(k+l\,|k-1)\ \cdots\ \mathbf{u}_m^T(k+l\,|k-1)\,]^T \\
&\hat{\mathbf{y}}^i(k+l\,|k) = \mathbf{C}\hat{\mathbf{x}}^i(k+l\,|k).
\end{aligned} \tag{18}$$

The optimization problem of centralized MPC can be written as

$$\begin{aligned}
&\min_{\Delta\mathbf{U}_i(k,M|k)} \sum_{l=1}^{P} \left\|\hat{\mathbf{y}}(k+l\,|k) - \mathbf{y}^d(k+l\,|k)\right\|_{\mathbf{Q}}^2 \\
&+ \sum_{l=1}^{M} \|\Delta\mathbf{u}_i(k+l-1\,|k)\|_{\mathbf{R}_i}^2 \\
&\text{s.t.} \begin{bmatrix} \hat{\mathbf{x}}_1(k+l+1\,|k) \\ \vdots \\ \hat{\mathbf{x}}_{i-1}(k+l+1\,|k) \\ \hat{\mathbf{x}}_i(k+l+1\,|k) \\ \hat{\mathbf{x}}_{i+1}(k+l+1\,|k) \\ \vdots \\ \hat{\mathbf{x}}_m(k+l+1\,|k) \end{bmatrix} = \mathbf{A}^l \begin{bmatrix} \mathbf{x}_1(k\,|k) \\ \vdots \\ \mathbf{x}_{i-1}(k\,|k) \\ \mathbf{x}_i(k\,|k) \\ \mathbf{x}_{i+1}(k\,|k) \\ \vdots \\ \mathbf{x}_m(k\,|k) \end{bmatrix} \\
&+ \sum_{s=1}^{l} \mathbf{A}^{s-1}\mathbf{B}\mathbf{U}(k,l\,|k) \\
&\tilde{\mathbf{U}}(k,M\,|k) = [\,\mathbf{u}_1^T(k\,|k)\ \ldots\ \mathbf{u}_{i-1}^T(k\,|k)\ \mathbf{u}_i(k\,|k) \\
&\mathbf{u}_{i+1}^T(k\,|k)\ \cdots\ \mathbf{u}_m^T(k\,|k)\ \cdots \\
&\mathbf{u}_1^T(k+l\,|k)\ \ldots\ \mathbf{u}_{i-1}^T(k+l\,|k)\ \mathbf{u}_i(k+l\,|k) \\
&\mathbf{u}_{i+1}^T(k+l\,|k)\ \cdots\ \mathbf{u}_m^T(k+l\,|k)\,]^T \\
&\hat{\mathbf{y}}(k+l\,|k) = \hat{\mathbf{C}}\mathbf{x}(k+l\,|k).
\end{aligned} \tag{19}$$

It can be seen that the performance indices in (18) and (19) are identical. The states evolution models are also similar. The only difference between these two problems is that, in NC-DMPC, the initial states and future control sequences of other subsystems at time k are substituted by the estimations calculated at time $k-1$. If there is disturbance, model mismatch or set point change, the future input sequences of subsystems calculated at time k are not equal to that calculated at time $k-1$, which induces estimation errors of future states between two optimization strategies. This affects the final performance of closed-loop system. Although this difference exists, the optimization problem of NC-DMPC is still very close to that of centralized MPC.

V. Simulation

In this section, the performance of the proposed NC-DMPC is investigated and compared with that of the networked decentralized MPC (N-DMPC) presented in [1]. Consider the nonminimum phase plant $\mathcal{S}$ appeared in [1]. We discretize this plant with a sampling time of 0.2 second, which yields

$$\begin{bmatrix} y_1(z) \\ y_2(z) \end{bmatrix} = \begin{bmatrix} \frac{-0.024(z-1.492)(z+0.810)}{(z-0.819)(z^2-1.922z+0.961)} & \alpha\frac{0.018(z+0.935)}{(z^2-1.676z+0.819)} \\ \alpha\frac{0.126}{(z-0.368)} & \frac{0.147(z-0.668)}{(z^2-1.572z+0.670)} \end{bmatrix} \begin{bmatrix} u_1(z) \\ u_2(z) \end{bmatrix}.$$

A state-space realization for $\mathcal{S}$ has the form (2), with matrices

$$\mathbf{A} = \begin{bmatrix} \mathbf{A}_{11} & 0 \\ 0 & \mathbf{A}_{22} \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \mathbf{B}_{11} & 0 \\ 0 & \mathbf{B}_{22} \end{bmatrix}$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{C}_{11} & \mathbf{C}_{12} \\ \mathbf{C}_{21} & \mathbf{C}_{22} \end{bmatrix}$$

$$\mathbf{A}_{11} = \begin{bmatrix} 2.74 & -1.27 & 0.97 & 0 \\ 2 & 0 & 0 & 0 \\ 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0.37 \end{bmatrix}$$

$$\mathbf{B}_{11} = \begin{bmatrix} 0.25 \\ 0 \\ 0 \\ 0.5 \end{bmatrix}$$

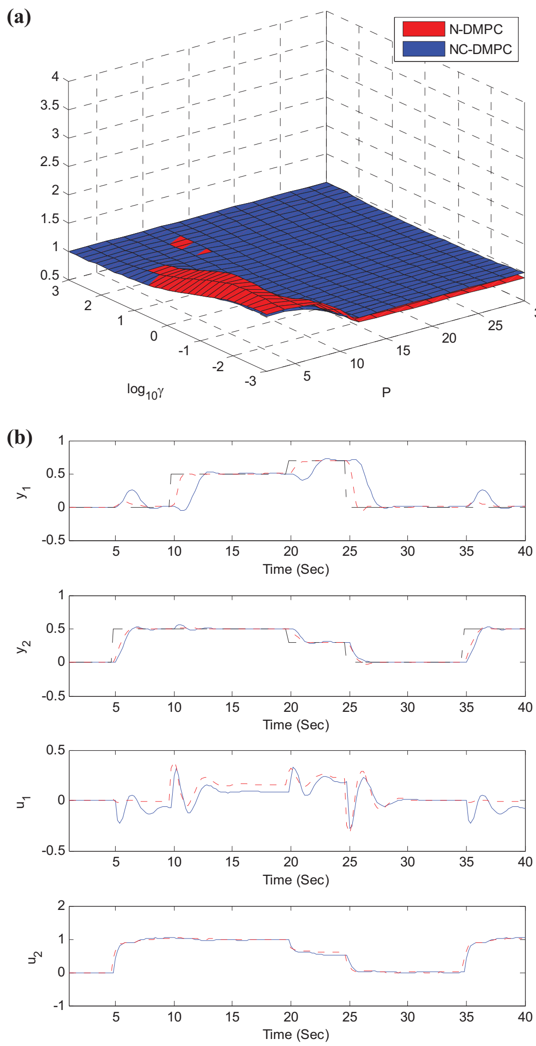

Fig. 2. Plant with $\alpha = 0.1$. (a) Maximum closed-loop eigenvalues with N-DMPC and NC-DMPC. (b) Control performance with $\gamma = 1$ and $P = 20$ for N-DMPC (blue line, MSE = 0.2568) and NC-DMPC (red line, MSE = 0.2086).

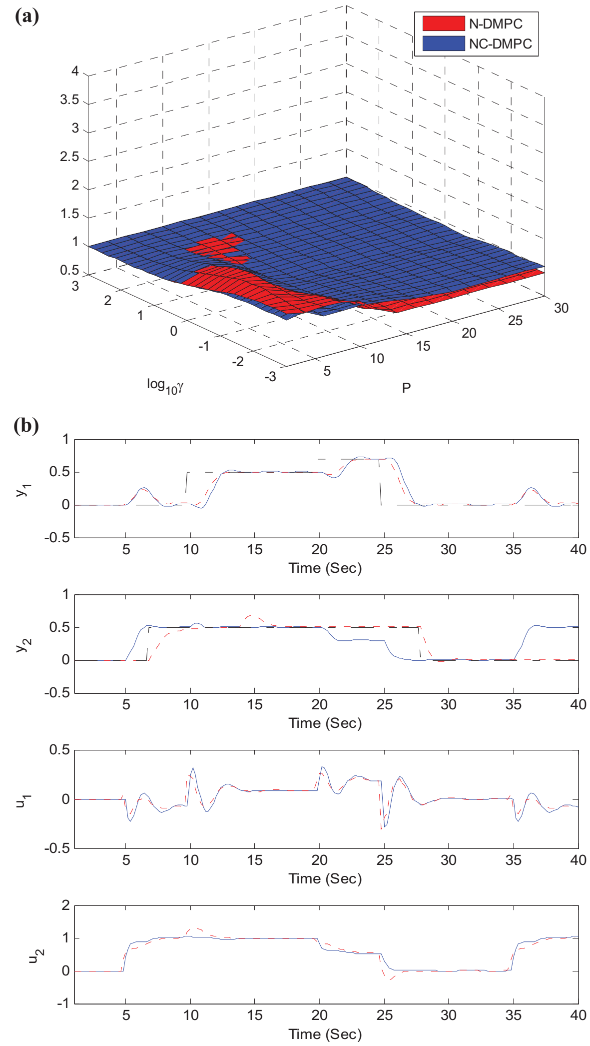

Fig. 3. Plant with $\alpha = 1$. (a) Maximum closed-loop eigenvalues with N-DMPC and NC-DMPC. (b) Control performance with $\gamma = 1$ and $p = 20$ for N-DMPC (blue line, MSE = 0.2277) and NC-DMPC (red line, MSE = 0.2034).

$$\mathbf{A}_{22} = \begin{bmatrix} 1.68 & -0.82 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1.57 & -0.67 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$
$$\mathbf{B}_{22} = \begin{bmatrix} 0.25 \\ 0 \\ 0.5 \\ 0 \end{bmatrix}$$
$$\mathbf{C}_{11} = [\,-0.1\ 0.03\ 0.12\ 0\,]$$
$$\mathbf{C}_{12} = \alpha[\,0.07\ 0.07\ 0\ 0\,]$$
$$\mathbf{C}_{21} = \alpha[\,0\ 0\ 0\ 2.25\,]$$
$$\mathbf{C}_{22} = [\,0\ 0\ 0.29\ -0.20\,].$$

Decompose $\mathcal{S}$ into two SISO subsystems, $\mathcal{S}_1$ and $\mathcal{S}_2$. The corresponding state-space realizations of $\mathcal{S}_1$ and $\mathcal{S}_2$ have the form (1), with matrices $\{\mathbf{A}_{11}, \mathbf{B}_{11}, \mathbf{C}_{11}\}$ and $\{\mathbf{A}_{22}, \mathbf{B}_{22}, \mathbf{C}_{22}\}$, respectively. The constant parameter α is used to study the interactions between $\mathcal{S}_1$ and $\mathcal{S}_2$.

Similar to N-DMPC, the computational complexity of NC-DMPC depends on the plant size, in particular on the number of inputs of independent controllers, while the feasibility depends on the possibility to satisfy Theorem 5 for a specific set of parameters P, M, $\mathbf{Q}$, and $\mathbf{R}_i$, $i = 1, \ldots, m$. Different weighting matrices and horizons can be selected. Here, for simplifying the graphical representations of the results, we choose $P = M$, $\mathbf{R} = \gamma\,\mathbf{I}_u$, and $\mathbf{Q} = \mathbf{I}_y$.

The maximum eigenvalues of the corresponding closed-loop systems are computed and plotted in 3-D graphs of Figs. 2(a), 3(a), and 4(a) for different combinations of γ and P. (More degrees of freedom can be used if the achieved performances and stability are not satisfactory). The Z-axis represents the maximum eigenvalues, the X- and Y-axis represent the logarithm of γ and P, respectively. The control performance of closed-loop system is plotted in Figs. 2(b), 3(b), and 4(b), where the black dashed lines correspond to the desired outputs, the blue solid lines correspond to the system outputs and inputs using N-DMPC, and the red dashed lines represent the system outputs and inputs using proposed NC-DMPC.

The stability depends on the choice of the tuning parameters γ and P. For weak interactions [see Figs. 2(a) and 3(a)],

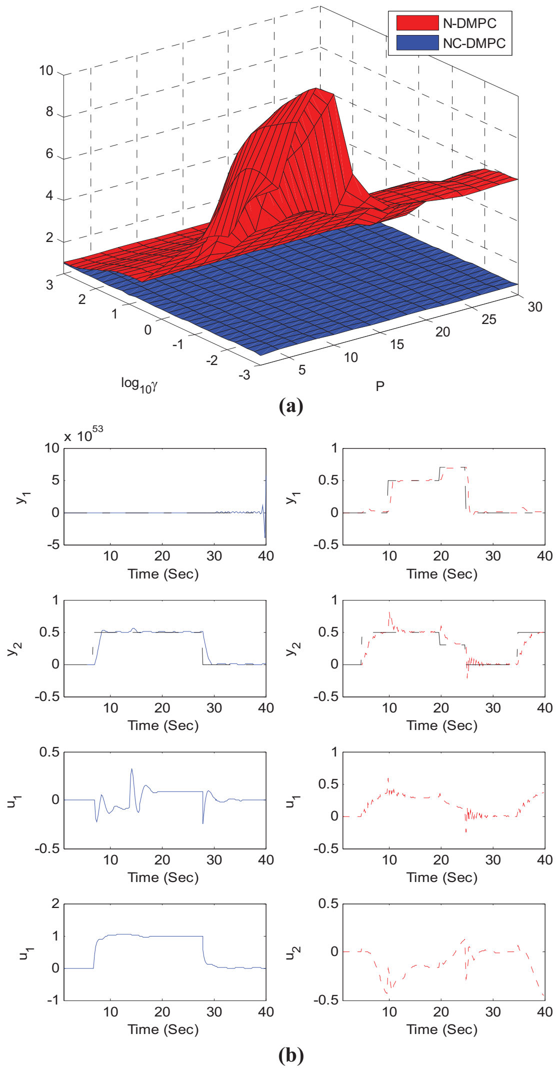

Fig. 4. Plant with $\alpha = 10$. (a) Maximum closed-loop eigenvalues with N-DMPC and NC-DMPC. (b) Control performance with $\gamma = 1$ and $p = 20$ for N-DMPC (blue line, unstable) and NC-DMPC (red line, MSE = 0.1544).

the ranges of tuning parameters in NC-DMPC are similar to those of N-DMPC. At the same time, a wider range of tuning parameters in NC-DMPC, than that in N-DMPC, is available for strong interactions [see Fig. 4(a)].

Most importantly, a better global performance of the closed-loop system is observed for the proposed NC-DMPC where the subsystems exhibit interactions [see Figs. 2(b) and 3(b)]. The MSEs of outputs with NC-DMPC are less than that using N-DMPC when $\alpha = 0.1$ and $\alpha = 1$, which are (0.2086, 0.2034) for NC-DMPC and (0.2568, 0.2277) for N-DMPC, respectively. The closed-loop system is unstable using N-DMPC, while it is stable using NC-DMPC when $\alpha = 10$, $\gamma = 1$ and $P = 20$ (see Fig. 4).

In conclusion, for the given example, NC-DMPC provides larger regions of tuning parameters than N-DMPC. Usually, the stable regions are associated with big prediction horizon P and small weight γ. The NC-DMPC can achieve a satisfactory global performance whether the interactions among subsystems are stronger or not. Furthermore, the cost of computation is very small as compared with the centralized.

VI. Conclusion

In this brief, the control for a class of large-scale system was discussed, in which the whole system is naturally divided into many small scale interacting subsystems and these subsystems interacts each other by both their states and their inputs. The NC-DMPC was proposed for improving the global performance of closed-loop system in the existence of one step network communication delay. The condition of closed-loop stability was given for local MPCs tuning and the performance of proposed methodology is discussed. Compared with the existing method, NC-DMPC could achieve an improved performance of the whole system through using global index in optimization. Further investigation might focus on designing stabilized distributed MPC, which could improve the global performance of the studied system.

Appendix A
Proof of Lemma 3

According to (7) and (9), the stacked predictions of states and outputs of $\mathcal{S}$ calculated by subsystem $\mathcal{S}_i$ at time k is

$$\begin{aligned}\hat{\mathbf{X}}^i(k+1, P\,|k) = \bar{\mathbf{S}}\Big[&\bar{\mathbf{A}}_i \mathbf{L}_i \mathbf{x}_i(k) + \bar{\mathbf{B}}_i \mathbf{U}_i(k, M\big|k) \\ &+\bar{\mathbf{A}}_a \mathbf{L}'_i \hat{\mathbf{x}}(k\big|k-1) \\ &+\tilde{\mathbf{B}}_i \mathbf{U}(k-1, M\big|k-1)\Big] \quad \text{(A.1a)}\end{aligned}$$

$$\hat{\mathbf{Y}}^i(k+1, P\,|k) = \bar{\mathbf{C}}_a \hat{\mathbf{X}}^i(k+1, P\,|k) \quad \text{(A.1b)}$$

where, the last $P - M + 1$samples of $\hat{\mathbf{U}}(k-1, P\,|k-1)$ and $\mathbf{U}_i(k, P\,|k)$ are assumed to equal the last element of $\mathbf{U}(k-1, M\,|k-1)$ and $\mathbf{U}_i(k, M\,|k)$, respectively.

By $\mathbf{u}_i(k+h\,|k) = \mathbf{u}_i(k-1) + \sum_{r=0}^{h} \Delta\mathbf{u}_i(k+r\,|k)$ and (9e), it has

$$\mathbf{U}_i(k, M\,|k) = \mathbf{\Gamma}'_i \mathbf{u}_i(k-1) + \bar{\mathbf{\Gamma}}_i \Delta\mathbf{U}_i(k, M\,|k). \quad \text{(A.2)}$$

Then the QP (10) can be deduced by substituting (9), (A.1) and (A.2) into (6).

Appendix B
Proof of Theorem 5

By (9) and (15), the stacked states prediction of $\mathcal{S}_i$ in $\mathcal{C}_i$ at time k is expressed as

$$\begin{aligned}\hat{\mathbf{X}}_i(k+1, P\big|k) = \bar{\mathbf{L}}_i^T \bar{\mathbf{S}}\Big[&\bar{\mathbf{A}}_a \mathbf{L}_i \mathbf{x}_i(k) + \bar{\mathbf{B}}_i \mathbf{U}_i(k, M\big|k) \\ &+\bar{\mathbf{A}}_i \mathbf{L}'_i \hat{\mathbf{x}}(k\big|k-1) \\ &+\tilde{\mathbf{B}}_i \mathbf{U}(k-1, M\big|k-1)\Big]. \quad \text{(B.1)}\end{aligned}$$

Through (14), it has

$$\hat{\mathbf{X}}(k, P\,|k-1) = \Omega\hat{\mathbf{X}}(k, P\,|k-1) \quad \text{(B.2a)}$$
$$\mathbf{U}(k, M\,|k-1) = \Pi\mathrm{U}(k, M\,|k-1). \quad \text{(B.2b)}$$

By (B.2), (15) and (B.1), the stacked prediction of the states of all subsystems at time k is expressed as

$$\hat{\mathbf{X}}(k+1, P\big|k) = \bar{\mathbf{L}}\mathbf{S}[\mathbf{A}\bar{\mathbf{L}}\mathbf{x}(k) + \bar{\mathbf{B}}\mathrm{U}(k, M\big|k)$$

$$+\bar{\mathbf{A}}\tilde{\mathbf{L}}\mathbf{\Omega}\hat{\mathbf{X}}(k, P\big|k-1)$$
$$+\tilde{\mathbf{B}}\mathbf{\Pi}\mathrm{U}(k-1, M\big|k-1)]. \tag{B.3}$$

Noting that $\mathbf{u}_i(k-1) = \mathbf{\Gamma}_i\mathbf{U}_i(k-1, M\,|k-1)$, by (12) and (15e), it has

$$\mathbf{U}_i(k, M\big|k) = \mathbf{\Gamma}_i'\mathbf{\Gamma}_i\mathbf{U}_i(k-1, M\big|k-1) + \bar{\mathbf{\Gamma}}_i\bar{\mathbf{K}}_i \times[\mathbf{Y}^d(k+1, P\big|k) - \hat{\mathbf{Z}}_i(k+1, P\big|k)]. \tag{B.4}$$

Substituting (11c) into (B.4), by (14), (B.2) and (15), the complete stacked open-loop optimal sequence becomes

$$\mathrm{U}(k, M\big|k) = \mathbf{\Psi}\mathrm{U}(k-1, M\big|k-1) + \Theta\mathbf{x}(k) + \Phi\hat{\mathrm{X}}(k, P\big|k-1) + \Xi\mathbf{Y}^d(k+1, P\big|k). \tag{B.5}$$

Noting that the complete feedback control law computed by all controllers is

$$\mathbf{u}(k) = \mathbf{\Gamma}\mathrm{U}(k, M\,|k). \tag{B.6}$$

Combining (2), (B.3), (B.5), and (B.6), the closed-loop state-space representation for the distributed case is derived as

$$\mathbf{x}(k) = \mathbf{A}\mathbf{x}(k-1) + \mathbf{B}\mathbf{\Gamma}\mathrm{U}(k-1, M\big|k-1) \tag{B.7a}$$
$$\hat{\mathbf{X}}(k, P\big|k-1) = \bar{\mathbf{L}}\mathbf{S}[\mathbf{A}\bar{\mathbf{L}}\mathbf{x}(k-1) + \bar{\mathbf{B}}\mathrm{U}(k-1, M\big|k-1) +\bar{\mathbf{A}}\tilde{\mathbf{L}}\mathbf{\Omega}\hat{\mathrm{X}}(k-1, P\big|k-2) +\tilde{\mathbf{B}}\mathbf{\Pi}\mathrm{U}(k-2, M\big|k-2)] \tag{B.7b}$$
$$\mathrm{U}(k, M\big|k) = \Theta[\mathbf{A}\mathbf{x}(k-1) + \mathbf{B}\Gamma\mathrm{U}(k-1, M\big|k-1)] +\Phi\bar{\mathbf{L}}\mathrm{S}[\mathbf{A}\bar{\mathrm{L}}\mathrm{x}(k-1) + \bar{\mathbf{B}}\mathrm{U}(k-1, M\big|k-1) +\bar{\mathbf{A}}\tilde{\mathbf{L}}\mathbf{\Omega}\hat{\mathrm{X}}(k-1, P\big|k-2) +\tilde{\mathbf{B}}\mathbf{\Pi}\mathrm{U}(k-2, M\big|k-2)] +\mathbf{\Psi}\mathrm{U}(k-1, M\big|k-1) + \Xi\mathrm{Y}^d(k+1, P\big|k) \tag{B.7c}$$
$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{B.7d}$$

where $\hat{x}(k\,|k)$ is substituted with $x(k)$ due to the assumption of fully accessible state.

Define the extended state as

$$\mathbf{X}_N(k) = [\mathbf{x}^T(k), \hat{\mathbf{X}}^T(k, P|k-1), \mathrm{U}^T(k, M|k), \mathrm{U}^T(k-1, M|k-1)]^T.$$

Then by (B.7), Theorem 5 is deduced.

References

[1] M. Vaccarini, S. Longhi, and M. R. Katebi, "Unconstrained networked decentralized model predictive control," *J. Process Control*, vol. 19, no. 2, pp. 328–339, 2009.

[2] Y. Zheng, S. Li, and N. Li, "Distributed model predictive control over network information exchange for large-scale systems," *Control Eng. Pract.*, vol. 19, no. 7, pp. 757–769, 2011.

[3] S. Khoo, L. Xie, and Z. Man, "Robust finite-time consensus tracking algorithm for multirobot systems," *IEEE/ASME Trans. Mech.*, vol. 14, no. 2, pp. 219–228, Apr. 2009.

[4] T. Li, M. Fu, L. Xie, and J. F. Zhang, "Distributed consensus with limited communication data rate," *IEEE Trans. Autom. Control*, vol. 56, no. 2, pp. 279–292, Feb. 2011.

[5] S. Liu, T. Li, and L. Xie, "Distributed consensus for multiagent systems with communication delays and limited data rate," *SIAM J. Control Optim.*, vol. 49, no. 6, pp. 2239–2262, 2011.

[6] J. M. Maciejowski, *Predictive Control with Constraints*. Englewood Cliffs, NJ: Prentice-Hall, 2002.

[7] A. G. Wills, D. Bates, A. J. Fleming, B. Ninness, and S. O. R. Moheimani, "Model predictive control applied to constraint handling in active noise and vibration control," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 1, pp. 3–12, Jan. 2008.

[8] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Eng. Pract.*, vol. 11, no. 7, pp. 733–764, Jul. 2003.

[9] S. Li, Y. Zheng, and B. Wang, "Steady-state target calculation for constrained predictive control systems based on goal programming," *Asia-Pacific J. Chem. Eng.*, vol. 3, no. 6, pp. 648–655, Dec. 2008.

[10] H. Peng, K. Nakano, and H. Shioya, "Nonlinear predictive control using neural nets-based local linearization ARX model-stability and industrial application," *IEEE Trans. Control Syst. Technol.*, vol. 15, no. 1, pp. 130–143, Jan. 2007.

[11] S. J. Qin and T. A. Badgwell, "An overview of nonlinear model predictive control applications," in *Nonlinear Model Predictive Control*, F. Allgower and A. Zheng, Eds. Berlin, Germany: Birkhauser, 2000, pp. 369–392.

[12] M. Xu, S. Li, and W. Cai, "Cascade generalized predictive control strategy for boiler drum level," *ISA Trans.*, vol. 44, no. 3, pp. 399–411, 2005.

[13] J. Richalet, "Industrial applications of model based predictive control," *Automatica*, vol. 29, no. 5, pp. 1251–1274, Sep. 1993.

[14] Y. Zheng, S. Li, and X. Wang, "Horizon-varying model predictive control for accelerated and controlled cooling process," *IEEE Trans. Ind. Electron.*, vol. 58, no. 1, pp. 329–336, Jan. 2011.

[15] S. Lee, S. Kumara, and N. Gautam, "Market-based model predictive control for large-scale information networks: Completion time and value of solution," *IEEE Trans. Autom. Sci. Eng.*, vol. 5, no. 4, pp. 630–640, Oct. 2008.

[16] X. Du, Y. Xi, and S. Li, "Distributed model predictive control for large-scale systems," in *Proc. Amer. Control Conf.*, 2001, pp. 3142–3143.

[17] E. Camponogara, D. Jia, B. H. Krogh, and S. Talukdar, "Distributed model predictive control," *IEEE Control Syst.*, vol. 22, no. 1, pp. 44–52, Feb. 2002.

[18] S. Li, Y. Zhang, and Q. Zhu, "Nash-optimization enhanced distributed model predictive control applied to the Shell benchmark problem," *Inf. Sci.*, vol. 170, nos. 2–4, pp. 329–349, 2005.

[19] Y. Zhang and S. Li, "Networked model predictive control based on neighbourhood optimization for serially connected large-scale processes," *J. Process Control*, vol. 17, no. 1, pp. 37–50, Jan. 2007.

[20] A. N. Venkat, I. A. Hiskens, J. B. Rawlings, and S. J. Wright, "Distributed MPC strategies with application to power system automatic generation control," *IEEE Trans. Control Syst. Technol.*, vol. 16, no. 6, pp. 1192–1206, Nov. 2008.

[21] Y. Zheng, S. Li, and X. Wang, "Distributed model predictive control for plant-wide hot-rolled strip laminar cooling process," *J. Process Control*, vol. 19, no. 9, pp. 1427–1437, Oct. 2009.

[22] A. Richards and J. P. How, "Robust distributed model predictive control," *Int. J. Control*, vol. 80, no. 9, pp. 1517–1531, 2007.

[23] B. Ding, L. Xie, and W. Cai, "Distributed model predictive control for constrained linear systems," *Int. J. Robust Nonlinear Control*, vol. 20, no. 11, pp. 1285–1298, 2010.

[24] L. Magni and R. Scattolini, "Stabilizing decentralized model predictive control of nonlinear systems," *Automatica*, vol. 42, no. 7, pp. 1231–1236, 2006.

[25] B. Ding, "Distributed robust MPC for constrained systems with polytopic description," *Asian J. Control*, vol. 13, no. 1, pp. 198–212, 2011.

[26] J. B. Rawlings and B. T. Stewart, "Coordinating multiple optimization-based controllers: New opportunities and challenges," *J. Process Control*, vol. 18, no. 9, pp. 839–845, 2008.