Contents lists available at ScienceDirect

Chemical Engineering Research and Design

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

Partitioning for distributed model predictive control of nonlinear processes

Rosiane R. Rocha[a], *Luís Cláudio Oliveira-Lopes*[b,*], *Panagiotis D. Christofides*[c,d]

[a] *Federal Institute of Espírito Santo, ES-010 km 6.5, Manguinhos, Serra, 29173-087, ES, Brazil*
[b] *School of Chemical Engineering, Federal University of Uberlândia, 2121 João Naves de Ávila Avenue, Uberlândia, 38400-902, MG, Brazil*
[c] *Department of Chemical and Biomolecular Engineering, University of California, Los Angeles, CA 90095-1592, USA*
[d] *Department of Electrical and Computer Engineering, University of California, Los Angeles, CA 90095-1592, USA*

ARTICLE INFO

Article history:
Received 14 July 2018
Received in revised form 21 August 2018
Accepted 3 September 2018
Available online 15 September 2018

Keywords:
Partitioning for distributed control
Model predictive control
Model decomposition

ABSTRACT

A distributed model predictive control (DMPC) strategy brings interesting features of topology, flexibility and maintenance to large-scale nonlinear systems. This work presents contributions in the study of distributed controllers for nonlinear and large-scale systems. Two types of distributed predictive control based on model (DMPC) are proposed: non-cooperative locally linearized DMPC and cooperative locally linearized DMPC. The decomposition is performed based on a local linearized version of the process model by using local matrices representing interactions between controlled outputs, states and inputs. The proposed strategy was successfully evaluated and compared to the centralized control strategy.

1. Introduction

The most basic requirement of a control system is that it guarantees closed-loop stability while increasing the overall process efficiency. With the great development of system analysis tools, inexpensive computer power and data for modeling and developing process identification, it is possible today to address large and complex systems in a systematic framework.

As computational numerical algorithms and computer power evolve, the type and size of potential applications also broadens. Problems that previously were considered intractable, from a computational point of view, now are solvable. However, in order to well understand the full potential of large-scale systems, especially in process control, is necessary to specify how a large-scale system is defined. A large-scale system has many different meanings in the literature. It can be used to define a system if it can be partitioned into subsystems. Of course one has to be reasonable about dimensions, otherwise a simple decomposition of a low dimensional system could be misused for large-scale system. A decomposed system can be structured as an interconnected system or as a hierarchically structured system requiring a coordinator.

Control of large-scale systems (LSS) has been long studied, particularly the issue of how to distribute control tasks in a complex large-scale process (Mesarovic et al., 2000; Sandell et al., 1978). The control of LSS requires coordination of all existing interactions among the constituent subsystems. Subsystems of a plant are usually designed independently or are added later as the installed plant evolves. These changes are usually motivated by production requirements or environmental regulations. Most LSS utilize decentralized control as the strategy of choice. However, for subsystems with strong interactions, this approach can lead to unacceptable performance. Centralized control is able to address optimally the problem of interaction, but with high structural and organizational costs, making the complex structure and the upgrade maintenance costly.

[*] *Corresponding author.*
E-mail address: lcol@ufu.br (L.C. Oliveira-Lopes).

https://doi.org/10.1016/j.cherd.2018.09.003
0263-8762/

Centralized and decentralized control are the two design extremes for the control of large-scale systems. While centralized control takes into account all possible interactions, decentralized control ignores them partially or completely. Additionally, both decentralized and centralized control require no communication between subsystems. An alternative control structure is needed, therefore, that does not have the organizational and maintenance costs of centralized control, but could give acceptable performance in large-scale, highly interacting systems.

An alternative to centralized control that preserves the topology and flexibility of decentralized control and at the same time may offer a nominal closed-loop stability guarantee is the distributed control approach. In this control structure, the interactions between subsystems are modeled and information between the subsystems is shared between them. Ignoring the structural constraints and addressing the control problem as a distributed optimization problem may lead to an unsuccessful industrial application, where it is desired high flexibility on the use of existing control structure (Rawlings and Stewart, 2008).

In order to design a controller that is able to address the issues noted above one needs to consider the following:

Partitioning How can one decompose a LSS into subsystems with known properties to address the structural coupling in the plantwide control problem?

Communication How can one design a distributed model-based control architecture by knowing the subsystems from the partitioning step above in a way of not having either a large communication burden not a large closed-loop deterioration behavior when compared to a centralized performance?

Performance How to evaluate the global properties of the distributed model based control law based on the set of subsystems?

Another important definition is the concept of complexity. A system is called complex if conventional system analysis techniques result in poor solution. A large-scale system is complex. There is also the definition based on the centrality concept. A system is of large-scale if it is not formed by subsystems grouped into a single center. There are also additional concepts such as: (a) System of System concept (SoS): it is a class of complex systems whose constituents are themselves complex. As defined by Jamshidi (2008), it is a metasystem that is comprised of multiple autonomous embedded complex systems that can be diverse in technology, context, operation, geography and conceptual frame; (b) the Enterprise Systems of Systems Engineering (SoSE) that is focused on coupling traditional systems engineering activities with enterprise activities of strategic planning and investment analysis (Carlock and Fenton, 2001), and (c) Ultra-large-scale system (ULSS): it is used in Computer Science and Software Engineering to refer to software intensive systems (Feiler et al., 2006).

Even though the above presented concepts are not universally accepted, in case of distributed process control, we are mostly interested to address standard LSS, but it can be also applied to more general systems, such as SoS. Due to the fact that one may have systems of interest of very high dimension, which are complex in nature and even might not be central at times. Therefore, because of the LSS broad range of understanding, it can be used to define systems in many different fields, such as electric power system networks, water distribution systems, manufacturing systems, communication networks and economy and management systems. In this work, we are mostly concerned with the process control view of large-scale system for distributed model predictive control.

There are three classes of models in LSS: (a) aggregated models, where a higher dimensional system is reduced by approximation to a lower dimensional system. In this case the major properties such as controllability, stability and so forth must be preserved from the original system to the reduced one; (b) perturbed models, where the reduced model neglects existing interactions in the original model by using a regular or singular approach for weak and strong coupling, respectively, and finally, (c) descriptive variable models, in which the system representation consists mainly of physical or economical variables of the system (Jamshidi, 1997). A classification of the LSS approaches can be also made according to the topology of the communication network, the different communication protocols used by the local controllers, and the cost function considered in the local controller optimization problem (Christofides et al., 2013).

Decomposition for distributed optimization has been widely used in the literature for process optimization and distributed control. Despite all of the existing research, for general problems, there is no systematic method for determining an optimal decomposition. The existing methodologies are based on: a case by case study aiming to use any special structure available to partition the system, or by using a simulation study to create blocks with small exogenous interaction; use of network science theory, such as the community detection strategy (Tang et al., 2018), to seek a strict block diagonal or block triangular structure by using network algorithms. There are two major aspects to note, either the existing methodology does not apply to a general system description, or it is based on a network distributed optimization that does not guarantee performance. The study of distributed optimization algorithms is still an open problem of research.

A communication-based MPC (Model Predictive Control) scheme was investigated by Jia and Krogh (2002) and DMPC and estimation problems are considered in Mercangöz and Doyle III (2007) for square plants perturbed by noise. Motee and Sayyar-Rodsari (2003) introduced a partitioning algorithm that uses an open-loop performance metric to partition the distributed system into subsystems balancing them against the closed-loop cost of the control actions for the overall distributed system. In Venkat et al. (2005), the authors proposed a DMPC algorithm designed for linear systems based on a process of negotiations among DMPC agents. Zhang and Wang (2012) presented a strategy for linear time invariant systems that requires decomposing the entire system into subsystems based on the control input distribution. This method, however, cannot be directly applied if the system has inputs affecting all states of the system. Zhang et al. (2013) proposed a DMPC algorithm for input-saturated polytopic uncertain systems. The subsystem controllers are obtained by optimizing a global cost function and acite min-max distributed MPC strategy was proposed for uncertain distributed systems.

It is known that linear systems are almost nonexistent in nature, but around the operating point of a system one can make use of the linearized model to predict the process behavior without significant loss in performance of the control system. LMPC (Linear Model Predictive Control) can be applied to these types of problems, where the goal is to operate them in a region around a steady-state. Some processes with a high degree of non-linearity may require the implementation of a nonlinear MPC approach (NMPC). The principle that

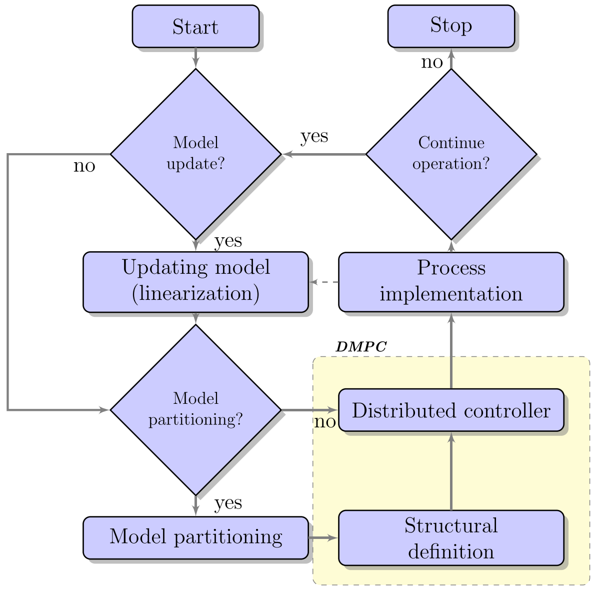


Fig. 1 – Representation of DMPC Strategies.

describes the NMPC is equivalent to the one of LMPC; the difference resides in whether the process dynamics are described with a linear or nonlinear model. One of the problems when working with NMPC approaches is that the computational demand required to calculate the control actions within the sampling period. One way to resolve this issue would be by using the controller based on a local representation of the plant model (by successive linearization). In that approach a local linearized model is obtained at each sampling instant and is used for control calculations.

The objective of this work is to develop two distributed predictive control strategies for nonlinear processes, with partitioning methodology for linear systems, which will be used after the successive linearization of the model. One of the control proposals has a non-cooperative structure and the other a cooperative structure. The novelty of this work is centered on the proposal of a new automatic partitioning of models for application in DMPC for non-linear processes. The remainder of this article is organized as follows: Section 2 presents the proposed new partitioning method and the two DMPC strategies based on this new system partitioning method are presented. Numeric examples are shown in Section 3, and Section 4 concludes the paper.

1.1. *Notation*

The operator $\|\cdot\|_Q^2$ is used to denote the square of a weighted Euclidean norm, i.e., $\|x\|_Q^2 = x^TQx$ for all $x \in \mathbb{R}^n$. The deviation variables are defined as $\overline{(\quad)}_k = (\quad)_k - (\quad)_{k-1}$. The following nomenclatures are equivalent: $x(k-1) = x_{k-1}$ and $u(k-1) = u_{k-1}$.

2. Problem formulation

This paper proposes two types of distributed control for nonlinear processes. The major difference between the proposed methodologies is the algorithm for the calculation of control actions, DMPC-1 uses a non-cooperative structure and DMPC-2 is a cooperative controller. The methodologies presented herein can be represented by the scheme of Fig. 1 and the following steps:

Step 1: if required, evaluate a locally linearized discrete-time model around the point $(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$;
Step 2: if required, apply the proposed partitioning methodology (Section 2.1);
Step 3: find the control $\mathbf{u}_k$ using a DMPC-1 (Section 2.2.1) or DMPC-2 (Section 2.2.2) algorithm;
Step 4: implement the $\mathbf{u}_k$ obtained on the nonlinear plant;
Step 5: measure/estimate the states $\mathbf{x}_k$;
Step 6: with new $\mathbf{u}_k$ and $\mathbf{x}_k$ returns to **Step 1**.

Let a process be assumed to be described by the following nonlinear ordinary differential equation system:

$$\dot{\mathbf{x}} = \mathbb{f}(\mathbf{x}, \mathbf{u}) \tag{1}$$

$$\mathbf{y} = \mathbf{g}(\mathbf{x}) \tag{2}$$

where $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{u} \in \mathbb{R}^m$ and $\mathbf{y} \in \mathbb{R}^l$ are the vector of state variables, the vector of manipulated variables and the vector of controlled variables, respectively. The problem of interest is the partitioning of this m-dimensional control problem into smaller control problems that can be better maintained and are also presented an easier solution strategy without a great loss of performance compared to the centralized control strategy for nonlinear control systems.

The local discrete-time representation of the process model can be described as (Rocha and Oliveira-Lopes, 2016a,b)

$$\bar{\mathbf{x}}(k+1) = \mathbf{A}_{k-1}\bar{\mathbf{x}}(k) + \mathbf{B}_{k-1}\bar{\mathbf{u}}(k) + \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \tag{3}$$

$$\bar{\mathbf{y}}(k) = \mathbf{C}_{k-1}\bar{\mathbf{x}}(k) + \mathbf{g}(\mathbf{x}_{k-1}) \tag{4}$$

where $\mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) = \mathrm{A_c}^{-1}(e^{\mathrm{A_c}T_s} - \mathbf{I})\mathbb{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$, T_s is the sampling time used in the model discretization, $\bar{\mathbf{x}}(k) = [\bar{x}_1 \quad \bar{x}_2 \quad \cdots \quad \bar{x}_n]^T \in \mathbb{R}^n$ is the state vector in deviation variable; $\bar{\mathbf{u}}(k) \in \mathbb{R}^m$ refers to the manipulated variable vector in deviation variable; $\bar{\mathbf{y}}(k) \in \mathbb{R}^l$ refers to the controlled output vector in deviation variable; $\mathbf{A}_{k-1} \in \mathbb{R}^{n\times n}$, $\mathbf{B}_{k-1} \in \mathbb{R}^{n\times m}$, $\mathbf{C}_{k-1} \in \mathbb{R}^{l\times n}$ are the discrete-time state, input and controlled variable matrices at time $k-1$, respectively, and $\mathbf{A}_c \in \mathbb{R}^{n\times n}$ is the continuous-time state matrix. Locally linearized versions of subsystems will be used by distributed predictive controllers that deal with plant-submodels mismatch as disturbances to the system.

2.1. *The proposed partitioning methodology*

The decomposition of the locally linearized version of the system model is based on graph theory using the effect from the input space to the controlled variables of interest. Firstly, one evaluates the output effects from each input, next, the state-output effect is analyzed. The proposed algorithm considers two levels of effects, the direct effect and the indirect effect. One needs to consider first the most dominant direct effects, and if they are not explicit in the model, then the selection goes to the indirect effect from all inputs to that specific controlled output. The partition can be evaluated at each sampling instant from the matrices $\mathbf{A}_{k-1}$, $\mathbf{B}_{k-1}$ and $\mathbf{C}_{k-1}$ of the linearized model (Eqs. (3)–(4)) or periodically according to specific process/scenario requirements. To simplify the nomenclature the index $k-1$ will be dropped in the following steps in this paper. Consider the following controllable linear time-variant (LTV) plantwide discrete-time system model given by

$$\bar{\mathbf{x}}(k+1) = \mathbf{A}\bar{\mathbf{x}}(k) + \mathbf{B}\bar{\mathbf{u}}(k) + \mathrm{A_c}^{-1}(e^{\mathrm{A_c}T_s} - \mathbf{I})\mathbb{f} \tag{5}$$

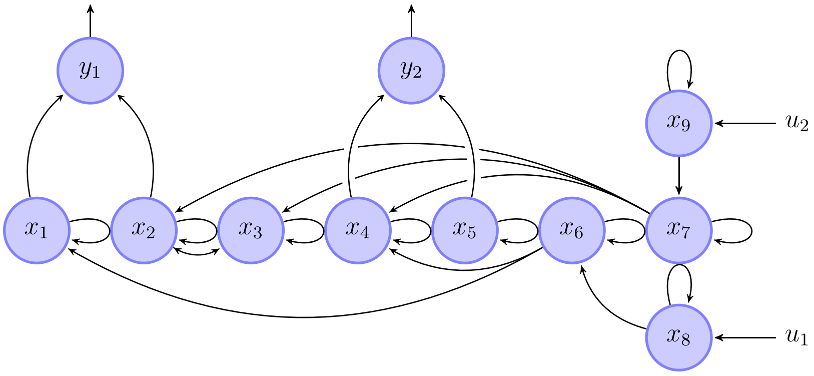
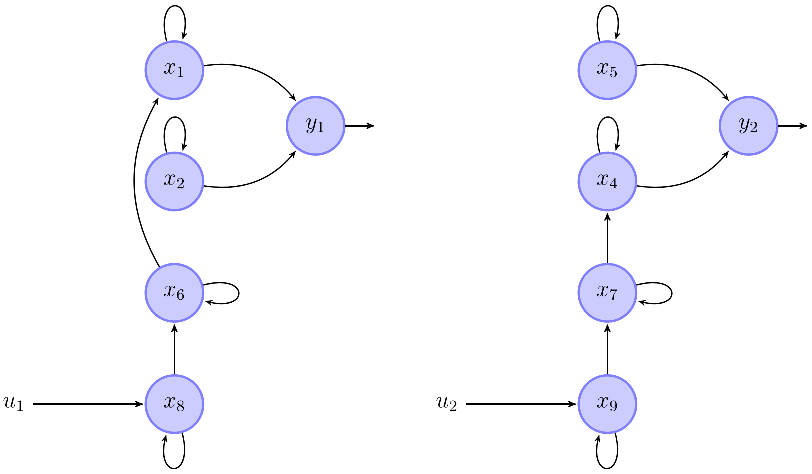


Fig. 2 – Example of a graph representation for the linear time-invariant system, System II of Section 3.2.



(a) Resulting subsystem 1 for System II of section 3.2. (b) Resulting subsystem 2 for System II of section 3.2.

Fig. 3 – Algorithm partitioning results, System II of Section 3.2.

$$\bar{\mathbf{y}}(k) = \mathbf{C}\bar{\mathbf{x}}(k) + \mathbf{g} \tag{6}$$

Consider the following definitions 1–3:

Definition 1: Define a set $\varphi_i = \{\alpha | \mathbf{B}_{\alpha i} \neq 0, \alpha = 1, \ldots, n\}$, in which $\mathbf{B}_{\alpha i}$ is the α-th row and i-th column element of the **B** matrix. The set $\vartheta_i = \{x_\alpha, \alpha \in \varphi_i\}$ contains the states directly affected by the input u_i.

Definition 2: Define a set $\xi_i = \{\alpha | \mathbf{A}_{\alpha i} \neq 0, \alpha = 1, \ldots, n\}$, in which $\mathbf{A}_{\alpha i}$ is the α-th row and i-th column element in the **A** matrix. The set $\rho_i = \{x_\alpha, \alpha \in \xi_i\}$ represents the states directly affected by the state x_i, which form the state structural matrix **S** (matrix with 1 where there is state interaction and zero otherwise).

Definition 3: Define a set π_i with the states that must be included in the submodel $M\pi_i$. Each submodel $M\pi_i$ will be defined based on the mapping of each state belonging to set ϑ_i and ρ_i affecting the output space (based on the **C** matrix).

A step-by-step algorithm for the introduced decomposition strategy is:

Step 0: build a graph representation for the matrices **A**, **B**, and **C** from the linear model and build a graph representation for it;

Step 1: create the sets φ_i and ξ_i (and the sets ϑ_i and ρ_i);

Step 2: create the set π_i with the states belonging to the sets ϑ_i and ρ_i affecting directly the states that affect the output controlled space;

Step 3: build submodel $M\pi_i$ with elements of the set π_i and the corresponding input;

Step 4: merge the $M\pi_i$ and $M\pi_j$ (for $i \neq j$) submodels if π_i and π_j are identical sets. Create a new set of inputs by merging the u_i and u_j inputs;

Step 5: check if the submodel $M\pi_i$ is controllable, otherwise π_i is augmented with elements (states) until that assumption is satisfied.

Figs. 2 and 3 illustrate the partitioning algorithm for a local representation of a linear system. The partitioning algorithm receives a graph, as indicated in the example of Fig. 2, with the system matrices and returns the partitioned subsystems, as presented in Fig. 3 with the corresponding system matrices.

The algorithm described above was implemented in the Scilab software and automatically generates controllable submodels from the matrices of the representation of the global state-space model of the process.

2.2. *Controller design*

The directly controlled states can be written in compact form as $\mathbf{x}_i$, and the rewritten plantwide system model is given by

$$\begin{aligned}\bar{\mathbf{x}}_i(k+1) = \; & \mathbf{A}_{ii}\bar{\mathbf{x}}_i(k) + \mathbf{B}_{ii}\bar{\mathbf{u}}_i(k) + \mathbf{f}_i(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) \\ & + \sum_{j=1(j \neq i)}^{M} \mathbf{A}_{ij}\bar{\mathbf{x}}_j(k) + \mathbf{B}_{ij}\bar{\mathbf{u}}_j(k)\end{aligned} \tag{7}$$

$$\bar{\mathbf{y}}_i(k) = \mathbf{C}_{ii}\bar{\mathbf{x}}_i(k) + \mathbf{g}_i(\mathbf{x}_{k-1}) + \sum_{j=1(j \neq i)}^{M} \mathbf{C}_{ij}\bar{\mathbf{x}}_j(k) \tag{8}$$

where $i = 1, \ldots, M$; $\overline{\mathbf{y}}_i(k) \in \mathbb{R}^{l_i}$, $\overline{\mathbf{x}}_i(k) \in \mathbb{R}^{n_i}$, $\overline{\mathbf{u}}_i(k) \in \mathbb{R}^{m_i}$ are output, state and input vectors of subsystem modeled by $M\pi_i$ at time k, respectively; $\overline{\mathbf{x}}_j(k)$, $\overline{\mathbf{u}}_j(k)$ are the state and input vector of subsystem modeled by $M\pi_j$, respectively. Herein all matrices are evaluated at time $k-1$.

In this paper, the interrelationships among subsystems are usually treated as disturbances. Let

$$\mathbf{d}_i(k) = \sum_{j=1(j \neq i)}^{M} \mathbf{A}_{ij}\overline{\mathbf{x}}_j(k) + \mathbf{B}_{ij}\overline{\mathbf{u}}_j(k) \tag{9}$$

$$\mathbf{h}_i(k) = \sum_{j=1(j \neq i)}^{M} \mathbf{C}_{ij}\overline{\mathbf{x}}_j(k) \tag{10}$$

The i-th distributed system model in Eqs. (3)–(4) can be transformed into the following form

$$\overline{\mathbf{x}}_{d,i}(k+1) = \mathbf{A}_{ii}\overline{\mathbf{x}}_{d,i}(k) + \mathbf{B}_{ii}\overline{\mathbf{u}}_i(k) + \mathbf{f}_i(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{d}_i(k) \tag{11}$$

$$\overline{\mathbf{y}}_{d,i}(k) = \mathbf{C}_{ii}\overline{\mathbf{x}}_{d,i}(k) + \mathbf{g}_i(\mathbf{x}_{k-1}) + \mathbf{h}_i(k), \quad i = 1, 2, \ldots, M \tag{12}$$

where $\mathbf{d}_i(k)$ and $\mathbf{h}_i(k)$ represent disturbances from other subsystems and $\mathbf{f}_i(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ and $\mathbf{g}_i(\mathbf{x}_{k-1})$ represent the values of the functions $\mathbf{f}$ and $\mathbf{g}$ at the points $(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ and $(\mathbf{x}_{k-1})$ belonging to the subsystem i. Again, the index $k-1$ will not be displayed in the equations in this section, but one must remember that the local representation of the state-space model was obtained by the linearization process around the $(\mathbf{x}_{k-1}, \mathbf{u}_{k-1})$ condition. For each subsystem, the distributed MPC problem is cast into the problem of designing a robust MPC control law. It is assumed that the nominal model of (9)–(10) can be described as

$$\overline{\mathbf{x}}_{n,i}(k+1) = \mathbf{A}_{ii}\overline{\mathbf{x}}_{n,i}(k) + \mathbf{B}_{ii}\overline{\mathbf{u}}_{n,i}(k) \tag{13}$$

$$\overline{\mathbf{y}}_{n,i}(k) = \mathbf{C}_{ii}\overline{\mathbf{x}}_{n,i}(k), \quad i = 1, 2, \ldots, M \tag{14}$$

Predictions of the controlled outputs are obtained as follows

$$\begin{aligned}\overline{\mathbf{Y}}_i(k|k) = {} & \boldsymbol{\Omega}_i\boldsymbol{\Psi}_i\overline{\mathbf{x}}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Theta}_i\boldsymbol{\Delta}\overline{\mathbf{U}}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Upsilon}_i\overline{\mathbf{u}}_i(k-1|k) \\ & + \boldsymbol{\Omega}_i\boldsymbol{\Xi}_i\mathbf{d}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Gamma}_i\mathbf{f}_i(k-1|k) + \boldsymbol{\Phi}_i\mathbf{g}_i(k|k) + \boldsymbol{\Phi}_i\mathbf{h}_i(k|k)\end{aligned} \tag{15}$$

where

$$\boldsymbol{\Psi}_i = \begin{bmatrix} \mathbf{A}_{ii} \\ \mathbf{A}_{ii}^2 \\ \vdots \\ \mathbf{A}_{ii}^{H_{p_i}} \end{bmatrix}, \boldsymbol{\Theta}_i = \begin{bmatrix} \mathbf{B}_{ii} & \cdots & 0 \\ \mathbf{A}_{ii}\mathbf{B}_{ii} + \mathbf{B}_{ii} & \cdots & 0 \\ \vdots & \ddots & \vdots \\ \sum_{j=0}^{H_{p_i}-1} \mathbf{A}_{ii}^j\mathbf{B}_{ii} & \cdots & \sum_{j=0}^{H_{p_i}-H_{u_i}} \mathbf{A}_{ii}^j\mathbf{B}_{ii} \end{bmatrix}, \tag{16}$$

$$\boldsymbol{\Upsilon}_i = \begin{bmatrix} \mathbf{B}_{ii} \\ \mathbf{A}_{ii}\mathbf{B}_{ii} + \mathbf{B}_{ii} \\ \vdots \\ \mathbf{A}_{ii}^{H_{pi}-1}\mathbf{B}_{ii} + \cdots + \mathbf{B}_{ii} \end{bmatrix}, \boldsymbol{\Xi}_i = [\begin{matrix} \mathbf{C}_{ii} \\ \mathbf{A}_{ii}\mathbf{C}_{ii} + \mathbf{C}_{ii} \\ \vdots \\ \mathbf{A}_{ii}^{H_{pi}-1}\mathbf{C}_{ii} + \cdots + \mathbf{C}_{ii} \end{matrix}], \tag{17}$$

$$\boldsymbol{\Gamma}_i = \begin{bmatrix} \mathbf{I}_{li} \\ \mathbf{A}_{ii} + \mathbf{I}_{li} \\ \vdots \\ \mathbf{A}_{ii}^{H_{pi}-1} + \cdots + \mathbf{I}_{li} \end{bmatrix}, \boldsymbol{\Omega}_i = \begin{bmatrix} \mathbf{C}_{ii} & 0 & \cdots & 0 \\ 0 & \mathbf{C}_{ii} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \mathbf{C}_{ii} \end{bmatrix}$$

$$\text{and} \quad \boldsymbol{\Phi}_i = \begin{bmatrix} \mathbf{I}_{li} \\ \mathbf{I}_{li} \\ \vdots \\ \mathbf{I}_{li} \end{bmatrix}. \tag{18}$$

2.2.1. DMPC-1 – Non-cooperative locally linearized DMPC

In the DMPC-1 case, each controller optimizes a local objective function. The cost function provided is composed of local controlled outputs and manipulated variables, which can be represented as

$$\begin{aligned} J_i(k) = {} & \sum_{j=H_{w_i}}^{H_{p_i}} ||\overline{\mathbf{y}}_i(k+j|k) - \mathbf{r}_{y_i}(k+j)||^2_{\mathbf{Q}_i(j)} \\ & + \sum_{j=0}^{H_{u_i}-1} ||\Delta\overline{\mathbf{u}}_i(k+j|k)||^2_{\mathbf{R}_i(j)} \\ & + \sum_{j=0}^{H_{u_i}-1} ||\overline{\mathbf{u}}_i(k+j|k) - \mathbf{r}_{u_i}(k+j)||^2_{\mathbf{W}_i(j)} \end{aligned} \tag{19}$$

where $\mathbf{Q}_i > 0$, $\mathbf{R}_i \geq 0$ and $\mathbf{W}_i \geq 0$ are the weighting matrices for the states, variations of inputs and inputs, respectively; $\mathbf{r}_{y_i}$ and $\mathbf{r}_{u_i}$ are the reference trajectories for the controlled outputs and the manipulated inputs, respectively. The cost function is penalized in the prediction horizon in the range $H_{w_i} \leq j \leq H_{p_i}$. The objective is to design a DMPC algorithm for computing an input sequence based on the system states and inputs. The control inputs obtained can guarantee closed-loop asymptotic convergence to the origin of the state and the local cost function is minimal. The proposed DMPC controller is solved (sequentially or in a parallel framework) by the following optimization problem:

$$\begin{aligned} & \min_{u_i(k+j|k), j=0,\ldots,H_{u_i}-1} J_i(k) \\ & \text{s.t.} \\ & \overline{\mathbf{Y}}_i(k|k) = \boldsymbol{\Omega}_i\boldsymbol{\Psi}_i\overline{\mathbf{x}}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Theta}_i\boldsymbol{\Delta}\overline{\mathbf{U}}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Upsilon}_i\overline{\mathbf{u}}_i(k-1|k) + \\ & \boldsymbol{\Omega}_i\boldsymbol{\Xi}_i\mathbf{d}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Gamma}_i\mathbf{f}_i(k-1|k) + \boldsymbol{\Phi}_i\mathbf{g}_i(k|k) + \boldsymbol{\Phi}_i\mathbf{h}_i(k|k) \\ & \mathbf{x}_i(k|k) = \mathbf{x}_i(k) \\ & \mathbf{x}_i(0) = \mathbf{x}_{i0} \\ & \mathbf{u}_i(k+j|k) \in \boldsymbol{\Lambda}, j = 0, \ldots, H_{ui} - 1 \end{aligned} \tag{20}$$

where $\boldsymbol{\Lambda}$ is the available input space. Based on the receding horizon strategy, only the first control action is implemented on the plant. So if the number of plant inputs is m, then only the first m rows of the vector $\Delta\overline{\mathbf{U}}_i(k)_{opt}$. One can represent this as

$$\Delta\overline{\mathbf{u}}_i(k)_{opt} = [\mathbf{I}_m, \quad \mathbf{0}_m, \quad \ldots, \quad \mathbf{0}_m]\Delta\overline{\mathbf{U}}_i(k)_{opt} \tag{21}$$

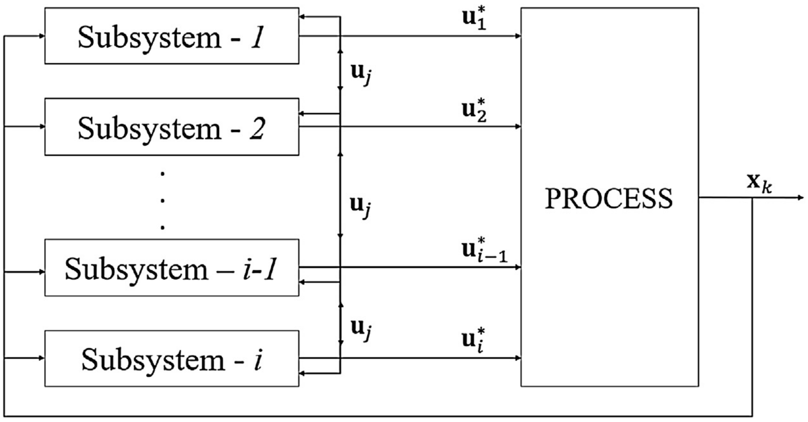


Fig. 4 – Structure of the cooperation process.

where $\mathbf{I}_m$ is the $m \times m$ identity matrix, and $\mathbf{0}_m$ is the $m \times m$ zero matrix.

2.2.2. *DMPC-2 – Cooperative locally linearized DMPC*

The process of cooperation in the calculation of the control actions of a distributed control structure allows the exchange of information between the subsystems during the same instant of sampling. Information from other subsystems is present for each controller.

In this section, the cooperative structure used in the DMPC control proposal is presented in detail. Fig. 4 presents a simplified representation of this type of approach.

In this type of control, there is a controller responsible for decision making in each subsystem i. Each subsystem i consists of l_i controlled outputs and m_i manipulated inputs. Remember that the selection of these variables is done by partitioning.

At each sampling instant, all controllers receive the values of the measurable/estimated states ($\mathbf{x}_k$) from the plant and begin an iterative process to determine their optimal trajectories of their control actions. This step is performed through a local objective function, to have an initial estimate of the actions to be calculated. This stage characterizes the first iteration. Once the initial estimation is done, the subsystems exchange information between them and the second iteration begins, in which a global objective function is used, which includes the effects of all control actions, including those of the other subsystems. Each controller is responsible for including the effects of its inputs on the overall objective function. The information exchanged between subsystems allows each controller to know the models and control actions generated by the other plant controllers. In the cooperative process, in each iteration, each controller uses the values of the actions of the other controllers obtained in the previous iteration and their respective models to optimize their own control trajectory. When the terminal condition is satisfied, ending the iterative process, each controller sends its optimized control actions to the actuators present in the plant. In Fig. 4, the optimum trajectory reached by each controller is identified by an overwritten * (asterisk).

The methodology of the cooperative process is schematized as follows:

1. In the first iteration ($c = 1$):
   (a) Each controller receives the measurable or estimated values of the states ($\mathbf{x}_k$);
   (b) Each controller calculates its respective trajectories of its control actions using a local objective function;
   (c) The trajectories calculated by each controller are shared between them;
2. In subsequent iterations ($c > 1$):
   (a) Each controller calculates a new trajectory of its control actions using a global objective function and the previously calculated trajectories of its neighbors;
   (b) The convergence condition is evaluated: If satisfied, each controller sends the calculated optimal actions ($\mathbf{u}^*$) to the actuators; if not satisfied, return to **Step 2** ($c \longleftarrow c + 1$).

The local optimization problem used by each controller in the first iteration is identical to that used in the non-cooperative linear DMPC proposal. The optimization problem, can be solved sequentially or in a parallel framework, as well as the overall objective function used in the other iterations as presented below:

$$
\begin{aligned}
&\min_{u_i(k+j|k),j=0,\ldots,H_{ui-1}} J(k) = \sum_i J_i(k)\\
&\text{s.t.}\\
&\overline{\mathbf{Y}}_i(k|k) = \boldsymbol{\Omega}_i \boldsymbol{\Psi}_i \overline{\mathbf{x}}_i(k|k) + \boldsymbol{\Omega}_i \boldsymbol{\Theta}_i \boldsymbol{\Delta} \overline{\mathbf{U}}_i(k|k) + \boldsymbol{\Omega}_i \boldsymbol{\Upsilon}_i \overline{\mathbf{u}}_i(k-1|k) +\\
&\boldsymbol{\Omega}_i \boldsymbol{\Xi}_i \mathbf{d}_i(k|k) + \boldsymbol{\Omega}_i \boldsymbol{\Gamma}_i \mathbf{f}_i(k-1|k) + \boldsymbol{\Phi}_i \mathbf{g}_i(k|k) + \boldsymbol{\Phi}_i \mathbf{h}_i(k|k)\\
&\mathbf{x}_i(k|k) = \mathbf{x}_i(k)\\
&\mathbf{x}_i(0) = \mathbf{x}_{i0}\\
&\mathbf{x}(k+j|k) \in X, j > 0\\
&\mathbf{x}(k+H_u|k) \in X_f,\\
&\mathbf{u}_i(k+j|k) \in \Lambda_i, j = 0, \ldots, H_{ui-1},\\
&\mathbf{u}_l(k+j|k) = \mathbf{u}_l(k+j|k)^{c-1}, l \neq i
\end{aligned}
\tag{22}
$$

with

$$
\begin{aligned}
J_i(k) \quad = \quad & \sum_{j=H_{w_i}}^{H_{p_i}} ||\overline{\mathbf{y}}_i(k+j|k) - \mathbf{r}_{y_i}(k+j)||^2_{\mathbf{Q}_i(j)} +\\
& \sum_{j=0}^{H_{u_i}-1} ||\overline{\mathbf{u}}_i(k+j|k) - \mathbf{r}_{u_i}(k+j)||^2_{\mathbf{W}_i(j)} +\\
& \sum_{j=0}^{H_{u_i}-1} ||\Delta \overline{\mathbf{u}}_i(k+j|k)||^2_{\mathbf{R}_i(j)} + ||\mathbf{x}(k+H_{ui})||^2_{\mathbf{P}_i}
\end{aligned}
\tag{23}
$$

Stability properties are guaranteed by the terminal assembly $X_f \subseteq X$ and the terminal cost $V_f = ||\mathbf{x}(k+H_{ui})||^2_{\mathbf{P}_i}$. For the unrestricted scenario it is desirable to stabilize the system with the control law $\mathbf{u}(k) = \mathbf{K}\cdot\mathbf{x}(k)$, this is, $\mathbf{A}+\mathbf{BK}$ is stable. In this work the gain $\mathbf{K}$ is the solution of a quadratic linear control (LQ) problem with infinite horizon with the same weights $\mathbf{Q}$ and $\mathbf{R}$ used in the last equation. So if $\mathbf{P}$ is the solution of the Lyapunov equation:

$$(\mathbf{A}+\mathbf{BK})^T\mathbf{P}(\mathbf{A}+\mathbf{BK}) - \mathbf{P} = -(\mathbf{Q}+\mathbf{K}^T\mathbf{RK}) \tag{24}$$

it is possible the set $V_f = \mathbf{x}^T\mathbf{Px}$ and $X_f = \left\{\mathbf{X}|\mathbf{x}^T\mathbf{Px} \le d\right\}$, wherein d is a small positive value chosen for $\mathbf{u}(k) = \mathbf{K}\cdot\mathbf{x}(k)$ to any $\mathbf{x} \in X_f$.

2.3. *Partitioning linear systems*

The discrete time representation of a linear time-invariant process can be described as

$$\overline{\mathbf{x}}(k+1) = \mathbf{A}\overline{\mathbf{x}}(k) + \mathbf{B}\overline{\mathbf{u}}(k) \tag{25}$$

$$\overline{\mathbf{y}}(k) = \mathbf{C}\overline{\mathbf{x}}(k) \tag{26}$$

where $\overline{\mathbf{x}}(k) = [\overline{x}_1 \quad \overline{x}_2 \quad \cdots \quad \overline{x}_n]^T \in \mathbb{R}^n$ is the state vector in deviation variable; $\overline{\mathbf{u}}(k) \in \mathbb{R}^m$ refers to the manipulated variable vector in deviation variable; $\overline{\mathbf{y}}(k) \in \mathbb{R}^l$ refers to the controlled output vector in deviation variable; $\mathbf{A} \in \mathbb{R}^{n\times n}$, $\mathbf{B} \in \mathbb{R}^{n\times m}$, $\mathbf{C} \in \mathbb{R}^{l\times n}$ are the discrete-time state, input and controlled variable matrices, respectively.

The i-th distributed system model for the linear plant can be transformed into the following form

$$\overline{\mathbf{x}}_{d,i}(k+1) = \mathbf{A}_{ii}\overline{\mathbf{x}}_{d,i}(k) + \mathbf{B}_{ii}\overline{\mathbf{u}}_i(k) + \mathbf{d}_i(k) \tag{27}$$

$$\overline{\mathbf{y}}_{d,i}(k) = \mathbf{C}_{ii}\overline{\mathbf{x}}_{d,i}(k) + \mathbf{h}_i(k), \quad i = 1, 2, \ldots, M \tag{28}$$

where $\mathbf{d}_i(k)$ and $\mathbf{h}_i(k)$ represent, as before, disturbances from other subsystems. For each subsystem, the distributed MPC problem is cast into the problem of designing a control law. It is assumed that the nominal model of (27) and (28) can be described as

$$\overline{\mathbf{x}}_{n,i}(k+1) = \mathbf{A}_{ii}\overline{\mathbf{x}}_{n,i}(k) + \mathbf{B}_{ii}\overline{\mathbf{u}}_{n,i}(k) \tag{29}$$

$$\overline{\mathbf{y}}_{n,i}(k) = \mathbf{C}_{ii}\overline{\mathbf{x}}_{n,i}(k), \quad i = 1, 2, \ldots, M \tag{30}$$

Predictions of controlled outputs are obtained as follows

$$\begin{aligned}\overline{\mathbf{Y}}_i(k|k) = \quad & \boldsymbol{\Omega}_i\boldsymbol{\Psi}_i\overline{\mathbf{x}}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Theta}_i\boldsymbol{\Delta}\overline{\mathbf{U}}_i(k|k) + \boldsymbol{\Omega}_i\boldsymbol{\Upsilon}_i\overline{\mathbf{u}}_i(k-1|k) + \\ & \boldsymbol{\Omega}_i\boldsymbol{\Xi}_i\mathbf{d}_i(k|k) + \boldsymbol{\Phi}_i\mathbf{h}_i(k|k)\end{aligned} \tag{31}$$

where the matrices have representations similar to these presented previously.

For the case of a linear time-invariant system one can also show the closed-loop stability of the model predictive controller using partitioned submodels.

2.3.1. *Closed-loop stability analysis*

For the linear time-invariant case, the closed-loop stability of the controller is evaluated following the work by Zhang and Wang (2012). For the control based on the submodels, the stability of the closed-loop plant can be described according to the following theorem:

Theorem: The plantwide closed-loop system satisfies the quadratic Lyapunov stability condition if there exists a sequence of positive-definite matrices $\mathbf{P}_i = \mathbf{P}_i^T$ such that

$$\mathbf{A}_{CL_i}^T\mathbf{P}_i\mathbf{A}_{CL_i} - \mathbf{P}_i < 0 \tag{32}$$

for all subsystems $i \in \{1, \ldots, M\}$. Then,

$$\begin{cases}\mathbf{A}_{CL_1}^T\mathbf{P}_1\mathbf{A}_{CL_1} - \mathbf{P}_1 < 0 \\ \mathbf{A}_{CL_2}^T\mathbf{P}_2\mathbf{A}_{CL_2} - \mathbf{P}_2 < 0 \\ \ldots\ldots\ldots\ldots\ldots \\ \mathbf{A}_{CL_M}^T\mathbf{P}_M\mathbf{A}_{CL_M} - \mathbf{P}_M < 0\end{cases} \tag{33}$$

The condition described by the system matrix inequalities in Equation (33) guarantees the asymptotic stability of each of the subsystems. However, using similar arguments, it is possible to prove the asymptotic stability of the overall closed-loop system. If there exist $\mathbf{P}_1, \mathbf{P}_2, \ldots, \mathbf{P}_M$ satisfying condition (33), then the following conditions can be obtained

$$\begin{cases}V_1(k+1) - V_1(k) < 0 \\ V_2(k+1) - V_2(k) < 0 \\ \ldots\ldots\ldots\ldots\ldots \\ V_M(k+1) - V_M(k) < 0\end{cases} \tag{34}$$

Due to the convexity of V_i, $i = 1, \ldots, M$, one can add them to originating a stability condition for the overall system as $V(k+1) - V(k) < 0$. The closed-loop system is obtained by combining the state-affine model the MPC controller equations into the following equation:

$$\begin{bmatrix}\hat{\mathbf{x}}_{d,i}(k+1) \\ \hat{\mathbf{x}}_i(k+1)\end{bmatrix} = \mathbf{A}_{CL_i}\begin{bmatrix}\hat{\mathbf{x}}_{d,i}(k) \\ \hat{\mathbf{x}}_i(k)\end{bmatrix} \tag{35}$$

where,

$$\mathbf{A}_{CL_i} = \begin{bmatrix}\mathbf{A}_{ii} - \mathbf{B}_{ii}\mathbf{K}_i(\boldsymbol{\psi}_{ii} + \mathbf{L}_i) & \mathbf{B}_{ii}\mathbf{K}_i\mathbf{L}_i \\ -\mathbf{B}_{ii}\mathbf{K}_i\mathbf{L}_i & \mathbf{A}_{ii} - \mathbf{B}_{ii}\mathbf{K}_i(\boldsymbol{\psi}_{ii} - \mathbf{L}_i)\end{bmatrix} \tag{36}$$

Using Lyapunov arguments, the plantwide close-loop system is asymptotic stable if there exist a positive-definite symmetric matrix $\mathbf{P}$ and a positive-definite quadratic Lyapunov function V for all initial conditions.

3. Applications

In this section, dynamic simulations are carried out to evaluate the performance of the proposed DMPC controllers. The order of the subsystems in the calculation of the control actions (structural control definition) in the distributed strategy was the same order obtained by the proposed partitioning. Since the dimension of the toy example explored are not large to gain full benefit of the strategy, in this work was chosen not to find the best sequence for these controllers nor to evaluate the parallel implementation of the calculation in the processing time.

This section presents two illustrative applications for the proposed methodology: (i) two reactors in series with a separator and recycle (case study I: OP1A and OP1B), and (ii) a flowsheet of an industrial grinding system (case study II) proposed by Ylinen et al. (1987).

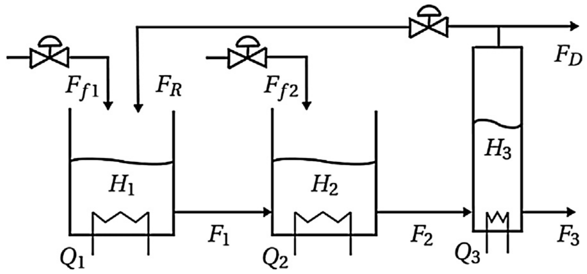


Fig. 5 – Two reactors in series with a separator and recycle (Stewart et al. (2010)).

Table 1 – Process variables.

H_1, H_2, H_3	Liquid heights in each equipment
x_{A1}, x_{B1}, x_{C1}	Percent compositions of A, B and C in the first reactor
x_{A2}, x_{B2}, x_{C2}	Percent compositions of A, B and C in the second reactor
x_{A3}, x_{B3}, x_{C3}	Percent compositions of A, B and C on the bottom of the separator
x_{AR}, x_{BR}, x_{CR}	Percent compositions of A, B and C at the top of the separator
x_{A0}, x_{B0}	Percent compositions of A and B in the feed flows
F_{f1}, F_{f2}	Feed flow rates of pure reagent A in reactors
T_0	Temperature of feed flows
T_R	Temperature of the recycle flow
F_1, F_2, F_3	Effluent flow rates in each equipment
F_D	Effluent flow rate of the process
F_R	Recycle flow rate exiting the separator to the first reactor
ρ	Density of the reaction mixture
c_p	Specific heat of the reaction mixture
ΔH_A, ΔH_B	Heat of reactions
A_1, A_2, A_3	Cross-sectional area of each equipment
T_1, T_2, T_3	Temperatures in each equipment
Q_1, Q_2, Q_3	External heat/coolant inputs to each equipment
k_{A1}, k_{B1}	Reaction constants in the first reactor
k_{A2}, k_{B2}	Reaction constants in the second reactor

3.1. *System I: Two reactors in series with a separator and recycle*

In this example, consider a plant shown in Fig. 5 presented in Stewart et al. (2010). This plant consists of two reactors in series, where a first-order reaction occurs that converts the reactant A to the product B. In the reactors, another first-order reaction occurs, in which the desired product B is lost to a side product C. In series with the second reactor is a flash separator, where the distillate is split and partially redirected to the first reactor.

The model for this system is:

$$\frac{dH_1}{dt} = \frac{1}{\rho A_1}(F_{f1} + F_R - F_1) \tag{37}$$

$$\frac{dx_{A1}}{dt} = \frac{1}{\rho A_1 H_1}[F_{f1}(x_{A0} - x_{A1}) + F_R(x_{AR} - x_{A1})] - K_{A1}x_{A1} \tag{38}$$

$$\begin{aligned}\frac{dx_{B1}}{dt} = {} & \frac{1}{\rho A_1 H_1}[F_{f1}(x_{B0} - x_{B1}) + F_R(x_{BR} - x_{B1})] + K_{A1}x_{A1} \\ & -K_{B1}x_{B1}\end{aligned} \tag{39}$$

Table 2 – Parameters values of the model for System I

Parameter	Value	Units	Parameter	Value	Units
c_P	5.0	kJ/kg K	T_0	313.0	K
x_{A0}	1.0	%	x_{B0}	0.0	%
ΔH_A	−200.0	kJ/kg	ΔH_B	−50.0	kJ/kg
k_A	2.0	1/*min*	K_B	0.18	1/min
System Case IA					
A_1, A_2	0.02	m^2	A_3	0.01	m^2
ρ	600.0	kg/m^3	K_{v1}, K_{v2}, K_{v3}	8.5	kg/m s
E_A/R	100.0	K	E_B/R	150.0	K
α_A, α_B	1.0	–	α_C	2.0	–
System Case IB					
A_1, A_2	3.0	m^2	A_3	1.0	m^2
ρ	1000.0	kg/m^3	K_{v1}, K_{v2}, K_{v3}	1020	kg/m s
E_A/R	300.0	K	E_B/R	250.0	K
α_A, α_B	1.5	–	α_C	2.0	–

Table 3 – Stationary conditions for System I.

	$OP1A_{ss1}$	$OP1A_{ss2}$	$OP1B_{ss1}$	$OP1B_{ss2}$	Units
H_1	0.706	0.423	0.706	0.423	m
x_{A1}	0.233	0.194	0.210	0.177	
x_{B1}	0.604	0.570	0.626	0.650	
T_1	345.516	348.246	348.107	350.958	K
H_2	1.176	0.541	1.176	0.541	m
x_{A2}	0.173	0.118	0.150	0.101	
x_{B2}	0.626	0.598	0.597	0.605	
T_2	348.250	352.015	350.660	354.553	K
H_3	0.939	0.351	0.939	0.351	m
x_{A3}	0.194	0.134	0.200	0.142	
x_{B3}	0.701	0.677	0.796	0.851	
T_3	348.300	352.407	350.670	354.651	K
F_{f1}	4.000	6.000	480.000	240.000	kg/min
Q_1	2.000	6.000	60.000	180.000	kJ/min
F_{f2}	4.000	1.000	480.000	120.000	kg/min
Q_2	2.000	9.000	60.000	210.000	kJ/min
F_R	2.000	1.600	240.000	192.000	kg/min
Q_3	2.000	9.000	60.000	270.000	kJ/min

Table 4 – Input constraints for System I.

Input u_i	u_{min} OP1A, B	u_{max} OP1A	u_{max} OP1B	Units
F_{f1}	0	10	500	kg/min
Q_1	0	10	500	kJ/min
F_{f2}	0	10	500	kg/min
Q_2	0	10	500	kJ/min
F_R	0	10	500	kg/min
Q_3	0	10	500	kJ/min

$$\begin{aligned}\frac{dT_1}{dt} = {} & \frac{1}{\rho A_1 H_1}[F_{f1}(T_0 - T_1) + F_R(T_R - T_1)] \\ & -\frac{1}{c_P}(K_{A1}x_{A1}\Delta H_A + K_{B1}x_{B1}\Delta H_B) + \frac{Q_1}{\rho A_1 c_P H_1}\end{aligned} \tag{40}$$

$$\frac{dH_2}{dt} = \frac{1}{\rho A_2}(F_{f2} + F_1 - F_2) \tag{41}$$

$$\frac{dx_{A2}}{dt} = \frac{1}{\rho A_2 H_2}[F_{f2}(x_{A0} - x_{A2}) + F_1(x_{A1} - x_{A2})] - K_{A2}x_{A2} \tag{42}$$

$$\begin{aligned}\frac{dx_{B2}}{dt} = {} & \frac{1}{\rho A_2 H_2}[F_{f2}(x_{B0} - x_{B2}) + F_R(x_{B1} - x_{B2})] + K_{A2}x_{A2} \\ & -K_{B2}x_{B2}\end{aligned} \tag{43}$$

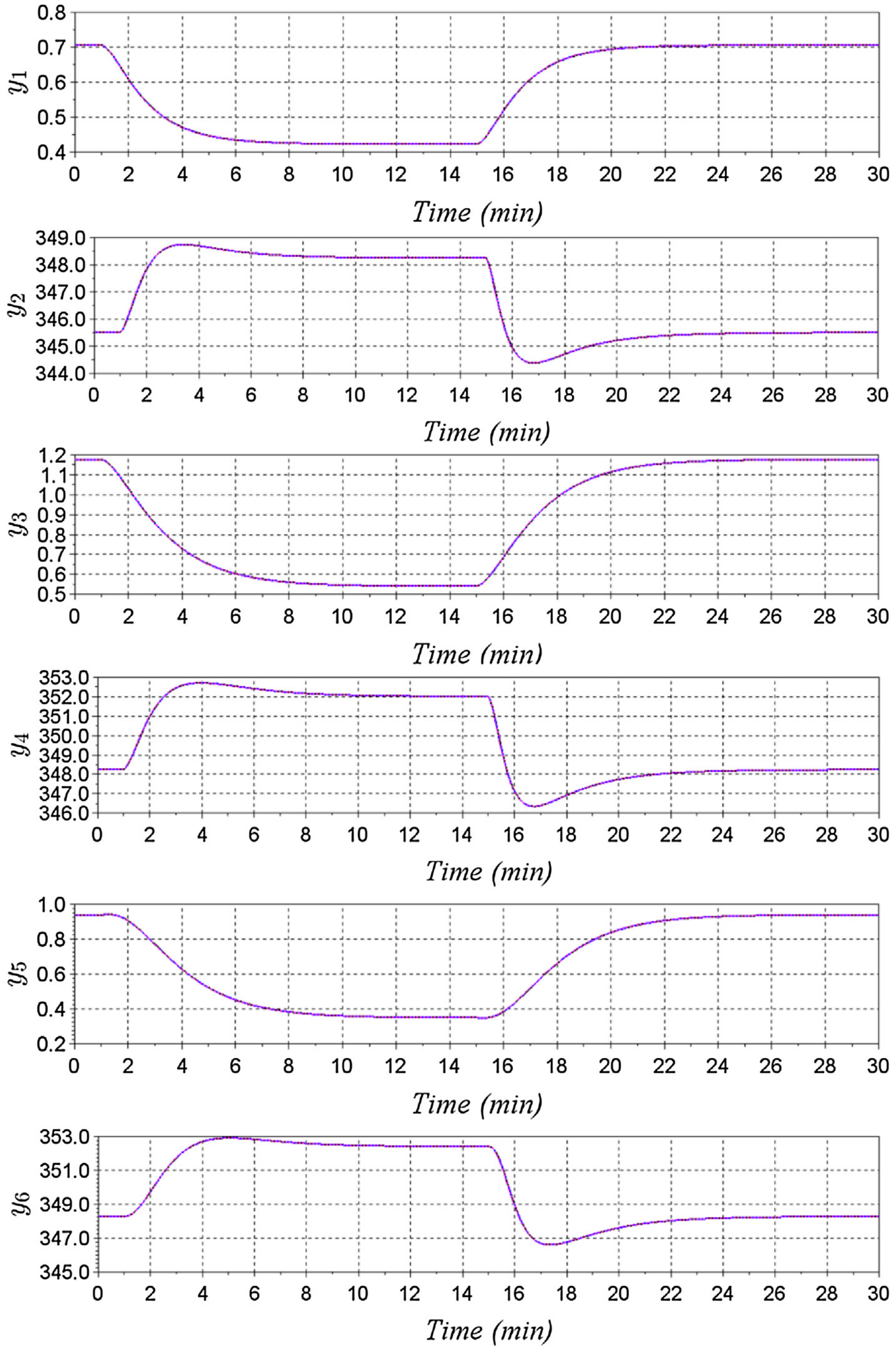


Fig. 6 – Dynamic evolution of controlled outputs for OP1A obtained by: centralized control (black solid line), DMPC-1 (blue dashed dot line) and DMPC-2 (magenta dotted line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

$$\begin{aligned}\frac{dT_2}{dt} &= \frac{1}{\rho A_2 H_2}[F_{f2}(T_0 - T_2) + F_1(T_1 - T_2)] \\ &\quad - \frac{1}{c_P}(K_{A2}x_{A2}\Delta H_A + K_{B2}x_{B2}\Delta H_B) + \frac{Q_2}{\rho A_2 c_P H_2}\end{aligned} \qquad (44)$$

$$\frac{dH_2}{dt} = \frac{1}{\rho A_3}(F_2 - F_D - F_R - F_3) \qquad (45)$$

$$\frac{dx_{A3}}{dt} = \frac{1}{\rho A_3 H_3}[F_2(x_{A2} - x_{A3}) + F_D(x_{A3} - x_{AR}) + F_R(x_{A3} - x_{AR})] \qquad (46)$$

$$\frac{dx_{B3}}{dt} = \frac{1}{\rho A_3 H_3}[F_2(x_{B2} - x_{B3}) + F_D(x_{B3} - x_{BR}) + F_R(x_{B3} - x_{BR})] \qquad (47)$$

$$\begin{aligned}\frac{dT_3}{dt} &= \frac{1}{\rho A_3 H_3}[F_2(T_2 - T_3) + F_D(T_3 - T_R) + F_R(T_3 - T_R)] \\ &\quad + \frac{Q_3}{\rho A_3 c_P H_3}\end{aligned} \qquad (48)$$

Flows are defined as:

$$F_i = K_{vi} H_i, \quad i \in I_{1:3} \qquad (49)$$

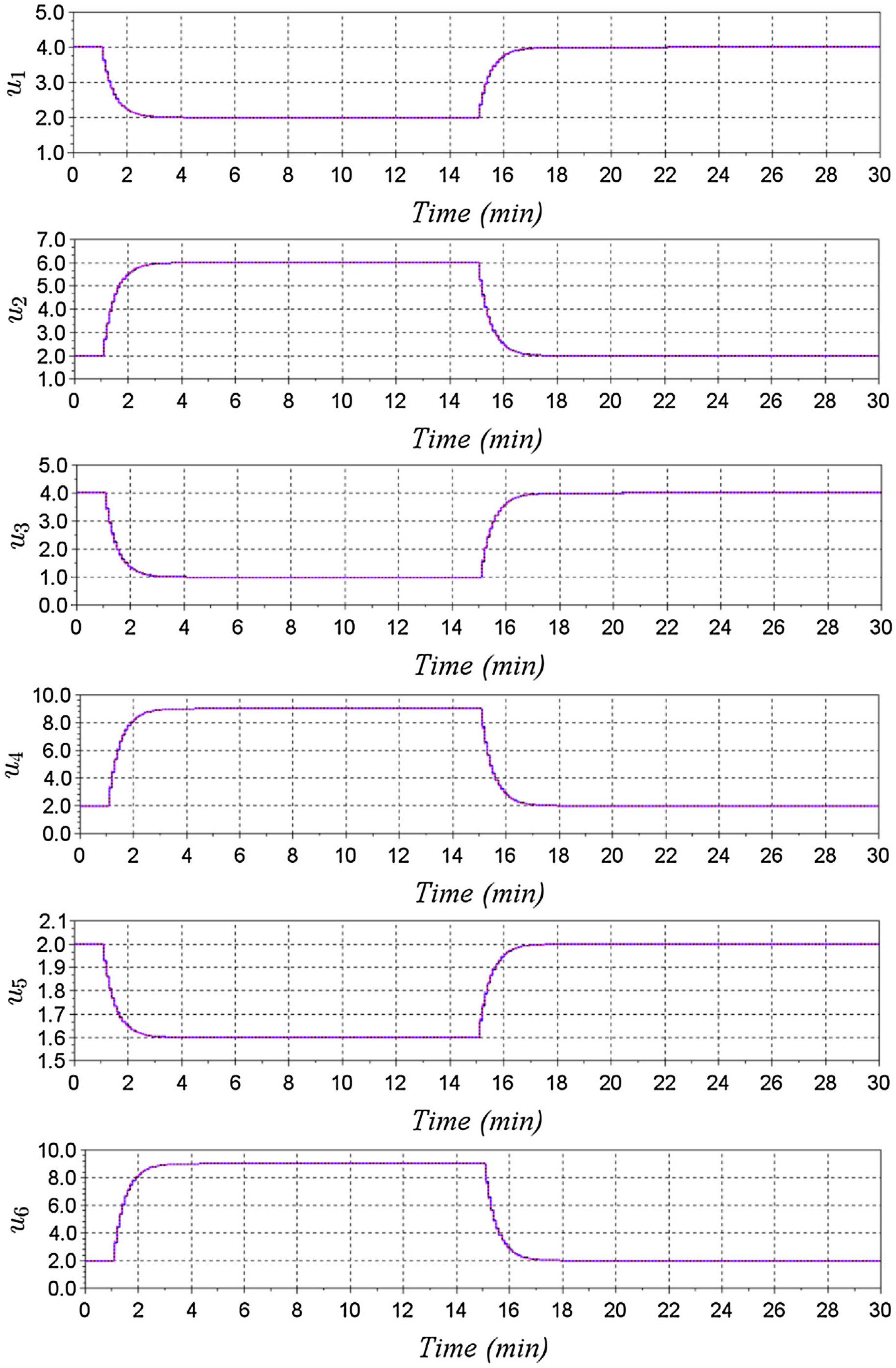


Fig. 7 – Dynamic evolution of manipulated inputs for OP1A obtained by: centralized control (black solid line), DMPC-1 (blue dashed dot line) and DMPC-2 (magenta dotted line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and the reaction constants are defined as:

$$K_{Ai} = K_A \exp\left(\frac{-E_A}{RT_i}\right), \quad i \in I_{1:3} \tag{50}$$

$$K_{Bi} = K_B \exp\left(\frac{-E_B}{RT_i}\right), \quad i \in I_{1:3} \tag{51}$$

The recycle flow and weight percents satisfy

$$F_D = 0.01F_R \tag{52}$$

$$x_{AR} = \frac{\alpha_A x_{A3}}{\bar{x}_3} \tag{53}$$

$$x_{BR} = \frac{\alpha_B x_{B3}}{\bar{x}_3} \tag{54}$$

$$\bar{x}_3 = \alpha_A x_{A3} + \alpha_B x_{B3} + \alpha_C x_{C3} \tag{55}$$

The controlled outputs and manipulated inputs are denoted, respectively, as

$$y = [H_1, \quad T_1, \quad H_2, \quad T_2, \quad H_3, \quad T_3]^T \tag{56}$$

$$u = \left[F_{f1}, \quad Q_1, \quad F_{f2}, \quad Q_2, \quad F_R, \quad Q_3\right]^T \tag{57}$$

The definitions for the variables used in the above model can be found in Table 1. Table 2 shows the parameters val-

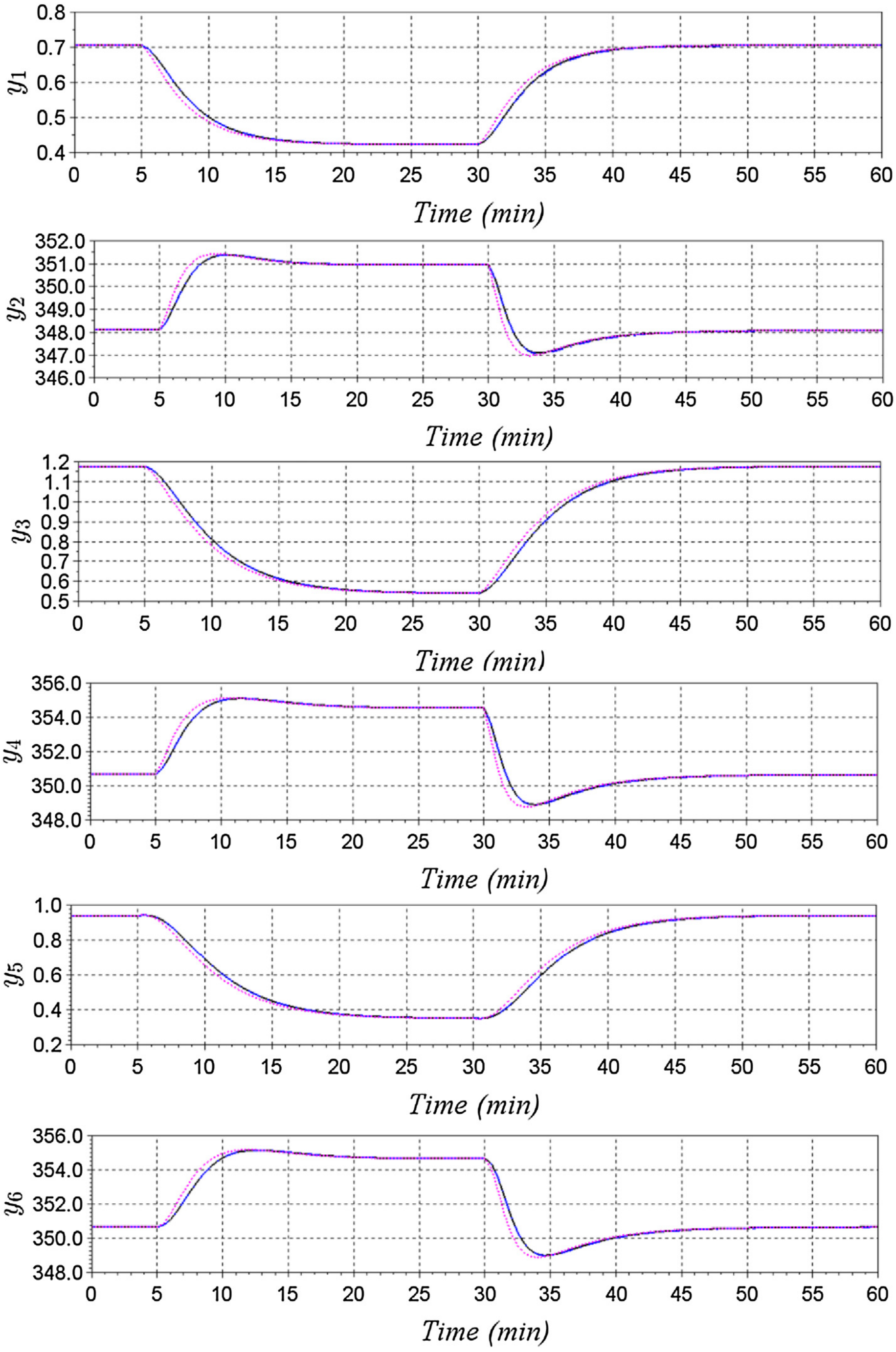


Fig. 8 – Dynamic evolution of controlled variable obtained for OP1B by: centralized control (black solid line), DMPC-1 (blue dashed dot line) and DMPC-2 (magenta dotted line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

ues and Table 3 presents stationary conditions of the system considered in this example.

The simulations presented herein use two design scenarios: *OP1A* and *OP1B*, both with a prediction horizon of $H_p = 5$ and a control horizon of $H_u = 5$. The weighting matrices for centralized control of *OP1A* are set as $\mathbf{Q} = \mathbf{I}(l, l)$, $\mathbf{R} = 100\mathbf{I}(m, m)$ and $\mathbf{W} = 10\mathbf{I}(m, m)$. The weighting matrices $\mathbf{Q}_i$, $\mathbf{R}_i$ and $\mathbf{W}_i$ (relating to subsystem *i*) for the DMPCs controllers of *OP1A* are: $\mathbf{Q}_i = \mathbf{I}(l_i, l_i)$, $\mathbf{R}_i = 100\mathbf{I}(m_i, m_i)$ and $\mathbf{W} = 10\mathbf{I}(m_i, m_i)$. In the scenario *OP1B* the weighting matrices for centralized control are set as $\mathbf{Q} = 10\mathbf{I}(l, l)$, $\mathbf{R} = \mathbf{I}(m, m)$ and $\mathbf{W} = \mathbf{I}(m, m)$. The weighting matrices $\mathbf{Q}_i$, $\mathbf{R}_i$ and $\mathbf{W}_i$ (relating to subsystem *i*), for DMPC controllers of *OP1B* are: $\mathbf{Q}_i = 10\mathbf{I}(l_i, l_i)$, $\mathbf{R}_i = \mathbf{I}(m_i, m_i)$ and $\mathbf{W} = \mathbf{I}(m_i, m_i)$. In the operation of this process the constraints in Table 4 need to be satisfied.

The control objective is to keep the controlled variables at the steady-state *ss*1 until the instant $k = 10T_s$, where T_s is the sampling time used (0.1 min, and 0.5 min, for *OP1A* and *OP1B*, respectively). Subsequently, the control objective is to drive the system to the steady-state *ss*2 and to keep in that condition until $k = 150T_s$ (*OP1A*) or $k = 60T_s$ (*OP1B*). Subsequently, the control objective is to drive the system to the steady-state *ss*1 and to keep in that condition until $k = 300T_s$ (*OP1A*) or $k = 120T_s$, when process operation ends.

Fig. 6 and Fig. 8 present the dynamics of the controlled outputs for centralized control and the distributed MPC strategy proposed in this paper with automatic partitioning at every sampling time. The control actions are given in Fig. 7 and Fig. 9.

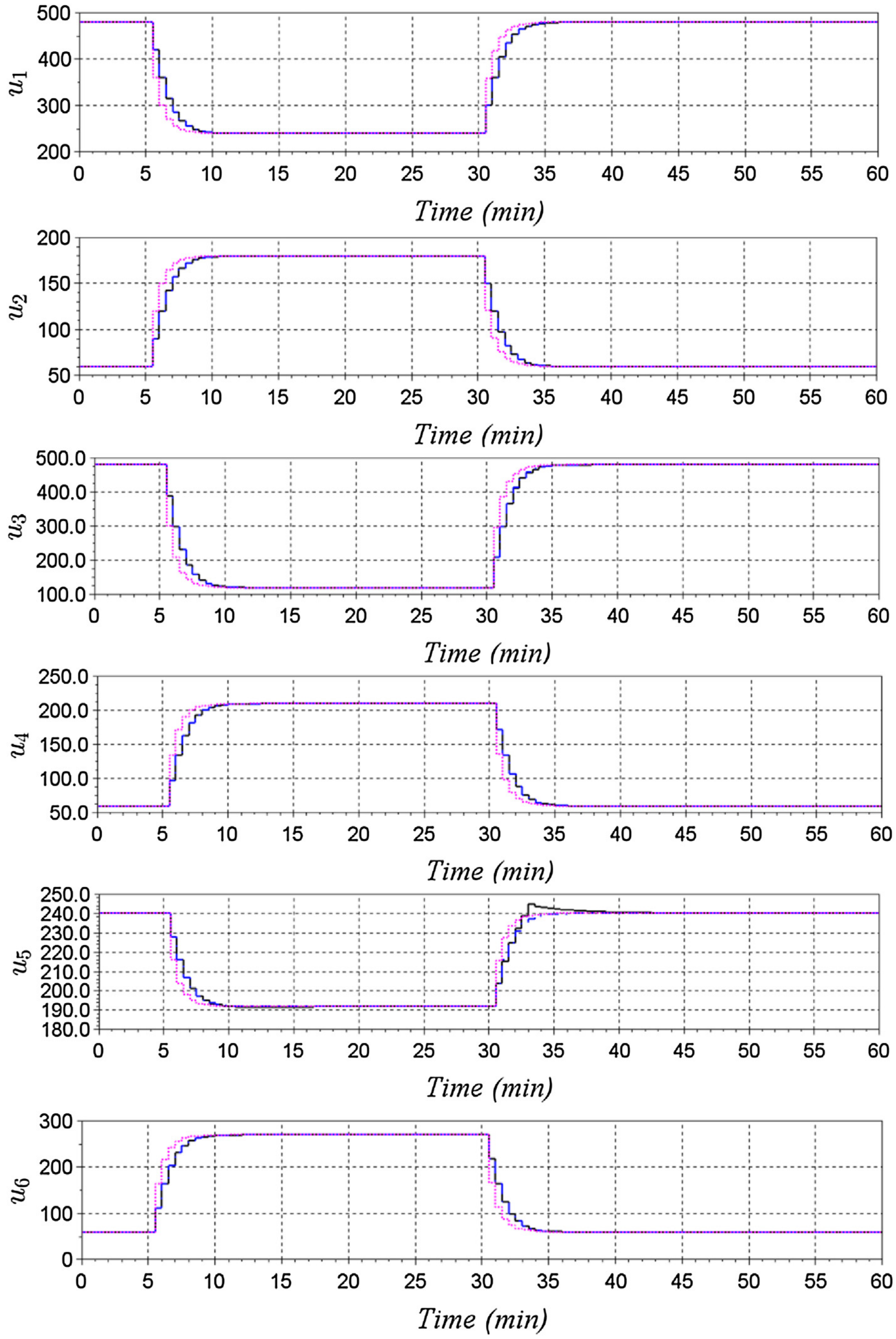


Fig. 9 – Dynamic evolution of manipulated inputs obtained for *OP1B* by: centralized control (black solid line), DMPC-1 (blue dashed dot line) and DMPC-2(magenta dotted line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

For the scenario *OP1A*, it can be observed from the Figs. 6 and 7 that the response obtained by the centralized control and the proposed DMPC strategies present coincident curves proving that the proposed methodologies present good results.

For the scenario *OP1B*, it should be noted that the responses obtained for the dynamics of the controlled outputs and manipulated inputs are equivalent, except for the responses obtained by the DMPC-2. The trajectory obtained by the DMPC-2 is the one closest to the pre-established reference trajectory for the variables. This shows that the presence of the iterative process in this scenario, allowed to obtain improved results. The DMPC-2 for scenario *OP1B* presented performance improvement of more than 64% for the manipulated inputs and more than 14% for the controlled outputs when compared to the centralized control with same control setting as before.

The processing time of the controllers' calculations were recorded in order to promote another type of comparison between the control structures evaluated. Here are some characteristics of the computer used in the simulations:

Processor: Intel ® Core$^{\mathrm{TM}}$ i7-45100U CPU @ 2.00GHz
RAM Memory: 2 × 8GB DDR3 @ 1600MHz
Operational System: Windows 10 Home 64 bits
Simulation Software: Scilab (5.5.2) 64 bits

Next, Table 5 shows the sum of the calculation times of the control actions for each control strategy throughout the

Table 5 – Total processing time for control calculations.

Type of control	Time (s) for *OP1A*	Time (s) for *OP1B*
Centralized	228.96	201.46
DMPC-1	74.88	41.61
DMPC-2	377.16	404.21

Table 6 – Maximum processing time for control calculations.

Type of control	Time (s) for *OP1A*	Time (s) for *OP1B*
Centralized	2.97	7.69
DMPC-1	0.83	1.27
DMPC-2	3.95	4.66

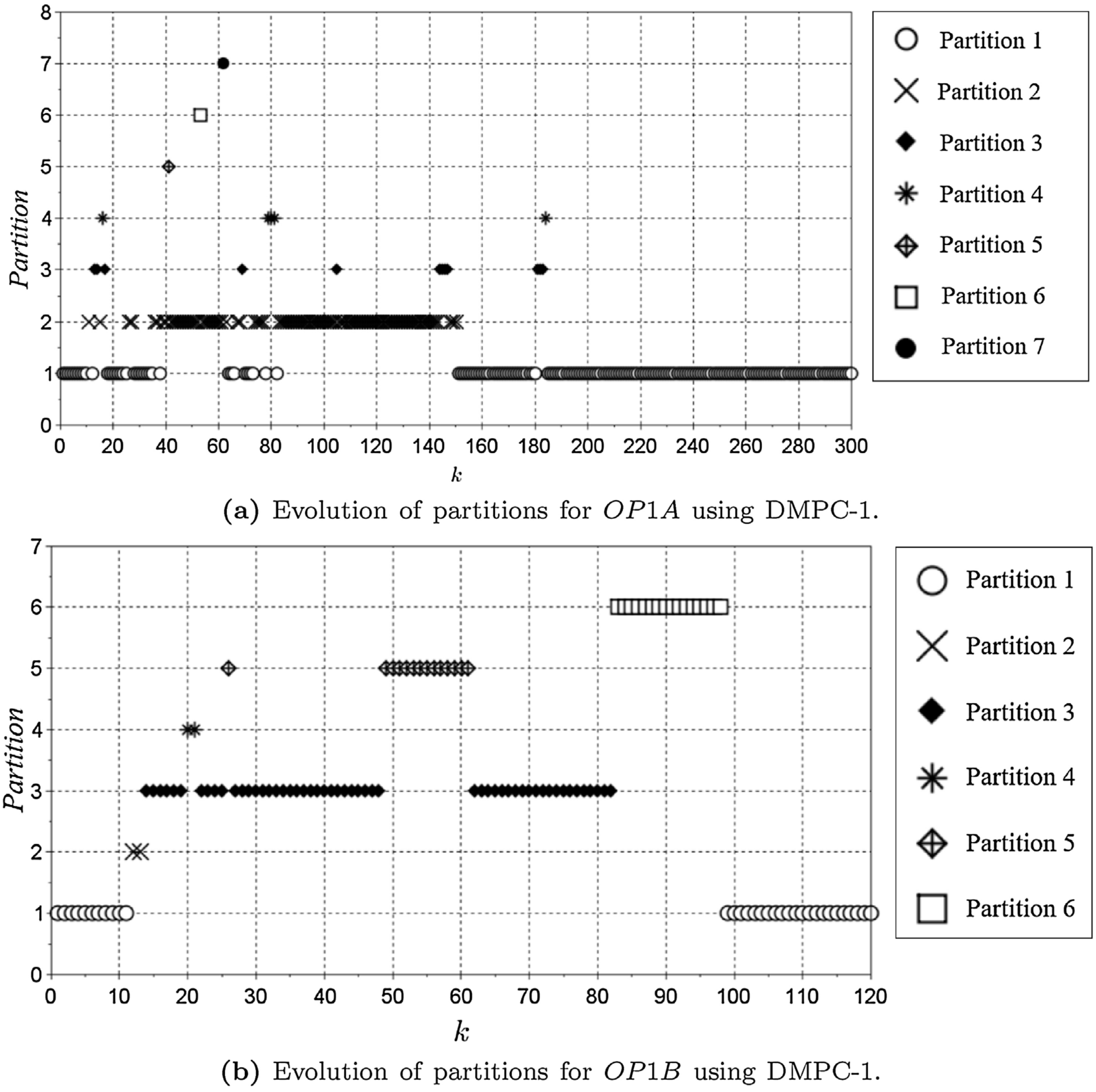
operation time considered and for the two scenarios considered. (*OP1A* and *OP1B*). Table 6 shows the maximum time to the processing time of the control action calculations at the time when this was done more slowly. This information is important to verify if the time used by controllers has exceeded sampling time.

Analyzing the results presented in Table 5, it is noted that DMPC-1 is the fastest control strategy and DMPC-2 is the one that presents the most time-consuming in its calculations. From Table 6, it is noted that the maximum time of the controllers does not exceed the sampling time used in each scenario (10 seconds for OP1A and 30 seconds for OP1B). Of course this performance would likely have larger benefit if the system dimension was larger than the toy example presented, just to show the methodology.

The developed controllers presented responses equivalent or close to those obtained by centralized control. Evaluating all the considered aspects, it can be affirmed for the scenarios considered, that if better performance is desired without worries with the processing time, the DMPC-2 control is a good choice, but if the processing time is an important question, the DMPC-1 control is a reasonable choice with no significant loss of performance. One could also decrease the processing time by selecting a less frequent model update and partitioning for that situation.

3.1.1. *Partition analysis*

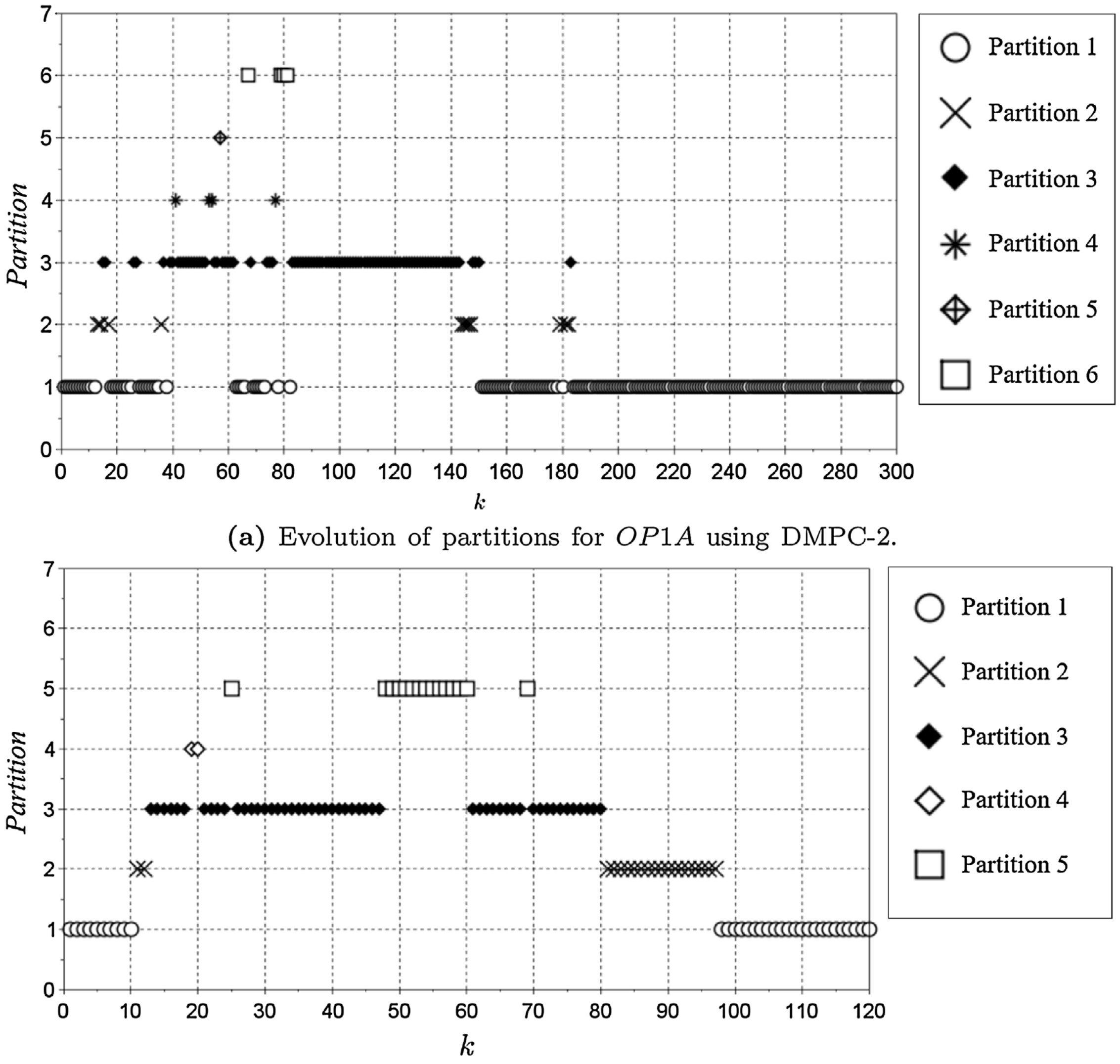
Figs. 10a–b and 11 a–b present the evolution of the proposed partitions for DMPC-1 and DMPC-2, respectively for the case studies *OP1A* and *OP1B*. In this section and all situations, the partitioning and model update were done at every sampling time, and the number of distinct partitions found by the pro-



(a) Evolution of partitions for *OP1A* using DMPC-1.

(b) Evolution of partitions for *OP1B* using DMPC-1.

Fig. 10 – Evolution of partitions for OP1A and OP1B using DMPC-1.



(a) Evolution of partitions for $OP1A$ using DMPC-2.

(b) Evolution of partitions for $OP1B$ using DMPC-2.

Fig. 11 – Evolution of partitions for OP1A and OP1B using DMPC-2.

posed methodology was found to be 7 (seven) and 6 (six), respectively.

For conciseness, the figures indicating each partition are not plotted for the case study I, but only their structural dependencies are presented. For the DMPC-1 control, note that in Table 7, the system presented 7 (seven) different types of submodels (partitions) in the *OP1A* scenario, with the first partition the most frequent, while in the *OP1B* scenario (see Table 8) 6 (six) different partitions were proposed during the studied transition, with the third partition the most frequent one.

For the DMPC-2 controller, note that in Table 9, the system presented six different types of partitions in the *OP1A* scenario, with the first partition the most frequent, while in the *OP1B* scenario (see Table 10) 5 (five) different partitions were proposed during the studied transition, with the third partition the most frequent one.

Naturally, this behavior is transition dependent and the submodel type does not imply that the submodel has constant values, but only the structural form of the partition is given by the partition algorithm. The local linearized matrices depend on the model update (linearization step) performed.

The effect of the partitioning frequency was evaluated in the two scenarios for the DMPC-2 strategy, applying partitioning at each sampling instant, every five, every ten, every fifteen and every twenty sampling times (T_s, $5T_s$, $10T_s$, $15T_s$ e $20T_s$, respectively). This evaluation may determine if it is really advantageous to carry out the proposed partitioning at each sampling instant. In addition it suggests that a procedure for selecting whether or not the partitioning would be advantageous could be implemented by evaluating the plant-model mismatch during the transition.

The results obtained by the partitioning frequency analysis indicate that performance losses are negligible. Therefore, for the case studies evaluated, the frequency of partitioning in the evaluated range does not impact the performance of the proposed controllers, but there is a gain in processing time. Tables Table 11 and Table 12 present the processing times of the DMPC-1 and DMPC-2 controllers, respectively, for the different partitions frequencies tested.

It is noted in Tables 11 and 12 that, as was expected, there was a decreasing in the processing time when the frequency of partitioning decreased. For the DMPC-1 strategy and OP1A scenario, for example, there was a 52 % decrease in partitioning every 20 sampling instants, instead doing so at every instant, without losing much performance.

It is known that the given performance is a function of the used tuning by the controllers, the tested scenario, and the

Table 7 – Simulation partitioning results for case study IA using DMPC-1.

Partition	Submodel	Description	Frequency
#1	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8 \text{ and } x_{12}\}$	61.0%
	2	$\{u_2, u_4 \text{ and } u_6\} \Rightarrow \{x_4, x_8 \text{ and } x_{12}\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8 \text{ and } x_{12}\}$	
	4	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
#2	1	$\{u_2, u_4 \text{ and } u_6\} \Rightarrow \{x_4, x_8 \text{ and } x_{12}\}$	32.3%
	2	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
	3	$\{u_1 \text{ and } u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
#3	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8 \text{ and } x_{12}\}$	4.0%
	2	$\{u_2, u_4 \text{ and } u_6\} \Rightarrow \{x_4, x_8 \text{ and } x_{12}\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
	4	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
#4	1	$\{u_2, u_4 \text{ and } u_6\} \Rightarrow \{x_4, x_8 \text{ and } x_{12}\}$	1.7%
	2	$\{u_3\} \Rightarrow \{x_4, x_5, x_8 \text{ and } x_{12}\}$	
	3	$\{u_1 \text{ and } u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
#5	1	$\{u_2, u_4 \text{ and } u_6\} \Rightarrow \{x_4, x_8, x_9 \text{ and } x_{12}\}$	0.3%
	2	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
	3	$\{u_1 \text{ and } u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
#6	1	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	0.3%
	2	$\{u_4\} \Rightarrow \{x_4, x_8 \text{ and } x_{12}\}$	
	3	$\{u_1 \text{ and } u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
	4	$\{u_2 \text{ and } u_6\} \Rightarrow \{x_4, x_8, x_9 \text{ and } x_{12}\}$	
#7	1	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	0.3%
	2	$\{u_1 \text{ and } u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9 \text{ and } x_{12}\}$	
	3	$\{u_6\} \Rightarrow \{x_4, x_8, x_9 \text{ and } x_{12}\}$	
	4	$\{u_2 \text{ and } u_4\} \Rightarrow \{x_4, x_8, x_9 \text{ and } x_{12}\}$	

disturbances used to investigate it. As it can be seen in this case, the results show that the distributed controller is not far behind in output performance compared to the centralized controller, and indeed the partitioning frequency can be chosen for each case without much loss of performance and if desired, only the most important one could be previously investigated (with optimizing tuning). In all cases herein, no effort was made to reach an optimal control tuning. However, the performance of partitioned control system may require that a tuning strategy need to be investigated for the DMPC depending on the model update and frequency selected in the proposed methodology.

3.2. *System II: Industrial Grinding System*

The system shown in Fig. 12 is a flowchart of a typical industrial grinding system and is presented in Ylinen et al. (1987). The controlled outputs are: the grain size of the hydrocyclone overflow stream, denoted as y_1 and power density of the hydrocyclone, denoted as y_2. These outputs are controlled by manipulating the feed ore (u_1) and secondary water supply (u_2). This system is linear, so the local linearization step and model update are not necessary. Additionally, the partitioning algorithm is used once only. The state-space matrix representation will be constant at all times. The motivation to include this linear system here is two folds, the first is to present details on how the algorithm works at each time instant k, and how to deal with processes in which the controlled variables are a combination of states.

The model of the plant expressed in incremental form for the inputs is given by:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\Delta\mathbf{u}(k) \tag{58}$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) \tag{59}$$

where,

$$\mathbf{A} = \begin{bmatrix} 0.716 & 0 & 0 & 0 & 0 & -0.283 & 0 & 0 & 0 \\ 0 & 0.863 & 1.049 & 0 & 0 & 0 & 0.755 & 0 & 0 \\ 0 & -0.109 & 0.186 & 0 & 0 & 0 & -0.529 & 0 & 0 \\ 0 & 0 & 0 & 0.670 & 0 & 0.176 & 0.865 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.135 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.819 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.819 \end{bmatrix} \tag{60}$$

Table 8 – Simulation partitioning results for case study IB using DMPC-1.

Partition	Submodel	Description	Frequency
#1	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	27.5%
	2	$\{u_2\} \Rightarrow \{x_4\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
#2	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	1.7%
	2	$\{u_2\} \Rightarrow \{x_4\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
#3	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	44.2%
	2	$\{u_2\} \Rightarrow \{x_4$ and $x_8\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
#4	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	1.7%
	2	$\{u_2\} \Rightarrow \{x_4$ and $x_8\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
#5	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	11.7%
	2	$\{u_2\} \Rightarrow \{x_4$ and $x_8\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_9\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
#6	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	13.3%
	2	$\{u_2\} \Rightarrow \{x_4\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	

Table 9 – Simulation partitioning results for case study IA using DMPC-2.

Partition	Submodel	Description	Frequency
#1	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	62.0%
	2	$\{u_2, u_4$ and $u_6\} \Rightarrow \{x_4, x_8$ and $x_{12}\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8$ and $x_{12}\}$	
	4	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
#2	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	3.7%
	2	$\{u_2, u_4$ and $u_6\} \Rightarrow \{x_4, x_8$ and $x_{12}\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
#3	1	$\{u_2, u_4$ and $u_6\} \Rightarrow \{x_4, x_8$ and $x_{12}\}$	31.3%
	2	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	3	$\{u_1$ and $u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
#4	1	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	1.3%
	2	$\{u_4\} \Rightarrow \{x_4, x_8$ and $x_{12}\}$	
	3	$\{u_1$ and $u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_2$ and $u_6\} \Rightarrow \{x_4, x_8, x_9$ and $x_{12}\}$	
#5	1	$\{u_2, u_4$ and $u_6\} \Rightarrow \{x_4, x_8$ and $x_{12}\}$	0.3%
	2	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	3	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	
#6	1	$\{u_2, u_4$ and $u_6\} \Rightarrow \{x_4, x_8$ and $x_{12}\}$	0.1%
	2	$\{u_3\} \Rightarrow \{x_4, x_5, x_8$ and $x_{12}\}$	
	3	$\{u_1$ and $u_5\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	

Table 10 – Simulation partitioning results for case study IB using DMPC-2.

Partition	Submodel	Description	Frequency
# 1	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	27.5%
	2	$\{u_2\} \Rightarrow \{x_4\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_9\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
# 2	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	15.8%
	2	$\{u_2\} \Rightarrow \{x_4\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
# 3	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	42.5%
	2	$\{u_2\} \Rightarrow \{x_4$ and $x_8\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
# 4	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	1.7%
	2	$\{u_2\} \Rightarrow \{x_4$ and $x_8\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	
# 5	1	$\{u_1\} \Rightarrow \{x_1, x_4, x_5, x_8$ and $x_{12}\}$	12.5%
	2	$\{u_2\} \Rightarrow \{x_4$ and $x_8\}$	
	3	$\{u_3\} \Rightarrow \{x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	4	$\{u_4\} \Rightarrow \{x_8$ and $x_{12}\}$	
	5	$\{u_5\} \Rightarrow \{x_1, x_4, x_5, x_8, x_9$ and $x_{12}\}$	
	6	$\{u_6\} \Rightarrow \{x_4$ and $x_{12}\}$	

Table 11 – Control calculation processing times for the partitioning frequency of DMPC-1.

Partitioning frequency	OP1A	OP1B
Every T_s	230.02	62.65
Every $5T_s$	156.77	60.72
Every $10T_s$	154.57	60.13
Every $15T_s$	139.09	59.88
Every $20T_s$	111.31	59.22

Table 12 – Control calculation processing times for the partitioning frequency of DMPC-2.

Partitioning frequency	OP1A	OP1B
Every T_s	590.14	229.11
Every $5T_s$	530.56	227.93
Every $10T_s$	511.97	226.77
Every $15T_s$	503.67	226.13
Every $20T_s$	485.23	226.05

$$\mathbf{B} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0.181 & 0 \\ 0 & 0.181 \end{bmatrix} \quad \text{and} \quad \mathbf{C} = \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{61}$$

In this case study, it was established that the control system must always obey the speed restriction on the manipulated variables, $|\Delta \mathbf{u}_{max}| = 1$ and the system is initially at $\mathbf{x}(0) = [0, 0, 0, 0, 0, 0, 0, 0, 0]^T$. The presented system was partitioned into two distinct subsystems using the proposed partitioning strategy. The resulting figures for the partitioned subsystems are given in Fig. 3, and the subsystems are represented by: Subsystem 1: $\{u_1\} \Rightarrow \{x_1, x_2, x_6$ and $x_8\}$, and Subsystem 2: $\{u_2\} \Rightarrow \{x_4, x_5, x_7$ and $x_9\}$. The respective subsystem models are given by:

Subsystem Model #1:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \\ x_6(k+1) \\ x_8(k+1) \end{bmatrix} = \begin{bmatrix} 0.716 & 0 & -0.283 & 0 \\ 0 & 0.863 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0.819 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_6(k) \\ x_8(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.181 \end{bmatrix} u_1(k) \tag{62}$$

$$\begin{bmatrix} y_1(k) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1(k) \\ x_2(k) \\ x_6(k) \\ x_8(k) \end{bmatrix} \tag{63}$$

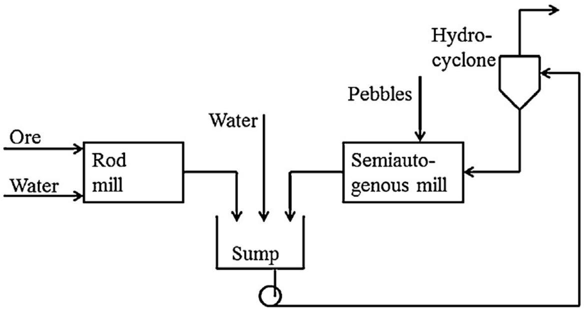


Fig. 12 – Flowsheet of industrial grinding system (Ylinen et al. (1987)).

Subsystem Model #2:

$$\begin{bmatrix} x_4(k+1) \\ x_5(k+1) \\ x_7(k+1) \\ x_9(k+1) \end{bmatrix} = \begin{bmatrix} 0.670 & 0 & 0 & 0 \\ 0 & 0.165 & 0.865 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0.819 \end{bmatrix} \begin{bmatrix} x_4(k) \\ x_5(k) \\ x_7(k) \\ x_9(k) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0.181 \end{bmatrix} u_2(k) \tag{64}$$

$$\begin{bmatrix} y_2(k) \end{bmatrix} = \begin{bmatrix} 1 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_4(k) \\ x_5(k) \\ x_7(k) \\ x_9(k) \end{bmatrix} \tag{65}$$

Figs. 2 and 3 show the structural graph for the original plant model, and the two partitioned submodels, respectively. The simulation results used a prediction horizon $H_p = 10$ and control horizon $H_u = 5$; the weighting matrices for centralized control are set as $\mathbf{Q} = diag(9.30,\ 10)$ and $\mathbf{R} = diag(1,\ 1)$. The weighting matrices $\mathbf{Q}_1$ and $\mathbf{R}_1$ (related to subsystem #1), $\mathbf{Q}_2$ and $\mathbf{R}_2$ (relating to subsystem #2) for the DMPC controllers are $\mathbf{Q}_1 = [10]$, $\mathbf{R}_1 = [1]$, $\mathbf{Q}_2 = [2.58]$ and $\mathbf{R}_2 = [6.02]$.

In the investigated scenario for System II, the control objective for the control variable y_1 is to keep it at its initial value until the instant $k = 10$, at which time it is desired to change the value to $y_1 = 1$ until the $k = 100$, and subsequently to return it to the initial condition until $k = 160$, when the process operation ends. For the controlled variable y_2, the control objective is to keep it at its initial value until $k = 60$, then to change it to $y_2 = 1$ until $k = 110$ and then to return it to the initial condition until $k = 160$, when the process operation ends. The subsystems order used in the calculation of the control actions in the distributed strategy was sequential from subsystem # 1 and then subsystem #2. It is important to highlight that a parallel calculation could also be implemented in case of LSS such as to decrease the overall processing time.

Fig. 13 presents the dynamics of the controlled outputs y_1 and y_2. The figure shows the responses obtained by centralized control and by DMPC strategy proposed in this paper. The control actions implemented by the manipulated inputs u_1 and u_2 that caused the responses in Fig. 13, are shown in Fig. 14.

Analyzing the results, one can notice that for the trajectory of the controlled output y_1 under the proposed DMPC is close to that obtained using centralized control, though the trajectory of y_2 under the DMPC scheme oscillates more than under the centralized scheme. The integral deviation of the values of the controlled outputs for DMPC in relation to the centralized control was high, but at the expense of a total input smaller than the one for a centralized approach. A better controller tuning could improve the response of this process.

The stability can be assessed by evaluating the **P** matrices for all subsystems. In this work, the set of inequality was evaluated by an LMI (Linear Matrix Inequality) solver. In this case, it was found two partitions and a stable closed-loop DMPC controller, since $\mathbf{P} < 0$.

$$\mathbf{P}_1 = \begin{bmatrix} \mathbf{P}_{1a} & \mathbf{P}_{1b} \end{bmatrix} \tag{66}$$

where,

$$\mathbf{P}_{1a} = \begin{bmatrix} 2235.53 & 0.00 & 708.94 & -630.73 & -525.24 \\ 0.00 & 4173.02 & 0.00 & 0.00 & 0.00 \\ 708.94 & 0.00 & 3051.99 & 245.21 & 264.67 \\ -630.73 & 0.00 & 245.21 & 2970.49 & 2613.17 \\ -525.24 & 0.00 & 264.67 & 2613.17 & 4984.49 \\ -1384.67 & 0.00 & 669.53 & 577.34 & 496.13 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 682.79 & 0.00 & -361.50 & -357.19 & -241.90 \\ 264.97 & 0.00 & -150.15 & -2467.64 & -2322.20 \\ 178.57 & 0.00 & -29.11 & -1549.13 & -4423.78 \end{bmatrix} \tag{67}$$

and

$$\mathbf{P}_{1b} = \begin{bmatrix} -1384.67 & 0.00 & 682.79 & 264.97 & 178.56 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 669.53 & 0.00 & -361.50 & -150.15 & -29.11 \\ 577.34 & 0.00 & -357.19 & -2467.64 & -1549.13 \\ -496.13 & 0.00 & -241.90 & -2322.20 & -4423.77 \\ 2206.18 & 0.00 & 678.60 & -168.52 & -244.79 \\ 0.00 & 4173.02 & 0.00 & 0.00 & 0.00 \\ 678.60 & 0.00 & 3092.32 & 29.51 & 153.12 \\ -168.52 & 0.00 & 29.51 & 2340.54 & 1498.27 \\ -244.79 & 0.00 & 153.12 & 1498.27 & 5534.57 \end{bmatrix} \tag{68}$$

$$\mathbf{P}_2 = \begin{bmatrix} \mathbf{P}_{2a} & \mathbf{P}_{2b} \end{bmatrix} \tag{69}$$

where,

$$\mathbf{P}_{2a} = \begin{bmatrix} -59.88 & 742.99 & 0.00 & 1645.29 & 61.01 & 401.69 \\ 297.49 & 238.77 & 0.00 & 61.01 & 4058.61 & 3056.69 \\ 286.19 & -108.59 & 0.00 & 401.69 & 3056.68 & 6020.60 \\ -925.17 & -726.99 & 0.00 & 148.67 & -469.27 & -297.88 \\ -639.82 & -1431.97 & 0.00 & 489.94 & -107.89 & -4.19 \\ 0.00 & 0.00 & 0.00 & 0.00 & 0.00 & 0.00 \\ 107.04 & 489.68 & 0.00 & -636.34 & -274.56 & -251.61 \\ -297.23 & -412.47 & 0.00 & -14.49 & -3028.50 & -2817.23 \\ 20.15 & -229.53 & 0.00 & 118.81 & -1636.99 & -4461.76 \end{bmatrix} \tag{70}$$

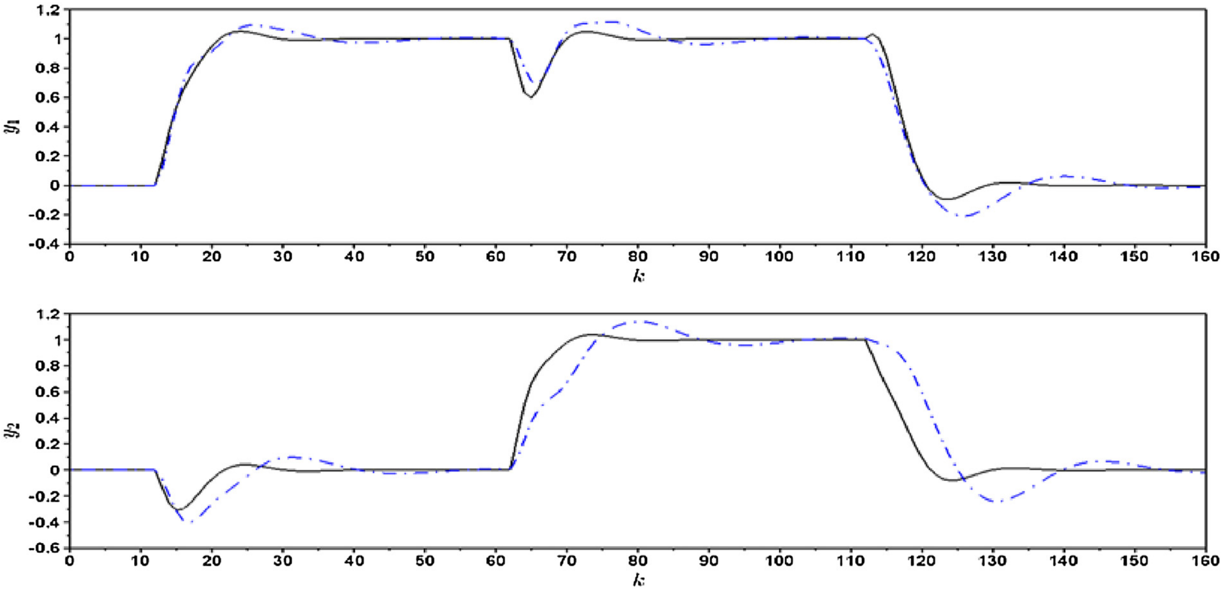


Fig. 13 – Dynamic evolution of controlled outputs obtained using centralized control (black solid line) and proposed DMPC (blue dashed line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

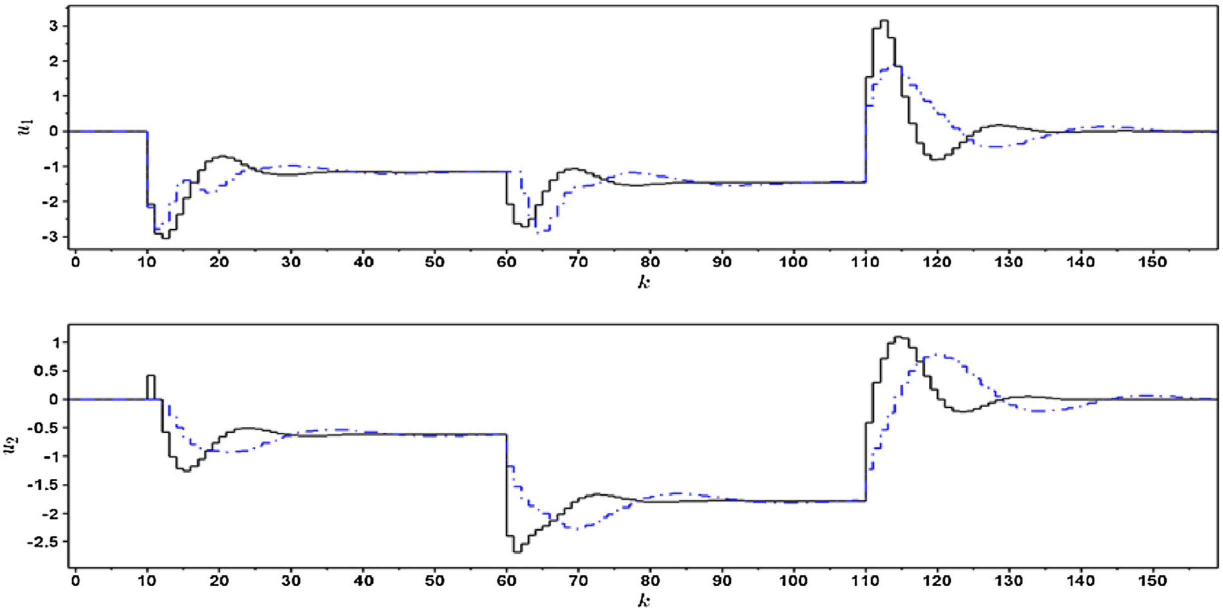


Fig. 14 – Dynamic evolution of manipulated inputs obtained using centralized control (black solid line) and proposed DMPC (blue dashed line). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

and

$$P_{2b} = \begin{bmatrix} 148.67 & 489.94 & 0.00 & -636.34 & -14.49 & 118.81 \\ -469.27 & -107.89 & 0.00 & -274.56 & -3028.50 & -1636.99 \\ -297.88 & -4.19 & 0.00 & -251.61 & -2817.23 & -4461.76 \\ 1311.24 & 1013.56 & 0.00 & 59.37 & 227.18 & -73.54 \\ 1013.56 & 3794.20 & 0.00 & 636.64 & 630.04 & -93.85 \\ 0.00 & 0.00 & 3467.49 & 0.00 & 0.00 & 0.00 \\ 59.37 & 636.64 & 0.00 & 1742.03 & -385.61 & 41.07 \\ 227.18 & 630.04 & 0.00 & -385.61 & 3315.43 & 1422.40 \\ -73.54 & -93.85 & 0.00 & 41.07 & 1422.40 & 5857.05 \end{bmatrix} \tag{71}$$

4. Conclusion

This paper introduces a simple and promising procedure that can be applied to partitioning a nonlinear model for distributed model predictive control. The methodology can be applied to nonlinear systems and extends the results of linear systems proposed by Zhang and Wang (2012) by handling processes with full input-state matrices. If guaranteed stability conditions are imposed to subsystems, the plantwide control can be implemented. In general, in all the case studies evaluated, the proposed controller developed in this study presented similar responses as those obtained by the centralized control, with the DMPC-2 strategy being the best performing strategy when compared to the centralized control version, but it is the technique with longer processing time. From the analysis of the frequency of application of the partitioning methodology by the controller, it can be stated that, for the case studies evaluated, it is possible to reduce the processing time by decreasing the frequency of partitioning, without losing much performance. It is also necessary to highlight that a procedure for achieving an offset-free MPC need to be imposed (plant-model mismatch) because the overall selected submodels may have different gains compared to the actual process used by the controller.

Acknowledgements

This research was supported by CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior, a Brazilian Agency for Support and Evaluation of Graduate Education) and CNPq (Brazilian National Council for Scientific and Technological Development).

References

Carlock, P.G., Fenton, R.E., 2001. System of systems (SoS) enterprise systems for information-intensive organizations. Syst. Eng. 4, 242–261.

Christofides, P.D., Scattolini, R., de la Peña, D.M., Liu, J., 2013. Distributed model predictive control: a tutorial review and future research directions. Comput. Chem. Eng. 51, 21–41, http://dx.doi.org/10.1016/j.compchemeng.2012.05.011.

Feiler, P., Northrop, L., Pollak, B., Institute, S.E., Pipitone, D., 2006. Ultra-large-scale Systems: The Software Challenge of the Future. Software Engineering Institute, Carnegie Mellon University https://books.google.com/books?id=MQKnDAEACAAJ.

Jamshidi, M., 1997. Large-Scale Systems: Modeling, Control and Fuzzy Logic. Prentice Hall.

Jamshidi, M., 2008. Systems of Systems Engineering Principles and Applications. CRC Press.

Jia, D., Krogh, B., 2002. Min-max feedback model predictive control for distributed control with communication. In: Proceedings of the 2002 American Control Conference (IEEE Cat. No. CH37301), American Automatic Control Council, pp. 4507–4512, http://dx.doi.org/10.1109/acc.2002.1025360.

Mercangöz, M., Doyle III, F.J., 2007. Distributed model predictive control of an experimental four-tank system. J. Process Control 17, 297–308, http://dx.doi.org/10.1016/j.jprocont.2006.11.003.

Mesarovic, M., Macko, D., Takahara, Y., 2000. Theory of hierarchical multilevel systems. In: Mathematics in Science and Engineering. Elsevier, Burlington, MA http://cds.cern.ch/record/1611599.

Motee, N., Sayyar-Rodsari, B., 2003. Optimal partitioning in distributed model predictive control. In: Proceedings of the American Control Conference, Denver, CO, pp. 5300–5305, http://dx.doi.org/10.1109/ACC.2003.1242570.

Rawlings, J.B., Stewart, B.T., 2008. Coordinating multiple optimization-based controllers: new opportunities and challenges. J. Process Control 18, 839–845, http://dx.doi.org/10.1016/j.jprocont.2008.06.005.

Rocha, R.R., Oliveira-Lopes, L.C., 2016a. A cooperative distributed model predictive control for nonlinear systems with automatic partitioning. In: Kravanja, Z., Bogataj, M. (Eds.), 26th European Symposium on Computer Aided Process Engineering. Elsevier, pp. 2205–2210, http://dx.doi.org/10.1016/B978-0-444-63428-3.50372-6, volume 38 of *Computer Aided Chemical Engineering*.

Rocha, R.R., Oliveira-Lopes, L.C., 2016b. A distributed model predictive controller for linear systems with automatic partitioning. Proceedings of the 6th IASTED International Conference on Intelligent Systems and Control, 296–303, http://dx.doi.org/10.2316/P.2016.841-045.

Sandell, N.R., Varaiya, P., Athans, M., Safonov, M.G., 1978. Survey of decentralized control methods for large scale systems. IEEE Trans. Autom. Control 23 (2), 108–128, http://dx.doi.org/10.1109/TAC.1978.1101704.

Stewart, B.T., Venkat, A.N., Rawlings, J.B., Wright, S.J., Pannocchia, G., 2010. Cooperative distributed model predictive control. Syst. Control Lett. 59, 460–469, http://dx.doi.org/10.1016/j.sysconle.2010.06.005.

Tang, W., Allman, A., Pourkargar, D.B., Daoutidis, P., 2018. Optimal decomposition for distributed optimization in nonlinear model predictive control through community detection. Comput. Chem. Eng. 111, 43–54, http://dx.doi.org/10.1016/j.compchemeng.2017.12.010.

Venkat, A.N., Rawlings, J.B., Wright, S.J., 2005. Stability and optimality of distributed model predictive control. In: Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference 2005, Seville, Spain, pp. 6680–6685, http://dx.doi.org/10.1109/CDC.2005.1583235.

Ylinen, R., Niemi, A.J., Iivarinen, T., 1987. A linear-quadratic-Gaussian control algorithm for sulphide ore grinding. Automatica 23, 287–294, http://dx.doi.org/10.1016/0005-1098(87)90002-1.

Zhang, L., Wang, J., 2012. Distributed model predictive control with a novel partition method. 31st Chinese Control Conference, 4108–4113.

Zhang, L., Wang, J., Li, C., 2013. Distributed model predictive control for polytopic uncertain systems subject to actuator saturation. J. Process Control 23, 1075–1089, http://dx.doi.org/10.1016/j.jprocont.2013.06.003.