# A communication-based distributed model predictive control approach for large-scale systems

P. Segovia[1,2,3*], L. Rajaoarisoa[3], F. Nejjari[1], E. Duviella[3], V. Puig[1,2]

*Abstract*— **This work presents a distributed model predictive control strategy as an alternative to conventional centralized approaches, which often suffer from implementation issues when applied to large-scale systems. The overall system is partitioned into minimally coupled subsystems based on its structural properties. Then, the coordination among the subproblems is achieved by means of a communication protocol, which allows each local controller to broadcast its solution to the rest of controllers with a coupled variable. The proposed approach is tested on the quadruple-tank process, and its effectiveness is proved by comparing the obtained results to those documented in an existing benchmark.**

## I. INTRODUCTION

The current complex technological processes lead to the concept of large-scale systems, which usually involve the transmission of matter and energy. Moreover, they are often spatially distributed and cover large areas. Typical examples include water networks, electrical grids and chemical and refining plants, among others. Thus, the control of these systems requires the use of advanced strategies.

Model predictive control (MPC) has found an overwhelming success due to its ease of understanding and application. The underlying principle of this approach consists in using a prediction of the system response, characterized by a dynamical representation of the process, to solve an online finite-horizon optimization problem at each sampling instant. Moreover, multiple performance objectives and physical and operational constraints can be dealt with in a natural manner. An exhaustive list of industrial applications using MPC is presented in [1], whereas other challenging applications of varied nature are addressed in [2].

Despite the adequacy of MPC to regulate the behavior of complex systems, centralized implementations are often impractical due to a number of reasons, e.g., spatial distribution, multi-time scales and non-scalability. Indeed, the large dimensionality of the centralized model often renders such representation impractical to reconfigure [3]. Moreover, the centralization of decisions in a single controller might affect the reliability of the network [4]. All these reasons foster the development of non-centralized strategies, which divide the system into a set of smaller subsystems, aiming at reducing the control design complexity and the computational demand. Two main types of strategies can be distinguished: decentralized approaches, which ignore the interactions between subsystems; and distributed approaches, which explicitly take into account the interactions in the control design, and thus usually offer a better performance with respect to the former approach. Extensive surveys on distributed MPC (DMPC) have been carried out in [5], [6]. Furthermore, a number of relevant works on the topic have been gathered in [7].

### *Summary of the paper and contribution*

This work presents a DMPC strategy applicable to large-scale systems. The design comprises two steps: first, the centralized problem is divided into smaller subproblems, thus distributing the computational burden among local controllers. System partitioning often leads to a set of subsystems which are not completely decoupled from one another, and thus a coordination mechanism among the local controllers is desirable. Several methods based on the decomposition of the Lagrangian relaxation (LR) can be found in the literature, but these are not well suited in the case of large-scale systems. Indeed, these approaches address the coordination of the subproblems by updating the Lagrange multipliers to achieve convergence, which is not trivial in the case of large-scale systems. Instead, the proposed coordination strategy consists in a communication protocol that allows the controllers to exchange information with one another regarding the coupled variables. Thus, the proposed approach does not require to compute and update the multipliers at each sampling instant.

The rest of the paper is organized as follows: Section II states the control problem to be solved and identifies the necessary steps to arrive at the final solution. Then, the system partitioning is proposed in Section III, and a communication-based DMPC is derived in Section IV. Section V tests the approach on the quadruple-tank process, which allows to draw conclusions in Section VI.

### *Notation*

Throughout this paper, let $\mathbb{Z}_{\geq 0}$, $\mathbb{R}^n$ and $\mathbb{R}^{n \times m}$ denote the set of natural non-negative scalars and the spaces of $n$-dimensional column vectors and $n$-by-$m$ matrices with real entries, respectively. Scalars are denoted by either lowercase or uppercase letters; vectors, by bold lowercase letters; matrices, by bold uppercase letters; and sets, by calligraphic symbols. Set inclusion is denoted with the symbol $\subseteq$; transposition, with the superscript $^\intercal$; and element-wise relations of vectors, with the operators $<$, $\leq$, $=$, $\geq$ and $>$.

[1] P. Segovia, F. Nejjari and V. Puig are with the Research Center for Supervision, Safety and Automatic Control (CS2AC), Universitat Politècnica de Catalunya (UPC), Rambla Sant Nebridi 22, 08222 Terrassa, Spain.

[2] P. Segovia and V. Puig are with Institut de Robòtica i Informàtica Industrial (CSIC-UPC), c/ Llorens i Artigas 4-6, 08028 Barcelona, Spain.

[3] P. Segovia, L. Rajaoarisoa and E. Duviella are with IMT Lille Douai, Univ. Lille, Lille, France.

* Corresponding author: P. Segovia: pablo.segovia@upc.edu

## II. PROBLEM STATEMENT

Consider the following general linear discrete-time state-space representation:

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k, \tag{1a}$$

$$\mathbf{y}_k = \mathbf{C}\mathbf{x}_k + \mathbf{D}\mathbf{u}_k, \tag{1b}$$

where $k \in \mathbb{Z}_{\geq 0}$ denotes the discrete-time instant. Moreover, the vectors $\mathbf{x}_k \in \mathcal{X} \subseteq \mathbb{R}^{n_x}$, $\mathbf{u}_k \in \mathcal{U} \subseteq \mathbb{R}^{n_u}$ and $\mathbf{y}_k \in \mathcal{Y} \subseteq \mathbb{R}^{n_y}$ represent the system states, control inputs and system outputs, respectively, while $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ are the system matrices. Furthermore, $\mathcal{X}$, $\mathcal{U}$ and $\mathcal{Y}$ define, using set membership, the feasible sets according to the physical and operational constraints [8].

Consider that a centralized MPC (CMPC) has been designed for (1) to achieve the desired control objectives. The following optimization problem must be solved:

$$\min_{\{\mathbf{u}_{i|k}\}_{i=k}^{k+H_p-1},\, \{\mathbf{x}_{i|k}\}_{i=k}^{k+H_p}} J\left(\mathbf{u}_{i|k}, \mathbf{x}_{i|k}\right) \tag{2a}$$

subject to:

$$\mathbf{x}_{i+1|k} = \mathbf{A}\mathbf{x}_{i|k} + \mathbf{B}\mathbf{u}_{i|k},\ i \in \{k, ..., k+H_p-1\}, \tag{2b}$$

$$\mathbf{u}_{i|k} \in \mathcal{U},\ i \in \{k, ..., k+H_p-1\}, \tag{2c}$$

$$\mathbf{x}_{j|k} \in \mathcal{X},\ j \in \{k, ..., k+H_p\}, \tag{2d}$$

$$\mathbf{x}_{k|k} = \mathbf{x}_k, \tag{2e}$$

with $\{\mathbf{u}_{i|k}\}_{i=k}^{k+H_p-1} \triangleq \{\mathbf{u}_{k|k}, \mathbf{u}_{k+1|k}, \cdots, \mathbf{u}_{k+H_p-1|k}\}$, and $\mathbf{x}_{i|k}$ is defined in the same manner. Constraint (2b) represents the state equation given by (1a), (2c) and (2d) define the feasible input and state sets, respectively, and (2e) sets the initial condition of the system.

The CMPC yields the optimal sequence of inputs (with respect to the chosen criteria) to be applied to the system, provided that (2) is feasible. However, only $\mathbf{u}_{k|k}$ is applied to the system and the rest of components are disregarded, according to the receding philosophy

$$\mathbf{u}_k^{MPC} \triangleq \mathbf{u}_{k|k}, \tag{3}$$

which is repeated at the next time instant.

Assuming that (2) is designed in such a way that stability and feasibility can be guaranteed, the problem at stake is that of deriving a non-centralized approach applicable to large-scale systems. To this end, consider that the overall system described by (1) can be divided into $M$ subsystems, where the model that represents the $l$-th subsystem is as follows:

$$\mathbf{x}_{k+1}^{(l)} = \mathbf{A}^{(l)}\mathbf{x}_k^{(l)} + \mathbf{B}^{(l)}\mathbf{u}_k^{(l)} + \sum_{\substack{m=1\\ m\neq l}}^{M} \left(\mathbf{A}^{(m)}\mathbf{x}_k^{(m)} + \mathbf{B}^{(m)}\mathbf{u}_k^{(m)}\right), \tag{4a}$$

$$\mathbf{y}_{k}^{(l)} = \mathbf{C}^{(l)}\mathbf{x}_k^{(l)} + \mathbf{D}^{(l)}\mathbf{u}_k^{(l)} + \sum_{\substack{m=1\\ m\neq l}}^{M} \left(\mathbf{C}^{(m)}\mathbf{x}_k^{(m)} + \mathbf{D}^{(m)}\mathbf{u}_k^{(m)}\right), \tag{4b}$$

where the summation terms in (4a) and (4b) represent the couplings between the $l$-th and $m$-th subsystems.

Then, the main idea consists in dividing the centralized control problem into several parts, where each of them is taken care of by a local controller. It is desirable that the local controllers exchange information with one another in order to account for the interactions among subsystems. In this regard, and taking into account the differences between the two main classes of non-centralized strategies described in Section I, a distributed control approach is preferred.

The overall system partitioning and the design of distributed controllers will be tackled in the next sections.

## III. SYSTEM PARTITIONING

The proposed partitioning strategy makes use of the adjacency matrix of the graph associated to (1). Once this matrix is obtained, it is then manipulated to identify the subsystems into which the overall system can be decomposed. The partitioning is performed offline, prior to solving the DMPC.

### *A. Building the adjacency matrix*

The adjacency matrix of a system defines its structural properties in a compact manner. The same information can be retrieved from the graph representation, where each variable is assigned to a vertex of the graph, and an edge connects a pair of variables whenever the corresponding coefficient in the system matrix is different from zero. This matrix can be built as follows [9]:

$$\mathbf{E} = \begin{bmatrix} \bar{\mathbf{A}} & \bar{\mathbf{B}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \bar{\mathbf{C}} & \mathbf{0} & \mathbf{0} \end{bmatrix}, \tag{5}$$

where the matrices $\bar{\mathbf{A}} = (\bar{a}_{ij}) \in \mathbb{R}^{n_x \times n_x}$, $\bar{\mathbf{B}} = \left(\bar{b}_{ij}\right) \in \mathbb{R}^{n_x \times n_u}$ and $\bar{\mathbf{C}} = (\bar{c}_{ij}) \in \mathbb{R}^{n_y \times n_x}$ are defined as

$$\bar{a}_{ij} = \begin{cases} 1,\ a_{ij} \neq 0, \\ 0,\ a_{ij} = 0, \end{cases} \quad \bar{b}_{ij} = \begin{cases} 1,\ b_{ij} \neq 0, \\ 0,\ b_{ij} = 0, \end{cases} \quad \bar{c}_{ij} = \begin{cases} 1,\ c_{ij} \neq 0, \\ 0,\ c_{ij} = 0, \end{cases} \tag{6}$$

and the dimensions of the several $\mathbf{0}$ blocks are as given in [9] so that $\mathbf{E}$ is a square matrix.

However, note that (5) needs to be adapted in order to deal with the general model (1). Indeed, the original formulation does not represent systems with a nonzero feedforward matrix $\mathbf{D}$. Thus, the definition of $\mathbf{E}$ is extended as follows:

$$\tilde{\mathbf{E}} = \begin{bmatrix} \tilde{\mathbf{A}} & \tilde{\mathbf{B}} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \tilde{\mathbf{C}} & \tilde{\mathbf{D}} & \mathbf{0} \end{bmatrix}, \tag{7}$$

where $\tilde{\mathbf{A}} = (\tilde{a}_{ij})$, $\tilde{\mathbf{B}} = \left(\tilde{b}_{ij}\right)$, $\tilde{\mathbf{C}} = (\tilde{c}_{ij})$ and $\tilde{\mathbf{D}} = \left(\tilde{d}_{ij}\right)$ are given by

$$\tilde{a}_{ij} = \begin{cases} 1,\ a_{ij} \neq 0, \\ 0,\ a_{ij} = 0, \end{cases} \quad \tilde{b}_{ij} = \begin{cases} 1,\ b_{ij} \neq 0, \\ 0,\ b_{ij} = 0, \end{cases} \quad \tilde{c}_{ij} = \begin{cases} 1,\ c_{ij} \neq 0, \\ 0,\ c_{ij} = 0, \end{cases} \quad \tilde{d}_{ij} = \begin{cases} 1,\ d_{ij} \neq 0, \\ 0,\ d_{ij} = 0. \end{cases} \tag{8}$$

**Algorithm 1** System partitioning based on the structural properties

```
Require: system matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$
Compute $\tilde{\mathbf{A}}$, $\tilde{\mathbf{B}}$, $\tilde{\mathbf{C}}$, $\tilde{\mathbf{D}}$ using (8)
Build $\tilde{\mathbf{E}}$ using $\tilde{\mathbf{A}}$, $\tilde{\mathbf{B}}$, $\tilde{\mathbf{C}}$, $\tilde{\mathbf{D}}$ as indicated in (7)
Apply the Cuthill-McKee ordering algorithm to $\tilde{\mathbf{E}}$
Identify blocks within $\tilde{\mathbf{E}}_{blk}$ with minimal couplings
Obtain the partitioned system (4) from each block
```

### *B. The reordering Cuthill-McKee algorithm*

Once the adjacency matrix has been obtained, the next step consists in determining possible *configurations* for the subsystems. The definition of configuration in this framework is that of a set of groups of variables and model equations, where each group defines a subproblem. The chosen approach consists in manipulating $\tilde{\mathbf{E}}$ such that a block-diagonal structure is attained. A number of methods can be employed, although not all of them are well suited for this task. For instance, some approaches compromise the coupling information due to the elimination of some matrix coefficients. In this regard, the Cuthill-McKee ordering algorithm [10] is chosen, which consists in performing row and column permutations to yield a reordered matrix with all its nonzero entries closer to the diagonal.

The steps to be followed in order to perform the partitioning of the system into a set of subsystems are detailed in Algorithm 1.

## IV. DMPC STRATEGY

Section III presented a methodology to decompose a system into a set of subsystems by identifying blocks within matrix $\tilde{\mathbf{E}}$. Since the identified subsystems are seldom completely decoupled from one another, the control strategy must ensure the coordination of the local controllers [11].

### *A. Motivation*

The coordination strategy proposed in this paper is based on a modification of the classical Lagrangian relaxation (LR) problem. This methodology approximates a constrained optimization problem by a simpler one by relaxing some conditions, thus providing approximate solutions to the original problem. The main idea consists in moving complicating constraints, i.e., constraints that hinder the solution of the optimization problem, to the objective function, penalizing their violation by means of Lagrange multipliers [12].

Several decomposition methods have been derived based on the LR problem, aiming at obtaining smaller subproblems that are easier to solve. For instance, the optimality condition decomposition (OCD) approach presented in [13] solves the decomposed LR problem by fixing some variables in the complicating constraints at each iteration, and then updating the multipliers. It is precisely the update of the multipliers which allows to coordinate the several subproblems. Nevertheless, the performance of the OCD is demonstrated only for small numerical problems, for which it is rather straightforward to obtain and update the values of the multipliers. This is not the case when dealing with large-scale systems. Thus, the proposed approach does not lean on the computation and update of the multipliers. Rather, it bounds the differences of the solutions obtained for the coupled variables in the different subproblems. The several local controllers exchange information with one another regarding the coupled variables, aiming at converging to the same value. This protocol leads to a distributed control strategy.

### *B. Control design*

The structure of the distributed controllers is based on the CMPC formulated in (2), which has been designed to guarantee stability and feasibility, as well as to perform according to the specifications. Since the local controllers are based on the centralized scheme, the communication-based DMPC approach is expected to provide a performance similar to the CMPC. Note that all the local control problems have the same structure. Thus, the controller assigned to the $l$-th subsystem solves the following problem:

$$\min_{\substack{\left\{\mathbf{u}^{(l)}_{i|k}\right\}_{i=k}^{k+H_p-1},\ \left\{\mathbf{x}^{(l)}_{i|k}\right\}_{i=k}^{k+H_p},\\ \left\{\boldsymbol{\gamma}^{(l)}_{i|k}\right\}_{i=k}^{k+H_p-1},\ \left\{\boldsymbol{\delta}^{(l)}_{i|k}\right\}_{i=k}^{k+H_p}}} J^{(l)} \tag{9a}$$

subject to:

$$\mathbf{x}^{(l)}_{i+1|k} = \mathbf{A}^{(l)}\mathbf{x}^{(l)}_{i|k} + \mathbf{B}^{(l)}\mathbf{u}^{(l)}_{i|k}, \tag{9b}$$
$$i \in \{k, ..., k+H_p-1\},$$
$$\mathbf{u}^{(l)}_{i|k} \in \mathcal{U}^{(l)},\ i \in \{k, ..., k+H_p-1\}, \tag{9c}$$
$$\mathbf{x}^{(l)}_{j|k} \in \mathcal{X}^{(l)},\ j \in \{k, ..., k+H_p\}, \tag{9d}$$
$$\mathbf{x}^{(l)}_{k|k} = \mathbf{x}_k, \tag{9e}$$
$$-\boldsymbol{\gamma}^{(l)}_{i|k} \leq \mathbf{u}^{(l)}_{i|k} - \mathbf{u}^{(m)}_{i} \leq \boldsymbol{\gamma}^{(l)}_{i|k}, \tag{9f}$$
$$i \in \{k, ..., k+H_p-1\},\ \forall m \mid \mathbf{u}^{(l)} \cap \mathbf{u}^{(m)} \neq \varnothing,$$
$$\boldsymbol{\gamma}^{(l)}_{i|k} \geq 0,\ i \in \{k, ..., k+H_p-1\}, \tag{9g}$$
$$-\boldsymbol{\delta}^{(l)}_{j|k} \leq \mathbf{x}^{(l)}_{j|k} - \mathbf{x}^{(m)}_{j} \leq \boldsymbol{\delta}^{(l)}_{j|k}, \tag{9h}$$
$$j \in \{k, ..., k+H_p\},\ \forall m \mid \mathbf{x}^{(l)} \cap \mathbf{x}^{(m)} \neq \varnothing,$$
$$\boldsymbol{\delta}^{(l)}_{j|k} \geq 0,\ j \in \{k, ..., k+H_p\}. \tag{9i}$$

Constraints (9b)–(9e) are equivalent to those in the centralized problem (2). However, the formulation of the DMPC is extended with respect to the CMPC by means of coordinating constraints for the coupled inputs and states, given by the pairs (9f)–(9g) and (9h)–(9i), respectively. The coordination is performed with all subsystems with a coupled input and state with respect to the $l$-th subsystem. The terms with a superscript $^{(m)}$ are certainly not decision variables to be optimized by the $l$-th but by the $m$-th controller, which are then transmitted to the $l$-th subsystem.

Aside from accounting for the communication between subsystems, constraints (9f)–(9i) guarantee that the differences between the solutions remain within the specified bounds. These bounds, denoted by $\boldsymbol{\gamma}^{(l)}_{i|k}$ and $\boldsymbol{\delta}^{(l)}_{i|k}$, are defined

**Algorithm 2** Communication-based DMPC implementation

```
Require: all parameters for the $M$ subproblems
 1: Initialize the subsystem states as $\mathbf{x}_0^{(l)}$, with $l = 1, ..., M$
 2: for $k = 0 : t_{sim} - 1$ do
 3:     Solve the $M$ control subproblems defined in (9)
 4:     while any difference between local solutions exceeds
        the threshold or any local solution does not converge
        do
 5:         Update the transmitted information as: $\mathbf{u}_i^{(l)} = \mathbf{u}_{i|k}^{(l)}$,
            $\mathbf{x}_i^{(l)} = \mathbf{x}_{i|k}^{(l)}$
 6:         Repeat step 3
 7:     end while
 8:     Set $\mathbf{u}_k^{MPC\ (l)} \triangleq \mathbf{u}_{k|k}^{(l)}$ and apply it to the $l$-th subsys-
        tem, for $l = 1, ..., M$
 9:     Measure/estimate the state $\mathbf{x}_{k+1}^{(l)}$, with $l = 1, ..., M$
10: end for
```

here as decision variables to avoid infeasibility problems in the first iterations at each time instant, but must be monitored to ensure that significant differences in the solutions are not allowed.

*Remark* 1. The general structure of (9) allows to consider couplings in the inputs and states. However, the particular system partitioning of each case study may override the need for certain communication links. □

Algorithm 2 illustrates the idea described in the previous lines. The *while* loop within the main simulation loop is only executed provided that the solution computed in the first iteration does not comply with the specifications, expressed in terms of convergence and accuracy of the local solutions. When all the local solutions converge from one iteration to the next one and the values of the coupled variables computed by all the agents involved are similar enough, the simulation can move on to the next time instant.

*Remark* 2. The only difference between consecutive iterations lies in the fact that the information broadcast to the rest of agents involved is updated to consider the most recent computed values. The current state used in the computation of the control law remains the same until the conditions are met, and the simulation can move on to the next instant. □

## V. CASE STUDY: THE QUADRUPLE-TANK PROCESS

In order to illustrate the system partitioning and the DMPC strategies presented in Sections III and IV, the quadruple-tank process depicted in Fig. 1 is considered. This multivariable zero location and direction laboratory process has been reported in detail in [14], proving its suitability for the purpose of control education. Furthermore, the four-tank process has been considered in [15] as the benchmark upon which several centralized and non-centralized control algorithms are tested, aiming at assessing and comparing their performances. A real prototype allows the authors to test their approaches and draw conclusions from real data collected by performing physical experiments.

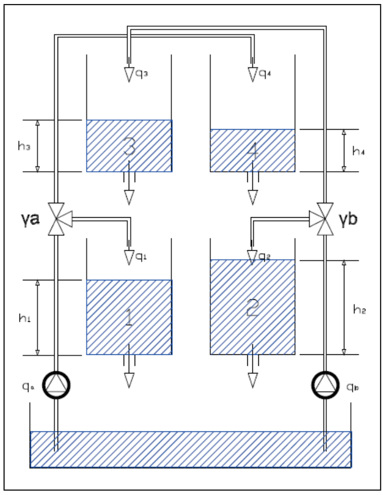

Fig. 1. Schematic diagram of the quadruple-tank process, taken from [15]

Motivated by the latter reference, the DMPC proposed in this work is also tested (in simulation) on the quadruple-tank process. Even though the proposed distributed approach was initially conceived for large-scale systems, it can still be tested on smaller systems that can be decomposed into a set of smaller subsystems, as is the case. First, the dynamical model of the process is provided, as well as the experimental design defined in [15], which allows to compare their results with those obtained in this work. The implementation is carried out in MATLAB®, using YALMIP [16] as parser.

### A. Description of the process

The goal of the control problem consists in steering the water levels of the two lower tanks to the desired values. The voltages applied to two pumps allow to convey water to the four tanks. Applying mass balances and Bernoulli's law to the system yields

$$
\begin{aligned}
\frac{dh_1}{dt} &= -\frac{a_1}{S}\sqrt{2gh_1} + \frac{a_3}{S}\sqrt{2gh_3} + \frac{\gamma_a}{S}q_a, \\
\frac{dh_2}{dt} &= -\frac{a_2}{S}\sqrt{2gh_2} + \frac{a_4}{S}\sqrt{2gh_4} + \frac{\gamma_b}{S}q_b, \\
\frac{dh_3}{dt} &= -\frac{a_3}{S}\sqrt{2gh_3} + \frac{1-\gamma_b}{S}q_b, \\
\frac{dh_4}{dt} &= -\frac{a_4}{S}\sqrt{2gh_4} + \frac{1-\gamma_a}{S}q_a,
\end{aligned}
\tag{10}
$$

where $h_i$ $[m]$ and $a_i$ $[m^2]$ are the water level and the discharge constant of the $i$-th tank, respectively, with $i \in \{1, 2, 3, 4\}$, $S$ $[m^2]$ is the cross section of the tanks, $q_j$ $[m^3h^{-1}]$ and $\gamma_j$ (dimensionless) denote the flow and the ratio of the three-way valve of the $j$-th pump, with $j \in \{a, b\}$, respectively, and $g$ $[ms^{-2}]$ is the gravitational acceleration. The values of the parameters are taken from Table 1 in [15]. Equation (10) constitutes the simulation model.

In order to test the DMPC, a linear prediction model is derived by linearizing (10) around an operating point, which is defined by the equilibrium levels given in the previously referenced table. This model is described by deviation variables around the operating point as: $x_i =$

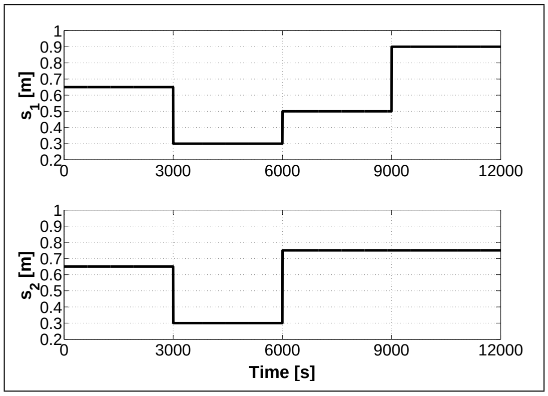

Fig. 2. Reference water levels for tanks 1 and 2

$h_i - h_i^0,\ i \in \{1,2,3,4\}$, $u_1 = q_a - q_a^0$ and $u_2 = q_b - q_b^0$. The linearized continuous-time state-space model reads as:

$$\begin{aligned} \frac{d\mathbf{x}}{dt} &= \mathbf{A}_c\mathbf{x} + \mathbf{B}_c\mathbf{u}, \\ \mathbf{y} &= \mathbf{C}_c\mathbf{x}, \end{aligned} \tag{11}$$

with $\mathbf{x} = [x_1\ x_2\ x_3\ x_4]^\intercal$, $\mathbf{u} = [u_1\ u_2]^\intercal$, $\mathbf{y} = [x_1\ x_2]^\intercal$ and

$$\begin{aligned} \mathbf{A}_c &= \begin{bmatrix} \frac{-1}{\tau_1} & 0 & \frac{1}{\tau_3} & 0 \\ 0 & \frac{-1}{\tau_2} & 0 & \frac{1}{\tau_4} \\ 0 & 0 & \frac{-1}{\tau_3} & 0 \\ 0 & 0 & 0 & \frac{-1}{\tau_4} \end{bmatrix}, \\ \mathbf{B}_c &= \begin{bmatrix} \frac{\gamma_a}{S} & 0 \\ 0 & \frac{\gamma_b}{S} \\ 0 & \frac{1-\gamma_b}{S} \\ \frac{1-\gamma_a}{S} & 0 \end{bmatrix}, \quad \mathbf{C}_c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \end{aligned} \tag{12}$$

where $\tau_i = S/\left(a_i\sqrt{2h_i^0/g}\right)$ $[s]$ is the time constant for the $i$-th tank. In order to implement the DMPC, (11) is discretized with a sampling time $T_s = 5\ s$.

### B. Experimental design

As stated before, the main objective consists in controlling the water levels of tanks 1 and 2 so that they stay as close as possible to the reference levels. A set of reference changes are featured throughout the experiment designed in [15], aiming at testing different equilibrium points that are far from one another. The reference signals are depicted in Fig. 2. Note that the initial values correspond to the operating points used in the linearization of (11).

In order to compare the proposed approach with those tested in [15], the same performance index is considered:

$$\begin{aligned} J = \sum_{k=0}^{N-1} \Big( &\big(h_1(k) - s_1(k)\big)^2 + \big(h_2(k) - s_2(k)\big)^2 + \\ &0.01\big(q_a(k) - q_a^s(k)\big)^2 + 0.01\big(q_b(k) - q_b^s(k)\big)^2\Big), \end{aligned} \tag{13}$$

where $q_a^s$ and $q_b^s$ are the values of the manipulable variables computed for the setpoints $s_1$ and $s_2$ in steady-state conditions. Besides, the performance index is not computed from the start of the experiment, but from the moment when the operating point is reached, i.e., at $t = 2700\ s$.

### C. Results

First, the partitioning approach presented in Section III is applied to (11). The adjacency matrix can be computed using (5)–(6) instead of the general case given by (7)–(8), as (11) does not feature a feedforward matrix. Its computation yields

$$\mathbf{E} = \begin{array}{c} \begin{matrix} x_1 & x_2 & x_3 & x_4 & u_1 & u_2 & & \end{matrix} \\ \left[\begin{array}{cccc|cc|cc} 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{array}\right] \end{array}. \tag{14}$$

Two possible partitionings can be identified from (14) by analyzing the incidence of each variable (columns) in the state and output equation (rows). Moreover, as the structure of the model is rather simple, it is not necessary to resort to the Cuthill-McKee algorithm in order to be able to identify possible decompositions for the quadruple-tank process. However, the partitioning of large-scale systems usually requires the reordering step, given the complexity and dimensionality of the centralized model.

- The *first partitioning* groups tanks 1 and 3 (rows 1, 3 and 7) in one subsystem, and tanks 2 and 4 (rows 2, 4 and 8) in the other one. This partitioning leads to a coupling in the inputs, and is the one proposed in [15].
- The *second partitioning* gathers tanks 1 and 4 (rows 1, 4 and 7) in one subsystem, and tanks 2 and 3 (rows 2, 3 and 8) in the other one. This partitioning leads to a coupling in the states.

Both partitionings result in two subsystems, each of them comprising two tanks. Since the degree of coupling is the same for both partitionings (two coupled inputs versus two coupled states), the second partitioning is chosen in this work due to its superior performance after trying both. However, in the future it will be investigated how to include the performance index in the partitioning algorithm.

*Remark* 3. The chosen partitioning overrides the need for constraints (9f) and (9g) in the DMPC provided in Section IV-B, being only necessary for the local controllers to share information regarding the coupled states. □

Therefore, the variables of each subsystem are as follows: $\mathbf{x}^{(1)} = \{x_1, x_4\}$, $\mathbf{u}^{(1)} = \{u_1\}$, $\mathbf{y}^{(1)} = \{x_1\}$, $\mathbf{x}^{(2)} = \{x_2, x_3\}$, $\mathbf{u}^{(2)} = \{u_2\}$ and $\mathbf{y}^{(2)} = \{x_2\}$. Based on this partitioning, matrices $\mathbf{A}^{(l)}$, $\mathbf{B}^{(l)}$ and $\mathbf{C}^{(l)}$ can be obtained from (12) by simple inspection, for $l = 1, 2$. Then, a DMPC is designed for each subsystem, with local objective functions based on (13). The results are depicted in Fig. 3, and show that the proposed controller is able to steer the water levels to the references with no significant error. Additionally, Fig. 3(c) depicts the controlled flows, whereas 3(d) shows the evolution of the water levels of the upper tanks, which are not forced to follow any reference signal.

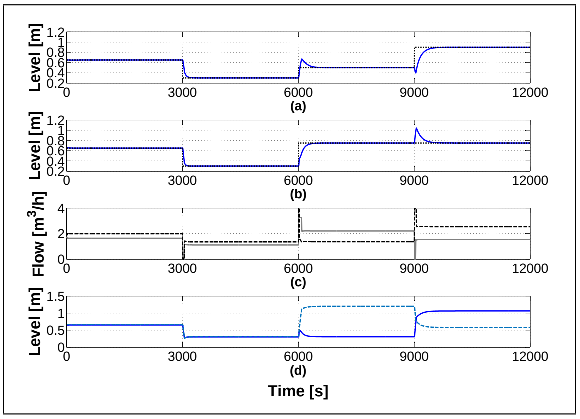

Fig. 3. DMPC results: (a) $y_1$ (solid blue) and $s_1$ (dotted black). (b) $y_2$ (solid blue) and $s_2$ (dotted black). (c) $q_a$ (solid gray) and $q_b$ (dashed black). (d) $h_3$ (solid dark blue) and $h_4$ (dashed light blue).

The performance index $J$, computed according to the indications given in Section V-B, equals 32.54. This performance is similar to those provided by the several controllers tested in [15], which allows to conclude that the performance of the proposed DMPC is comparable to the rest of controllers featured in the reference.

## VI. CONCLUSIONS

This work has presented a DMPC approach for large-scale systems as an alternative to centralized strategies, which often encounter implementation problems. The proposed approach is based on the partitioning of the system by considering its structural properties, which leads to a set of smaller subsystems with minimal couplings. Then, the obtained subproblems are coordinated by means of a communication protocol, allowing the local controllers to exchange information regarding the coupled variables. The proposed methodology is tested on the quadruple-tank process, for which an exhaustive benchmark exists. Although not strictly a large-scale system, this system possesses a number of interesting features that makes it a suitable case study. In the light of the results, it can be stated that the proposed approach performs satisfactorily, fulfilling the design objectives.

Despite the promising results, and following the statement in Section V-C, an issue to be addressed in future works consists in adding the cost function to the equations of the model to perform the partitioning of the system. Indeed, the current methodology is only based on the structural properties of the system, and thus only takes into account couplings in the dynamics. However, it is interesting to note that, depending on the proposed partitioning, the cost function might also be coupled. This consideration motivates the extension of the system partitioning approach.

Finally, and although the goal of this paper is to design a general methodology that can be applied to a large number of cases, the proposed approach has been derived with one particular system in mind: inland waterways. Indeed, centralized control and state estimation techniques have been successfully tested in [17] and [18], and even preliminary distributed control results have been obtained in [19]. However, applying the presented strategy to such systems requires to extend the approach with distributed state estimation, as the states are not directly measurable. Thus, it remains as an open problem that will be tackled in future works.

## ACKNOWLEDGMENTS

This work has been partially funded by the Spanish State Research Agency (AEI) and the European Regional Development Fund (ERFD) through the project SCAV (ref. MINECO DPI2017-88403-R). It has also been partially funded by AGAUR of Generalitat de Catalunya through the Advanced Control Systems (SAC) group grant (2017 SGR 482).

## REFERENCES

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

[2] L. Magni, D. M. Raimondo, and F. Allgöwer, Eds., *Nonlinear Model Predictive Control: Towards New Challenging Applications*. Springer Berlin Heidelberg, 2009.

[3] C. Ocampo-Martinez, D. Barcelli, V. Puig, and A. Bemporad, "Hierarchical and decentralised model predictive control of drinking water networks: Application to Barcelona case study," *IET Control Theory & Applications*, vol. 6, no. 1, pp. 62 –71, 5 2012.

[4] J. M. Grosso, C. Ocampo-Martínez, and V. Puig, "Non-centralized Predictive Control for Drinking-Water Supply Systems," in *Real-time Monitoring and Operational Control of Drinking-Water Systems*. Springer, 2017, pp. 341–360.

[5] R. Scattolini, "Architectures for distributed and hierarchical Model Predictive Control - a review," *Journal of Process Control*, vol. 19, no. 5, pp. 723 – 731, 2009.

[6] P. D. Christofides, R. Scattolini, D. Muñoz de la Peña, and J. Liu, "Distributed model predictive control: A tutorial review and future research directions," *Computers & Chemical Engineering*, vol. 51, pp. 21–41, 2013.

[7] J. M. Maestre and R. R. Negenborn, *Distributed model predictive control made easy*. Springer Sciene & Business Media, 2014, vol. 69.

[8] J. B. Rawlings and D. Q. Mayne, *Model predictive control: theory and design*. Nob Hill Pub. Madison, Wisconsin, 2009.

[9] D. D. Siljak, *Decentralized control of complex systems*. Courier Corporation, 2011.

[10] A. George and J. W. Liu, *Computer Solution of Large Sparse Positive Definite*. Prentice Hall Professional Technical Reference, 1981.

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

[12] H. Everett, "Generalized Lagrange multiplier method for solving problems of optimum allocation of resources," *Operations Research*, vol. 11, no. 3, pp. 399–417, 1963.

[13] A. J. Conejo, E. Castillo, R. Mínguez, and R. García-Bertrand, *Decomposition techniques in mathematical programming: engineering and science applications*. Springer Science & Business Media, 2006.

[14] K. H. Johansson, "The quadruple-tank process: a multivariable laboratory process with an adjustable zero," *IEEE Transactions on Control Systems Technology*, vol. 8, no. 3, pp. 456–465, 2000.

[15] I. Alvarado, D. Limón, D. Muñoz de la Peña, J. M. Maestre, M. A. Ridao, H. Scheu, W. Marquardt, R. R. Negenborn, B. De Schutter, F. Valencia, and J. Espinosa, "A comparative analysis of distributed MPC techniques applied to the HD-MPC four-tank benchmark," *Journal of Process Control*, vol. 21, no. 5, pp. 800 – 815, 2011.

[16] J. Löfberg, "YALMIP: a toolbox for modeling and optimization in MATLAB," in *IEEE International Symposium on Computer Aided Control Systems Design*, Taipei, Taiwan, 2004.

[17] P. Segovia, L. Rajaoarisoa, F. Nejjari, E. Duviella, and V. Puig, "Input-delay model predictive control of inland waterways considering the backwater effect," in *2018 IEEE Conference on Control Technology and Applications (CCTA)*. IEEE, 2018, pp. 589–594.

[18] ——, "Model predictive control and moving horizon estimation for water level regulation in inland waterways," *Journal of Process Control*, vol. 76, pp. 1–14, 2019.

[19] ——, "Distributed Input-Delay Model Predictive Control of Inland Waterways," in *HIC 2018. 13th International Conference on Hydroinformatics*, vol. 3. EasyChair, 2018, pp. 1893–1901.