

An Evolutionary-Games Approach for Distributed Predictive Control Involving Resource Allocation

Julian Barreiro-Gomez¹, Germán Obando², Carlos Ocampo-Martinez³, and Nicanor Quijano⁴

¹ Learning & Game Theory Laboratory, New York University in Abu Dhabi (NYUAD), Saadiyat Campus PO Box 129188, United Arab Emirates

² Department of Applied Mathematics and Computer Science, Universidad del Rosario, Bogotá, Colombia

³ Automatic Control Department, Universitat Politècnica de Catalunya, Institut de Robòtica i Informàtica Industrial (CSIC-UPC), Llorens i Artigas, 4-6, 08028 Barcelona, Spain

⁴ Departamento de Ingeniería Eléctrica y Electrónica, Universidad de los Andes, Carrera 1 No 18A-10, Bogotá Colombia

Abstract: This paper proposes a distributed model predictive control (DMPC) scheme based on population games for a system formed by a set of sub-systems. In addition to considering independent operational constraints for each sub-system, the controller addresses a coupled constraint that involves the sum of all control inputs. This constraint models an upper bound on the total amount of *energy* supplied to the plant. The proposed approach does not need a centralized coordinator when having a coupled constraint involving all the decision variables. The proposed methodology, which takes advantage of evolutionary game theory concepts, provides an optimal solution for the described problem. Moreover, it is shown that the methodology has *plug-and-play* features, i.e., for each already designed local MPC controller nothing changes when more sub-systems are added/removed to/from the global constrained control problem. Furthermore, the stability analysis of the proposed DMPC scheme is presented.

1 Introduction

An advantage of model predictive control (MPC) compared to other control techniques relies on its capability to deal with physical and operational constraints. Hence, this optimal control strategy has been successfully applied in different industrial applications, e.g., in networked large-scale systems [1], hydro-thermal power systems [2], wind farms [3], and drinking water system [4]. Nonetheless, if the goal is controlling large-scale systems, the big number of variables involved in the underlying optimization problem of MPC is an issue that limits the implementation of this controller, especially for satisfying sampling time constraints. To overcome the referred situation, a common approach is splitting the optimization problem into a set of smaller sub-problems that are addressed by local processors. This solution architecture is the so-called distributed model predictive control (DMPC). There are several problems that have been tackled by means of DMPC schemes. Among such problems, one issue that arises in many applications is the need to restrict the total energy supplied by the controller to the whole system. Basically, this constraint takes into account the limited amount of available resources (e.g., electric power, mechanical forces, cooling/heating power, inflows, etc.) that are used for controlling the plant. It is worth to mention that classic MPC formulations are able to take into consideration the aforementioned energy constraints. However, this classic scheme uses full information, i.e., there is a centralized authority that is in charge of finding optimal control inputs for the whole system. Thus, as was pointed out before, these centralized authorities suffer from computational burden issues. In contrast, the last few years have witnessed the emergence of novel techniques based on game theory to overcome the requirement of full information. A motivation to use a game-theoretical approach is the close relationship between this framework and distributed optimization. This connection has been identified and exploited in several works. For instance, [5] and [6] develop a methodology for designing payoff functions that ensure convergence to global optima of certain problems under distributed learning strategies. Similar ideas are studied in [7], where the authors point out the strong connection between variational inequality theory and game theory for solving convex optimization problems in a distributed way. In [8], the authors use replicator dynamics, a

learning mechanism of game theory, for addressing combinatorial optimization, while the same strategy is employed in [9–12] to solve distributed resource allocation problems in several contexts, i.e., economic dispatch of electric generators, urban drainage systems, and building temperature control. Other applications of game theory to engineering and economics problems that involves distributed optimization can be found in [13–16]. For further details on this topic, we refer the reader to [17].

DMPC has been extensively studied in recent years (see [18, 19] for a comprehensive review). Literature distinguishes several categories of DMPC techniques according to the type of model, coupling source, control architecture, and so forth. The technique proposed in this document aims to control a set of linear time-invariant systems that are coupled by an input constraint (which is related to an energy limitation). This problem has been tackled before using other distributed methods, e.g., Lagrangian relaxation of coupling constraints [20], alternating direction method of multipliers (ADMM) [21], local planning optimization and constraint tightening [22], and ADMM with early stopping criterion based on finite time consensus [23]. However, reported methods require either a centralized coordinator [21] or high communications load. These shortcomings are addressed by our approach. In fact, the method presented in this manuscript is fully distributed, i.e., it does not require the use a centralized authority. Among fully distributed methods, there are a variety of approaches. One of the most employed is based on the distributed alternating direction method of multipliers (D-ADMM). This approach was first proposed in [24] and it is possible to find several extensions in the literature (e.g., [22]). Even though D-ADMM-based techniques are flexible and efficient, they require more complex communications for solving the problem addressed in this research than the communications needed by our approach. The advantage of our method compared to D-ADMM comes from the fact that each node in D-ADMM handles a set of manipulated variables, while in our approach each node handles only a single variable. Therefore, the computation required by each node is also simplified. Other methods based on cooperative control have also been proposed in the literature to solve the problem tackled in this document [25, 26]. However, they either require that each agent has complete information of the constraints [26] or need a centralized

agent at some step of the algorithm [25]. Thus, they demand higher bandwidth communications.

Regarding the plug-and-play features in non-centralized MPC controllers, the main challenge consists on reducing the amount of required modifications over the design when plugging in or unplugging sub-systems of the whole systems. For instance, in [27], when sub-systems are added or removed, the distributed plug-and-play solution requires to redesign a number of controllers. Additionally, in [27] it is assumed that sub-systems get plugged in and unplugged in an off-line manner. In [28], before the modification is made by plugging or unplugging sub-systems, the feasibility should be verified. Moreover, when a new sub-system i is connected to a sub-system j , then the j^{th} controller should be redesigned. Likewise, the plug-and-play feature presented in [29] implies the modification of local control laws when it is desired to plug in or unplug sub-systems. Also, it is assumed that the modification is notified previously to the corresponding neighborhood by a request message. Differently, the advantage of the plug-and-play feature of the proposed method in this paper is that it does not require to modify any of the existing local controllers. In addition, the plug-and-play can be performed in an on-line manner.

The main contribution of this paper is the formalization of a distributed game-theory-based methodology for decoupling a traditional MPC scheme that controls medium/large scale plants formed by collections of sub-systems, which are subject to a coupling energy constraint. First, we propose to design local MPC controllers (one per sub-system) that are in charge of managing only local variables. Then, at a second stage, outputs from all local controllers are coordinated in a distributed fashion (without using a centralized authority) to satisfy the coupling constraint. Different from traditional dual decomposition and alternative direction method of multipliers [20], [30], the proposed distributed model predictive control (DMPC) scheme based on population dynamics does not require a central coordinator when managing a coupled constraint involving all the decision variables. Regarding the population dynamics stage in the proposed scheme, under some mild assumptions, it is shown that the solution computed by using the proposed method asymptotically converges to the optimal solution, while all constraints are satisfied. Besides, the stability of the closed-loop system with the DMPC is ensured by proving that there exists an equivalence between the proposed distributed scheme and a centralized MPC controller (CMPC).

The remainder of this paper is organized as follows. Section 3 presents the problem statement addressed throughout this manuscript. Section 4 introduces preliminary concepts about population games. Then, Section 5 presents the proposed distributed approach based on population games. Section 6 shows an illustrative example consisting in a system involving a limited resource with distributed information-sharing network. Next, Section 6 also presents the results and develops a discussion comparing the performance of the proposed DMPC with the performance obtained by using a centralized scheme. Finally, concluding remarks are drawn in Section 7.

2 Notation

Lower-case bold style denotes column vectors, e.g., \mathbf{p} . Non-bold style denotes scalar numbers, e.g., m . Calligraphic style denotes sets, e.g., \mathcal{S} . Real numbers, all the non-negative real numbers, and all the strictly positive real numbers are denoted by \mathbb{R} , $\mathbb{R}_{\geq 0}$, and $\mathbb{R}_{> 0}$, respectively. Non-negative integer numbers are denoted by $\mathbb{Z}_{\geq 0}$. Besides, $\mathbf{1}_n$ is a column vector of dimension n whose entries are all ones. The number of elements of a set is denoted by $|\cdot|$. The identity matrix of size $n \times n$ is denoted by \mathbb{I}_n , and $\mathbf{0}$ is a matrix or vector with null entries with suitable dimensions. Moreover, $\nabla f(\mathbf{p})$ denotes the gradient of the function $f(\mathbf{p})$, and $[\cdot]_+ = \max(0, \cdot)$. In the discrete time notation of the MPC controller, $\mathbf{x}_{k+j|k}$ refers to the prediction at time k of \mathbf{x} for $k+j$, where $k, j \in \mathbb{Z}_{\geq 0}$, i.e., in the argument $k+j|k$, the first element $k+j$ denotes prediction, whereas the second element k denotes current simulation time instant.

3 Control Problem Statement

Let $\tilde{\mathcal{S}} = \{1, \dots, \tilde{m}\}$ be the set of \tilde{m} controllable and stable sub-systems, which can share information according to a communication network. Let $\tilde{\mathcal{G}} = (\tilde{\mathcal{S}}, \tilde{\mathcal{E}}, \tilde{\mathbf{A}})$ be an undirected connected non-complete graph representing the topology of the information-sharing network, where $\tilde{\mathcal{S}}$ represents the nodes (i.e., each node is a sub-system), $\tilde{\mathcal{E}} \subseteq \{(i, j) : i, j \in \tilde{\mathcal{S}}\}$ corresponds to the edges, and $\tilde{\mathbf{A}}$ is the graph adjacency matrix. Each sub-system has a linear time-invariant discrete-time dynamics given by

$$\mathbf{x}_{i,k+1} = \mathbf{A}_i \mathbf{x}_{i,k} + \mathbf{B}_i \mathbf{u}_{i,k}^f, \quad \forall i \in \tilde{\mathcal{S}}, \quad (1)$$

where $\mathbf{x}_i \in \mathbb{R}^{n_{x,i}}$ denotes the state vector, $\mathbf{u}_i^f \in \mathbb{R}^{n_{u,i}}$ is the input vector of the i^{th} sub-system. Besides, matrices $\mathbf{A}_i \in \mathbb{R}^{n_{x,i} \times n_{x,i}}$ and $\mathbf{B}_i \in \mathbb{R}^{n_{x,i} \times n_{u,i}}$ are the system matrices with constant elements. Moreover, let $n_x \triangleq \sum_{i=1}^{\tilde{m}} n_{x,i}$ be the total number of states and $n_u \triangleq \sum_{i=1}^{\tilde{m}} n_{u,i}$ be the total number of control inputs in the system. Consider the optimization problem behind an MPC controller, denoted by \mathbf{P}_{MPC} , stated as

$$\min_{\mathbf{u}_k, \mathbf{u}_{k+1}, \dots, \mathbf{u}_{k+H_p-1}} J_k = \sum_{i=1}^{\tilde{m}} J_{i,k}, \quad (2a)$$

$$J_{i,k} = J_i^f(\mathbf{x}_{i,k+H_p|k}) + \sum_{j=0}^{H_p-1} J_i^\ell(\mathbf{x}_{i,k+j|k}, \mathbf{u}_{i,k+j|k}),$$

$$\text{s. t.} \quad \mathbf{x}_{i,k+1+j|k} = \mathbf{A}_i \mathbf{x}_{i,k+j|k} + \mathbf{B}_i \mathbf{u}_{i,k+j|k},$$

$$\forall i \in \tilde{\mathcal{S}}, j \in [0, H_p] \cap \mathbb{Z}_{\geq 0}, \quad (2b)$$

$$\mathbf{x}_{i,k+j|k} \in \mathcal{X}_i, \quad \forall i \in \tilde{\mathcal{S}}, j \in [0, H_p] \cap \mathbb{Z}_{\geq 0}, \quad (2c)$$

$$\mathbf{u}_{i,k+j|k} \in \mathcal{U}_i, \quad \forall i \in \tilde{\mathcal{S}}, j \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}, \quad (2d)$$

$$\sum_{i=1}^{\tilde{m}} \mathbf{1}_{n_{u,i}}^\top \mathbf{u}_{i,k+j|k} \leq \pi, \quad \forall j \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}, \quad (2e)$$

where $\mathcal{X}_i \triangleq \{\mathbf{x}_i \in \mathbb{R}^{n_{x,i}} : \underline{\mathbf{x}}_i \leq \mathbf{x}_i \leq \bar{\mathbf{x}}_i\}$ and $\mathcal{U}_i \triangleq \{\mathbf{u}_i \in \mathbb{R}^{n_{u,i}} : \underline{\mathbf{u}}_i \leq \mathbf{u}_i \leq \bar{\mathbf{u}}_i\}$. Moreover, vectors $\underline{\mathbf{x}}_i$ and $\bar{\mathbf{x}}_i$ are the lower and upper bound of the i^{th} sub-system state, respectively, while $\underline{\mathbf{u}}_i \geq 0$ and $\bar{\mathbf{u}}_i \geq \underline{\mathbf{u}}_i$ are the minimum and maximum admissible control inputs, respectively. The value $\pi \in \mathbb{R}_{\geq 0}$ in (2e) is the available resource (*energy* constraint) for the whole system. If the problem in (2) is feasible, there exists an optimal control sequence $\tilde{\mathbf{u}}_k^* \triangleq (\mathbf{u}_k^*, \mathbf{u}_{k+1}^*, \dots, \mathbf{u}_{k+H_p-1}^*)$ that minimizes the cost J_k in (2a) ($\tilde{\mathbf{u}}_{i,k}^*$ minimizes the cost $J_{i,k}$ for each sub-system $i \in \tilde{\mathcal{S}}$), which generates an optimal state trajectory $\tilde{\mathbf{x}}_k^* \triangleq (\mathbf{x}_{k+1}^*, \mathbf{x}_{k+2}^*, \dots, \mathbf{x}_{k+H_p}^*)$. The optimal control input that is applied to the system is $\mathbf{u}_{\text{MPC},k} = \mathbf{u}_k^*$, i.e., the control input for each sub-system, according to (1), is given by $\mathbf{u}_{i,k}^f = \mathbf{u}_{i,\text{MPC},k}$, for all $i \in \tilde{\mathcal{S}}$. Finally, the terms in the cost function J_k of the optimization problem (2) are of the following form [31][32]:

$$J_i^\ell(\mathbf{x}_{i,k}, \mathbf{u}_{i,k}) = \|\mathbf{x}_{i,k} - \mathbf{r}_{i,k}\|_{\mathbf{Q}_i}^2 + \|\mathbf{u}_{i,k}\|_{\mathbf{R}_i}^2, \quad (3)$$

and

$$J_i^f(\mathbf{x}_i) = \begin{cases} J_i^c(\mathbf{x}_i), & \mathbf{x}_i \in \mathcal{X}_i^f \\ \pi_i^c, & \mathbf{x}_i \notin \mathcal{X}_i^f \end{cases}, \quad (4)$$

where $\mathbf{Q}_i \in \mathbb{R}^{n_{x,i} \times n_{x,i}}$ is a positive semi-definite matrix whose entries are weights associated with the system's states, and $\mathbf{R}_i \in \mathbb{R}^{n_{u,i} \times n_{u,i}}$ is a positive definite weighting matrix that penalizes the control inputs. The set-point for the i^{th} sub-system is given by the vector $\mathbf{r}_{i,k}$. The function J_i^c is continuous and $J_i^c(\mathbf{x}_i) \geq 0$, for all $\mathbf{x}_i \in \mathcal{X}_i$, and the set $\mathcal{X}_i^f = \{\mathbf{x}_i \in \mathcal{X}_i : J_i^c(\mathbf{x}_i) \leq \pi_i^c\}$, where

$\mathbf{x}_i^s \in \mathcal{X}_i^f$ being \mathbf{x}_i^s the steady state of the i^{th} system. The aforementioned considerations are essential components for the MPC stability [31][33]. Consequently, J_i^f is also continuous. For instance, one selection might be $J_i^c(\mathbf{x}_{i,k}) = \|\mathbf{x}_{i,k} - \mathbf{r}_{i,k}\|_{\mathbf{V}_i}^2$, where $\mathbf{V}_i \in \mathbb{R}^{n_{x,i} \times n_{x,i}}$ is a positive semi-definite weighting matrix that penalizes the state error at the end of the prediction horizon. The aforementioned functions are selected in an appropriate way such that Assumption 1 holds [33].

Assumption 1. For each state $\mathbf{x}_{i,k} \in \mathcal{X}_i^f$, there exists a control input $\mathbf{u}_i^f = \mathbf{u}_i \in \mathcal{U}_i$ such that $\mathbf{x}_{i,k+1} \in \mathcal{X}_i^f$ for system in (1), i.e.,

$$\min_{\mathbf{u}_i \in \mathcal{U}_i} \left\{ J_i^f(\mathbf{x}_{i,k+1}) + J_i^{\ell}(\mathbf{x}_{i,k}, \mathbf{u}_{i,k}) : \mathbf{x}_{i,k+1} \in \mathcal{X}_i^f \right\} \leq J_i^f(\mathbf{x}_{i,k}),$$

which holds for all $\mathbf{x}_{i,k} \in \mathcal{X}_i^f$. \square

The problem addressed in this paper consists in the design of a DMPC by solving the optimization problem (2), which involves a full-coupled constraint (2e) and subject to the non-complete information-sharing graph $\tilde{\mathcal{G}}$. In order to solve this problem in a distributed way, population games are used and presented next.

4 Population Games

Assume a large population of agents (players) involved in a game, where each agent chooses a strategy from the set $\mathcal{S} = \{1, \dots, m\}$ of available strategies. Agents try to pursue an improvement on their benefits. There are n_u strategies associated to each one of the control inputs, plus a strategy representing the decision of not select any control input within the system (this latter strategy is going to be related to a slack variable in the optimization context), i.e., $m = n_u + 1$. Moreover, the set of strategies can be expressed as $\mathcal{S} = \{\mathcal{S}_1, \mathcal{S}_2, \dots, \mathcal{S}_{\tilde{m}}, m\}$, where $|\mathcal{S}_i| = n_{u,i}$, for all $i \in \tilde{\mathcal{S}}$. Let $p_i \in \mathbb{R}_{>0}$ be the amount of population playing strategy $i \in \mathcal{S}$. Hence, the vector

$$\mathbf{p} = [p_1^\top \dots p_m^\top]^\top = [p_1 \dots p_m]^\top,$$

with $\mathbf{p}_1 \in \mathbb{R}^{|\mathcal{S}_1|}, \dots, \mathbf{p}_m \in \mathbb{R}^{|\mathcal{S}_m|}$ and $p_m \in \mathbb{R}$ describes the population state, i.e., the distribution of the mass of players among the available strategies, i.e., \mathbf{p} describes the population state, where $p_i \in \mathbb{R}_{>0}^{n_{u,i}}$, for all $i \in \tilde{\mathcal{S}}$. The set of all feasible population states is the following simplex:

$$\Delta = \{\mathbf{p} \in \mathbb{R}_{\geq 0}^m : \mathbf{p}^\top \mathbf{1}_m = \pi\}, \quad (5)$$

where $\pi \in \mathbb{R}_{>0}$ is a constant value that represents the total mass of the population. The interior of the set of population states is denoted by $\text{int}\Delta = \{\mathbf{p} \in \mathbb{R}_{>0}^m : \mathbf{p}^\top \mathbf{1}_m = \pi\}$. The fitness function $f_i : \Delta \rightarrow \mathbb{R}$ maps a population state into a payoff (real value) that the population p_i receives for playing the i^{th} strategy. For notational convenience, $\mathbf{f} = [f_1 \dots f_m]^\top$ is the column vector of all fitness functions. This paper focuses on full-potential games due to their convergence properties and the relationship between this class of games with optimization problems (see Proposition 1).

Definition 1. A game \mathbf{f} is a full-potential game if there exists a continuously differentiable function $f(\mathbf{p})$, known as potential function, satisfying $\mathbf{f}(\mathbf{p}) = \nabla f(\mathbf{p})$, for all $\mathbf{p} \in \mathbb{R}_{\geq 0}^m$. \square

The reason to focus our work on this class of games is that in full-potential games, the Nash equilibrium of $\mathbf{f}(\mathbf{p})$, which is denoted by $\mathbf{p}^* \in \Delta$, is the solution of the following optimization problem:

$$\max_{\mathbf{p}} f(\mathbf{p}), \quad (6a)$$

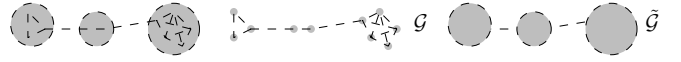


Fig. 1: Example of three sub-systems $\tilde{m} = 3$, where $\mathbf{u}_1 \in \mathbb{R}^3$, $\mathbf{u}_2 \in \mathbb{R}^2$, and $\mathbf{u}_3 \in \mathbb{R}^5$, then $n_u = 10$.

subject to

$$\sum_{i \in \mathcal{S}} p_i = \pi, \quad (6b)$$

$$p_i \geq 0, \quad \forall i \in \mathcal{S}, \quad (6c)$$

where $f(\mathbf{p})$ is strictly concave.

Proposition 1. (Adapted from [34]) If f is strictly concave on Δ , then the Nash equilibrium of the corresponding full-potential game is the unique maximizer of f on Δ .

Proof: This proof has been presented in [34]. \square

4.1 Distributed Smith Dynamics

As well as other population dynamics, Smith dynamics describe the changes along time of a population whose individuals are playing a strategic game [34]. An important feature of Smith dynamics is that population playing each strategy require complete knowledge of the whole population state (distribution of the population among available strategies) to decide how to evolve, i.e., individuals using Smith dynamics require full information. Recently, in [14], the authors propose an alternative formulation of Smith dynamics that only relies on local information*, the so-called distributed Smith dynamics (DSD). Under these dynamics, an agent playing the i^{th} strategy only needs information about agents playing certain strategies (the neighboring strategies) to evolve. This local framework is represented by means of an undirected and connected graph $\mathcal{G} = (\mathcal{S}, \mathcal{E}, \mathbf{A})$, where the set of nodes \mathcal{S} is associated with the available strategies of the game, the set of links $\mathcal{E} \subseteq \{(i, j) : i, j \in \mathcal{S}\}$ represents allowed interaction among strategies, where the sub-graphs, which represent the control inputs at each sub-system, correspond to a complete graph, i.e., $\{(i, j) : i, j \in \mathcal{S}_\ell\} \subset \mathcal{E}$, for all $\ell \in \tilde{\mathcal{S}}$ (see example in Figure 1). Finally, \mathbf{A} is the adjacency matrix whose element $a_{ij} = 1$ if $(i, j) \in \mathcal{E}$, and $a_{ij} = 0$ otherwise.

Notice that the graphs \mathcal{G} and $\tilde{\mathcal{G}}$ are equivalent only if $n_{u,i} = 1$, for all $i \in \tilde{\mathcal{S}}$, but they are different in the general case. As an illustrative example, consider a system composed of three heterogeneous sub-systems communicated by a path graph and with $n_{u,1} = 3$, $n_{u,2} = 2$, and $n_{u,3} = 5$. Therefore, the information-sharing graphs \mathcal{G} and $\tilde{\mathcal{G}}$ are presented in Figure 1.

The DSD are given by

$$\dot{p}_i = \sum_{j \in \mathcal{N}_i} p_j [f_i - f_j]_+ - p_i \sum_{j \in \mathcal{N}_i} [f_j - f_i]_+, \quad (7)$$

where $\mathcal{N}_i = \{j : (i, j) \in \mathcal{E}\}$ is the set of neighbors of the node $i \in \mathcal{S}$. The DSD can also be written as

$$\dot{p}_i = \sum_{j \in \mathcal{N}_i} \frac{1}{2} \left((1 - \phi_{ij}) p_i + (1 + \phi_{ij}) p_j \right) [f_i - f_j], \quad (8)$$

where $\phi_{ij} = \text{sgn}(f_i - f_j)$. The following propositions show that, the constraint associated to the simplex, including the positiveness of the amount of population, is satisfied under the DSD in (7), and that the DSD also solve the optimization problem in (6a), i.e., the Nash equilibrium given in Proposition 1 is asymptotically stable under the DSD.

*Information depending only on the neighbors.

Proposition 2. (Introduced in [14]) The simplex Δ in (5) is an invariant set along the trajectories of the DSD in (7), i.e., if $\mathbf{p}(0) \in \Delta$, then $\mathbf{p}(t) \in \Delta$, for all $t > 0$.

Proof: The DSD can be written by using the adjacency matrix of the graph \mathcal{G} as follows:

$$\begin{aligned} \dot{p}_i &= \sum_{j \in \mathcal{S}} a_{ij} p_j [f_i - f_j]_+ - \sum_{j \in \mathcal{S}} a_{ij} p_i [f_j - f_i]_+, \\ \sum_{i \in \mathcal{S}} \dot{p}_i &= \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} a_{ij} p_j [f_i - f_j]_+ - \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} a_{ij} p_i [f_j - f_i]_+, \end{aligned}$$

and, since \mathcal{G} is an undirected graph $a_{ij} = a_{ji}$, then

$$\sum_{i \in \mathcal{S}} \dot{p}_i = \sum_{j \in \mathcal{S}} \sum_{i \in \mathcal{S}} a_{ji} p_j [f_i - f_j]_+ - \sum_{i \in \mathcal{S}} \sum_{j \in \mathcal{S}} a_{ij} p_i [f_j - f_i]_+. \quad (9)$$

Notice that both terms in (9) have the same value (the only difference is the sub-index notation). Hence, $\sum_{i \in \mathcal{S}} \dot{p}_i = 0$. Moreover, notice that when $p_i = 0$ for any $i \in \mathcal{S}$, then $\dot{p}_i \geq 0$. Hence, if $p_i(0) \geq 0$, then $p_i(t) \geq 0$ for all $t \geq 0$ and for all $i \in \mathcal{S}$. \square

Proposition 3. (Adapted from [14]) Let the following assumptions hold:

- \mathbf{f} is a full-potential game with strictly concave potential function $f(\mathbf{p})$.
- $\dot{\mathbf{p}}$ is given by the DSD in (7).
- $\mathbf{p}^* = \arg \max_{\mathbf{p} \in \Delta} f(\mathbf{p})$.
- The graph \mathcal{G} is connected.
- The equilibrium point \mathbf{p}^* of the DSD belongs to $\text{int}\Delta$.

Then, \mathbf{p}^* is asymptotically stable.

Proof: Since $f(\mathbf{p})$ is strictly concave, $E_f(\mathbf{p}) = f(\mathbf{p}^*) - f(\mathbf{p})$ can be taken as a Lyapunov function candidate. The derivative of $E_f(\mathbf{p})$ along the trajectories of (7) is given by

$$\begin{aligned} \dot{E}_f(\mathbf{p}) &= -(\nabla f(\mathbf{p}))^\top \dot{\mathbf{p}}, \\ &= -(\mathbf{f}(\mathbf{p}))^\top \dot{\mathbf{p}}, \\ &= -(\mathbf{f}(\mathbf{p}))^\top \mathbf{L}^{(p)} \mathbf{f}(\mathbf{p}), \end{aligned}$$

where $\mathbf{L}^{(p)} = [l_{ij}^{(p)}]$ is a matrix whose entries $l_{ij}^{(p)}$ are defined as follows:

$$l_{ij}^{(p)} = \begin{cases} -\frac{a_{ij}}{2} \left((1 - \phi_{ij}) p_i + (1 + \phi_{ij}) p_j \right) & \text{if } i \neq j, \\ \sum_{k \in \mathcal{S}} \frac{a_{ik}}{2} \left((1 - \phi_{ik}) p_i + (1 + \phi_{ik}) p_k \right) & \text{if } i = j. \end{cases}$$

Notice that $\mathbf{L}^{(p)}$ is the Laplacian of the graph given by the tuple $\mathcal{G}^{(p)} = (\mathcal{V}, \mathcal{E}, \mathbf{A}^{(p)})$, where $\mathbf{A}^{(p)} = [a_{ij}^{(p)}]$ is the adjacency matrix whose entries are defined as

$$a_{ij}^{(p)} = \frac{a_{ij}}{2} \left((1 - \phi_{ij}) p_i + (1 + \phi_{ij}) p_j \right).$$

These entries are nonnegative since $\mathbf{p} \in \Delta$. Thus, $\mathbf{L}^{(p)} \geq 0$ and $\dot{E}_f(\mathbf{p}) \leq 0$. Therefore, the system is stable. Hence, a set \mathcal{B} around \mathbf{p}^* can be defined such that if $\mathbf{p}(0) \in \mathcal{B}$, then $\mathbf{p}(t) \in \text{int}\Delta$, for all $t \geq 0$. Thus, if $\mathbf{p}(0) \in \mathcal{B}$, the null space of $\mathbf{L}^{(p)}$ is equal to $\text{span}\{\mathbf{1}\}$ since $\mathcal{G}^{(p)}$ is connected. Notice that it is possible to conclude that $\mathcal{G}^{(p)}$ is connected since:

1. $\mathcal{G}^{(p)}$ and \mathcal{G} have the same topology in \mathcal{B} , i.e., if $\mathbf{p} \in \mathcal{B}$, $a_{ij}^{(p)} = 0$ only if $a_{ij} = 0$, and

2. \mathcal{G} is connected.

In this case, $\dot{E}_f(\mathbf{p}) = 0$ if and only if $f_i(\mathbf{p}) = f_j(\mathbf{p})$, for all $i, j \in \mathcal{S}$, i.e., $\dot{E}_f(\mathbf{p}) = 0$ only in \mathbf{p}^* . Therefore, applying the LaSalle's invariance principle, \mathbf{p}^* is asymptotically stable. \square

Corollary 1. The asymptotic stability of $\mathbf{p}^* \in \text{int}\Delta$ under the DSD stated in Proposition 3 holds for connected time-variant graphs $\mathcal{G}(t) = (\mathcal{V}, \mathcal{E}(t), \mathbf{A}(t))$. This statement can be seen from the Lyapunov function considered for the proof of Proposition 3 since it is a common function for all possible connected-graph topologies. \square

5 Proposed DMPC Design

The DMPC proposed in this paper has two main stages: at the first stage, local MPC controllers are designed for each one of the sub-systems that compose the entire system. These MPC controllers solve a local problem (considering only decision variables of the corresponding subsystem). Then, at the second stage, local solutions are coordinated in a distributed fashion by using DSD in order to provide an optimal solution for the whole system, i.e., an optimal solution for the problem in (2). Next, we elaborate on these two stages.

5.1 Designing of local MPC controllers

If constraint (2e) is omitted, then the problem in (2) can be decoupled since the dynamics of the sub-systems are decoupled as well as constraints (2b)-(2d). Hence, a local MPC controller for the i^{th} sub-system is designed by considering an optimization problem denoted by \mathbf{P}_{LMPC} as follows:

$$\min_{\hat{\mathbf{u}}_{k,i}, \hat{\mathbf{u}}_{k+1,i}, \dots, \hat{\mathbf{u}}_{k+H_p-1,i}} J_{i,k}, \quad (10a)$$

$$J_{i,k} = J_i^f(\mathbf{x}_{i,k+H_p|k}) + \sum_{j=0}^{H_p-1} J_i^{\ell}(\mathbf{x}_{i,k+j|k}, \hat{\mathbf{u}}_{i,k+j|k}),$$

$$\text{s. t. } \mathbf{x}_{i,k+1+j|k} = \mathbf{A}_i \mathbf{x}_{i,k+j|k} + \mathbf{B}_i \hat{\mathbf{u}}_{i,k+j|k},$$

$$\forall i \in \tilde{\mathcal{S}}, j \in [0, H_p] \cap \mathbb{Z}_{\geq 0}, \quad (10b)$$

$$\mathbf{x}_{i,k+j|k} \in \mathcal{X}_i, \forall i \in \tilde{\mathcal{S}}, j \in [0, H_p] \cap \mathbb{Z}_{\geq 0}, \quad (10c)$$

$$\hat{\mathbf{u}}_{i,k+j|k} \in \mathcal{U}_i, \forall i \in \tilde{\mathcal{S}}, j \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}. \quad (10d)$$

At time instant k , these local MPC controllers compute the optimal sequences $(\hat{\mathbf{u}}_{i,k}^*, \hat{\mathbf{u}}_{i,k+1}^*, \dots, \hat{\mathbf{u}}_{i,k+H_p-1}^*)$ for all sub-systems $i \in \tilde{\mathcal{S}}$, from which $\hat{\mathbf{u}}_{i,k}^*$ is obtained. Notice that it is not guaranteed that the control inputs $\hat{\mathbf{u}}_{i,k}^*$, for all $i \in \tilde{\mathcal{S}}$, satisfy the coupled constraint (2e) associated to a limited resource.

5.2 Coordination of the local control inputs via DSD

With the purpose of addressing the coupling constraint in (2e) in a distributed way, we design a full-potential game where players evolve according to DSD. The key idea is to take advantage of the optimality result in Proposition 1, so that the Nash equilibrium denoted by \mathbf{p}^* of the designed population game is the optimal solution of the entire control problem in (2). Thus, the control input applied to the i^{th} sub-system at time instant k is given by $\mathbf{u}_{i,k}^f = \mathbf{p}_i^*$, for all $i \in \tilde{\mathcal{S}}$.

To this end, since (2e) is not an equality constraint, then the additional strategy $m \in \mathcal{S}$ is a slack variable (i.e., the one that does not correspond to any control input), denoted by p_m , to the game. This slack variable is treated as a new node added to the graph and can be connected to any other arbitrary node. Additionally, its fitness function is chosen as $f_m = 0$. The slack variable allows the controller to use less than the total available resource when it is convenient. On the other hand, we consider bounds that guarantee feasibility of the computed control sequence. The bound $\mathbf{u}_i^{\min} \in \mathbb{R}^{n_{u,i}}$ is

locally computed by solving the optimization problem in (10a) with weighting matrices $\mathbf{Q}_i = \mathbf{V}_i = \mathbf{0}_{n_{x,i} \times n_{x,i}}$ and $\mathbf{R}_i = \mathbb{I}_{n_{u,i} \times n_{u,i}}$ according to the structure presented in (3) and (4). The bound $\mathbf{u}_i^{\max} \in \mathbb{R}^{n_{u,i}}$ is similarly found with $\mathbf{Q}_i = \mathbf{V}_i = \mathbb{I}_{n_{x,i} \times n_{x,i}}$ and $\mathbf{R}_i = \mathbf{0}_{n_{u,i} \times n_{u,i}}$. Both problems are solved subject to (2b), (2c), and (2d).

Remark 1. In general, the bounds $\underline{\mathbf{u}}_i$ and $\bar{\mathbf{u}}_i$ in (2d) are different from \mathbf{u}_i^{\min} and \mathbf{u}_i^{\max} , respectively. The values $\underline{\mathbf{u}}_i$ and $\bar{\mathbf{u}}_i$ determine the physical constraints for the control inputs, whereas \mathbf{u}_i^{\min} and \mathbf{u}_i^{\max} determine the bounds of control inputs that guarantee feasibility of the problem (10a). \square

To design the desired full-potential game, we employ the control inputs provided by local MPC controllers. Specifically, once the optimal control input has been computed from each local MPC controller by means of (10), we propose a strictly-concave full-potential function. This function corresponds to the objective function of problem (6a), which is denoted by \mathbf{P}_{DSD} . The full-potential function is written as follows [14]:

$$f(\mathbf{p}) = - \sum_{i=1}^{\tilde{m}} (\hat{\mathbf{u}}_i^* - \mathbf{p}_i)^\top \boldsymbol{\omega}_i (\hat{\mathbf{u}}_i^* - \mathbf{p}_i), \quad (11)$$

where $\boldsymbol{\omega}_i \in \mathbb{R}^{n_{u,i} \times n_{u,i}}$ assigns a weighting factor to each control input. The function f in (11) should be concave so that the corresponding full-potential game is stable, and it should be satisfied that, when there is enough resource, the maximum of the function $f(\mathbf{p})$ is obtained when $\mathbf{p}_i = \hat{\mathbf{u}}_i^*$. Notice that this function represents an error between an ideal case with infinite resources and the real case with limited resources, then the objective is to minimize the difference. Under the proposed full-potential function in (11), fitness functions of the population game are given by $\mathbf{f}(\mathbf{p}) = \nabla f(\mathbf{p})$, i.e., $\mathbf{f}_i(\mathbf{p}_i) = -2\boldsymbol{\omega}_i(\hat{\mathbf{u}}_i^* - \mathbf{p}_i)$. It is worth to mention that the described methodology does not need complete information about all control inputs and/or all sub-systems' states because:

1. the graph \mathcal{G} , which models the information constraints is non-complete, and
2. f_i only depends on information of the i^{th} sub-system, i.e, fitness functions are not coupled.

In order to satisfy the constraints of the optimization problem (2), the DSD are modified as follows:

$$\hat{p}_i = \sum_{j \in \mathcal{N}_i} \hat{p}_j [f_j - f_i]_+ - \hat{p}_i \sum_{j \in \mathcal{N}_i} [f_j - f_i]_+, \quad \forall i \in \mathcal{S}, \quad (12)$$

where the vector $\hat{\mathbf{p}} = [\hat{p}_1^\top \dots \hat{p}_m^\top]^\top$, or equivalently $\hat{\mathbf{p}} = [\hat{p}_1 \dots \hat{p}_m]^\top$. Moreover, $\hat{\mathbf{p}}_i = (\mathbf{p}_i - \mathbf{u}_i^{\min})$, for all $i \in \mathcal{S}$, and $\hat{p}_m = p_m$.

Summarizing, each one of the sub-systems is equipped with a local MPC controller that solves the optimization problem in (10a) every time instant $k \in \mathbb{Z}_{\geq 0}$. The control inputs computed by the local MPC controllers are then used for defining the fitness functions of the population game as in (11). Finally, population involved in this game evolve to the Nash equilibrium using the DSD. This Nash equilibrium solves the problem in (2). Therefore, it corresponds to the optimal control input applied to each sub-system.

In order to compute the fitness functions of the game, the DSD need \mathbf{u}_i^* , for all $i \in \mathcal{S}$. Furthermore, the DSD also require a feasible initial condition and the limits $(\mathbf{u}_i^{\min}, \mathbf{u}_i^{\max})$, for all $i \in \mathcal{S}$, to guarantee that the vector of final control inputs \mathbf{p}^* is feasible, i.e., $\mathbf{u}_i^{\min} \leq \mathbf{p}_i^* \leq \mathbf{u}_i^{\max}$, for all $i \in \mathcal{S}$. Hence, to establish this initial condition, a distributed algorithm for solving the associated constraint satisfaction problem (CSP) is needed. For instance, an alternative is to employ the CSP algorithm presented in [35].

Figure 2 presents the general scheme corresponding to the information requirements for the i^{th} sub-system. First, the local MPC

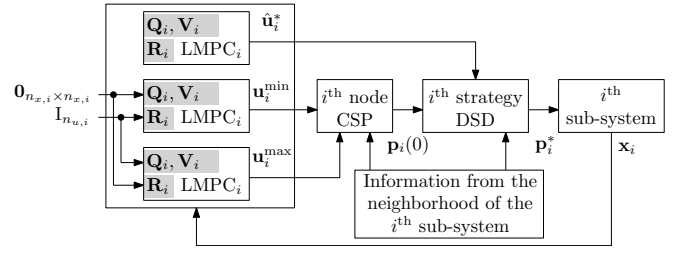


Fig. 2: General scheme for the information dependency at the i^{th} sub-system using the proposed approach.

controller computes the optimal control input $\hat{\mathbf{u}}_i^*$, and the vectors \mathbf{u}_i^{\min} and \mathbf{u}_i^{\max} are provided to the distributed CSP algorithm, which determines a feasible initial condition for the DSD. Finally, the DSD block computes the final control input that is applied to the sub-system.

5.3 Plug-and-play Property

The proposed DMPC controller has plug-and-play capabilities, which are consequence of Corollary 1 and the fact that control problems of sub-systems are independent of each other.

Assume that a new sub-system is added to the original problem in (2). Therefore, now we have $\tilde{m} + 1$ sub-systems. Notice that, under this assumption, the upper limit \tilde{m} of the sums in (2a) and (2e) is the only quantity that has to be changed in the optimization problem of the MPC formulation. Notice that for this new optimization problem, the decoupled set of optimization problems is the same as in (10a), including the optimization problem associated to the sub-system $\tilde{m} + 1$, and the bounds $\mathbf{u}_{\tilde{m}+1}^{\min}$ and $\mathbf{u}_{\tilde{m}+1}^{\max}$ can be found without requiring information from other sub-systems. Finally, a new node is added to the graph \mathcal{G} and the CSP computes a feasible initial condition for the DSD. Consequently, the proposed control scheme is plug and play since it is not necessary to modify previously already designed parts of the MPC controller in order to add a new sub-system to the overall problem. The same analysis can be done for the removal of sub-systems to the problem, but it should be taken into account that the graph \mathcal{G} should remain connected after this modification.

5.4 Control Convergence Cases

The adjacency matrix $\mathbf{A}^{(p)}$ of the graph $\mathcal{G}^{(p)}$ (see the proof of Proposition 3) depends on the amount of population in the population game. Consequently, when a constraint within the population-dynamics optimization problem is active, this might cause the disconnection of the graph (an element within the adjacency matrix $\mathbf{A}^{(p)}$ becomes null). However, this fact also depends on the topology of \mathcal{G} . The following two possible cases might occur when computing the optimal control input in a distributed way with the proposed methodology.

Case 1: $\mathbf{u}_i^* > \mathbf{u}_i^{\min}$, for all $i \in \mathcal{S}$. It is known that the optimal point $\mathbf{p}^* \in \text{int}\Delta$, the graph $\mathcal{G}^{(p)}$ is connected for all the time since \mathbf{u}^* is an interior point of \mathcal{U} , and Proposition 3 holds.

Case 2: $\mathbf{u}_i^* = \mathbf{u}_i^{\min}$, for any $i \in \mathcal{S}$. The optimal control input \mathbf{u}^* is at the edge of \mathcal{U} , i.e., there is an active constraint. Consequently, the node associated to that decision variable disappears and $\mathcal{G}^{(p)}$ might get disconnected depending on its topology. Then, each problem at each sub-complete graph $\mathcal{G}' \subset \mathcal{G}^{(p)}$ converges to an optimal solution. However, the optimal solution of the global problem is not reached. This drawback can be overcome by means of an appropriate design of the graph considering redundant links.

After having presented the proposed population-games-based DMPC controller, which is composed of two different main stages, i.e., designing of local MPC controllers, and a full-potential game

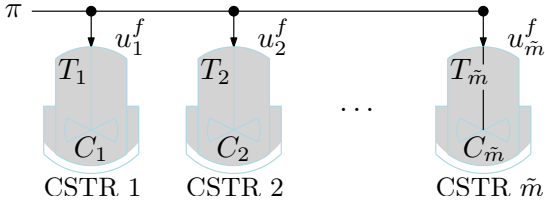


Fig. 3: Case study scheme. Set of \tilde{m} CSTRs with limited resource and information-sharing constraints.

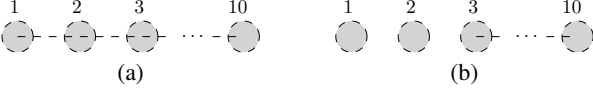


Fig. 4: Communication topology for the CSTRs case study. (a) The smaller possible connected graph representing a critic situation, and (b) The case when unplugging sub-systems 1 and 2.

with DSD, it is necessary to develop the stability analysis for the proposed approach. The stability analysis of the population-games-based DMPC is presented in Proposition 4.

Proposition 4. *The system (1) is stabilized by the proposed population-games-based DMPC controller with invariant region of attraction $\mathcal{X}_i^A = \{\mathbf{x}_i \in \mathcal{X}_i : J_i^c(\mathbf{x}_{i,k+H_p}^*) \leq \pi_i^c\}$, for all the sub-systems $i \in \tilde{\mathcal{S}}$, and considering that*

1. $\mathbf{x}_{i,0} \in \mathcal{X}_i$, for all the sub-systems $i \in \tilde{\mathcal{S}}$, and
2. Assumption 1 holds for all the sub-systems $i \in \tilde{\mathcal{S}}$.

Furthermore, given a CMPC with parameters \mathbf{Q}_i , \mathbf{R}_i , \mathbf{V}_i , and references \mathbf{r}_i , for all $i \in \tilde{\mathcal{S}}$, there is an equivalent population-games-based DMPC with parameters $\hat{\mathbf{Q}}_{i,k} = \psi_i \mathbf{Q}_i$, $\hat{\mathbf{R}}_{i,k} = \psi_i \mathbf{R}_i$, $\hat{\mathbf{V}}_{i,k} = \psi_i \mathbf{V}_i$, where $\psi_i > 0$ and the weights for fitness functions at each sub-system $\omega_i = [\mathbf{B}_{d,i}^\top \hat{\mathbf{Q}}_{i,k} \mathbf{B}_{d,i} + \hat{\mathbf{R}}_{i,k}] \in \mathbb{R}^{n_{u,i} \times n_{u,i}}$, for all $i \in \tilde{\mathcal{S}}$, i.e., $\mathbf{u}_{i,k}^* = \mathbf{p}_i^*$, for all $i \in \tilde{\mathcal{S}}$. \square

Proof: This proof is presented at the Appendix. \square

Corollary 2. *Given a CMPC with parameters $\mathbf{Q}_i = \mathbf{Q}_j$, $\mathbf{V}_i = \mathbf{V}_j$, and $\mathbf{R}_i = \mathbf{R}_j$, for all $i, j \in \tilde{\mathcal{S}}$, the equivalent population-games-based DMPC is obtained with the parameters $\hat{\mathbf{Q}}_{i,k} = \psi_{i,k} \mathbf{Q}_i$, $\hat{\mathbf{R}}_{i,k} = \psi_{i,k} \mathbf{R}_i$, $\hat{\mathbf{V}}_i(k) = \psi_{i,k} \mathbf{V}_i$, and $\omega_i = \mathbf{B}_{d,i}^\top \hat{\mathbf{Q}}_{i,k} \mathbf{B}_{d,i} + \hat{\mathbf{R}}_{i,k}$, with $\psi_{i,k} = (\mathbf{r}_i - \mathbf{x}_{i,k|k})^\top (\mathbf{r}_i - \mathbf{x}_{i,k|k})$, for all $i \in \tilde{\mathcal{S}}$. \square*

Remark 2. *Notice that the proposed DMPC approach implies more computational burden since extra tasks are required, i.e., (i) solving \tilde{m} optimization problems, (ii) a constraint satisfaction problem, and (iii) distributed evolutionary game. \square*

6 Numerical Results

6.1 Proof-of-Concept Example

The considered case study is an industrial process composed of $\tilde{m} = 10$ continuously stirred tank reactors (CSTR) as presented in Figure 3, with constraints over the information interactions for the DMPC as presented in Figure 4(a). The vector of states for each sub-system is $\mathbf{x}_i = [C_i \ T_i]^\top$, where C_i is the concentration, and T_i is the temperature inside the i^{th} CSTR, and the control input is given

by $u_i = q_i$, where q_i corresponds to the inflow. This is a proof-of-concept problem that shows the performance of the control approach proposed in this manuscript. Moreover, the methodology is scalable to any higher dimension, i.e., for systems composed of any number of sub-systems. The control objective here is to maintain the concentrations $[1 \ 0] \mathbf{x}_1, [1 \ 0] \mathbf{x}_2, [1 \ 0] \mathbf{x}_3, \dots, [1 \ 0] \mathbf{x}_9, [1 \ 0] \mathbf{x}_{10}$ as close as possible to the references $r_1 = 0.15, r_2 = 0.16, r_3 = 0.17, \dots, r_9 = 0.23, r_{10} = 0.24$ mol/l, respectively. Additionally, the system has a limited total inflow resource for the control inputs $q_1, q_2, q_3, \dots, q_{10}$ given by $\pi = 1500$ l/min. The physical constraints for the inflows are given by the range $(\underline{u}_i, \bar{u}_i) = (0, 300)$ (in l/min). It is assumed that there are local controllers guaranteeing that inflows achieve the values determined by the MPC controller, i.e., the proposed MPC controllers presented in this paper compute the references to local controllers. The discrete-time linear model with a sampling time $\tau = 0.1$ minutes, for the first CSTR around the operational point given by $C_1' = 0.0823$ mol/l, $T_1' = 442$ K, and $q_1' = 100$ mol/l is written as follows:

$$\mathbf{x}_{1,k+1} = \begin{bmatrix} 0.33 & 1.29 \times 10^{-5} \\ 0.61 & 2.45 \times 10^{-5} \end{bmatrix} \mathbf{x}_{1,k} + \begin{bmatrix} 5.49 \times 10^{-4} \\ 1.95 \times 10^{-4} \end{bmatrix} u_{1,k},$$

$$\mathbf{y}_{1,k} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \mathbf{x}_{1,k}.$$

All the sub-systems have different models and they can be expressed in terms of the first CSTR, i.e., $\mathbf{A}_{d,\ell} = 1.01\ell \mathbf{A}_{d,1}$, $\mathbf{B}_{d,\ell} = 1.01\ell \mathbf{B}_{d,1}$, for all $\ell = 2, \dots, 10$. On the other hand, the weights for the CMPC controller are given by $\mathbf{Q}_i = [10000 \ 0; 0 \ 0]$, and $\mathbf{R}_i = 1$, for all $i = 1, \dots, 10$.

Five different scenarios are presented, all of them with prediction horizon $H_p = 5$, i.e.,

Scenario 1: CMPC with weights \mathbf{Q}_i and \mathbf{R}_i , and without resource constraint, i.e., $\sum_{i=1}^{10} u_i \leq +\infty$. The optimization problem behind this CMPC controller corresponds to Problem in (2) neglecting the constraint in (2e).

Scenario 2: CMPC with weights \mathbf{Q}_i and \mathbf{R}_i , and with resource constraint, i.e., $\sum_{i=1}^{10} u_i \leq 1800$. The optimization problem behind this CMPC controller corresponds to Problem in (2).

Scenario 3: Population-games-based DMPC with weights corresponding to Proposition 4, i.e., $\hat{\mathbf{Q}}_{i,k} = \psi_i \mathbf{Q}_i$, $\hat{\mathbf{R}}_{i,k} = \psi_i \mathbf{R}_i$, for all $i = 1, \dots, 10$, with the weights in the population dynamics as in Proposition 4, i.e., $w_i = \mathbf{B}_{d,i}^\top \hat{\mathbf{Q}}_{i,k} \mathbf{B}_{d,i} + \hat{\mathbf{R}}_{i,k}$, with $\psi_{i,k} = (r_i - [1 \ 0] \mathbf{x}_{i,k|k})^2$, for all $i = 1, \dots, 10$, and with resource constraint, i.e., $\sum_{i=1}^{10} u_i \leq 1800$.

Scenario 4: Population-games-based DMPC with local MPC weights $\hat{\mathbf{Q}}_i$ and $\hat{\mathbf{R}}_i$ as in Scenario 3, with $\psi_i = 100(r_i - [1 \ 0] \mathbf{x}_{i,k|k})/r_i$, with weights in the population dynamics $w_i = \psi_i$, for all $i = 1, \dots, 10$, and with resource constraint, i.e., $\sum_{i=1}^{10} u_i \leq 1800$.

Scenario 5: Population-games-based DMPC with local MPC with parameters as in Scenario 4, and with resource constraint, i.e., $\sum_{i=1}^{10} u_i \leq 1800$. Moreover, at the interval of time between 8 minutes and 12 minutes, the sub-systems 1 and 2 are disconnected to illustrate the plug-and-play features of the proposed method (see Figure 4(b)).

6.2 Simulation Results

Evolution of system states (concentrations) is shown in Figure 5 for the ten CSTRs, for all $i \in \tilde{\mathcal{S}}$, and the five scenarios. As expected, the concentration of each CSTR achieves its corresponding set-point if the total inflow is not constrained. Otherwise (i.e., when the sum of inflows is limited to a value lower than 1800 l/min), concentrations are below their corresponding set-points since there is

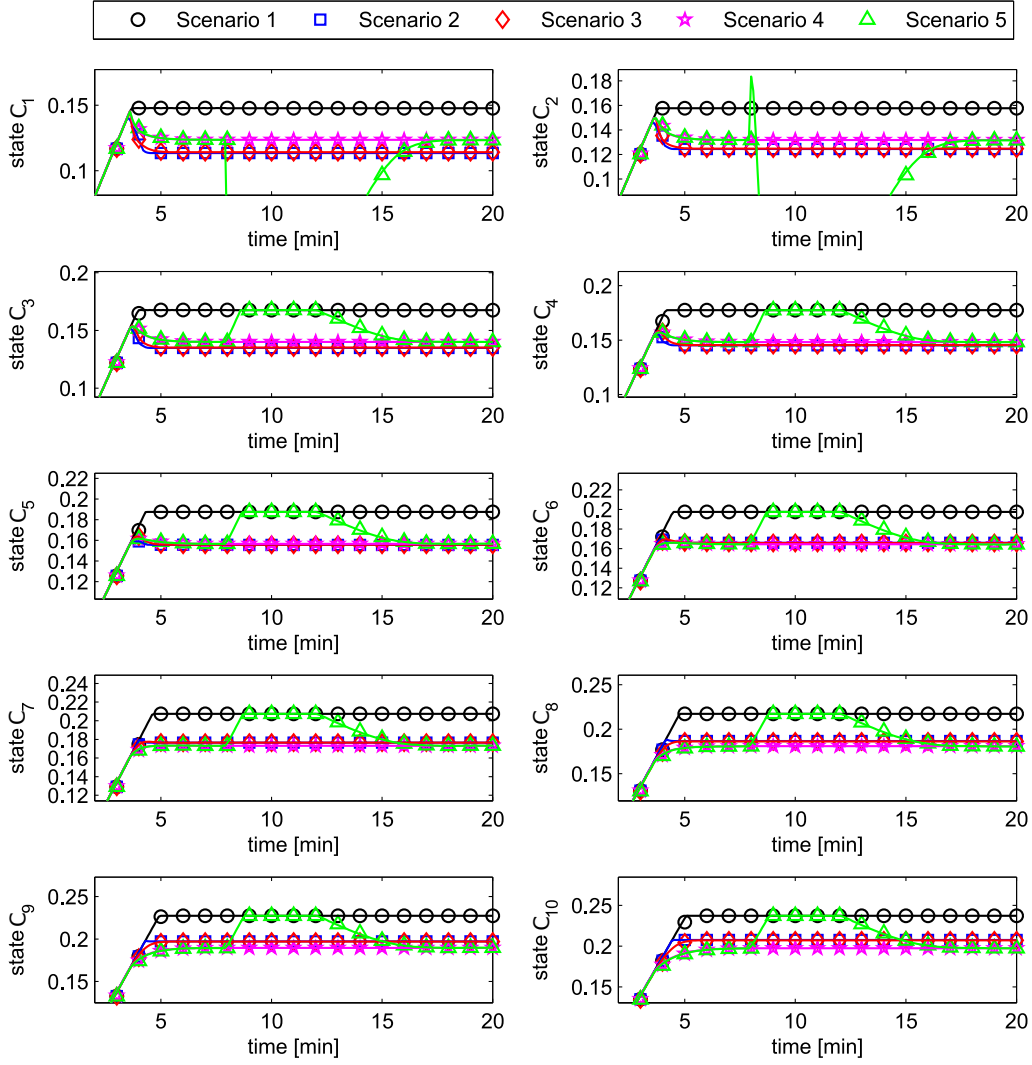


Fig. 5: Evolution of system states in [mol/l] for the case study and five different scenarios.

not enough feed-flow rate in the reactor mass balance to increase the concentration up to the desired value. Nevertheless, in the latter situation, controllers employ all the available resource to keep the controlled variables close to the desired value. Furthermore, for the fifth scenario, two sub-systems are unplugged from the system and the remaining seven sub-systems have more available resource achieving the required reference. However, once the two sub-systems are plugged into the system, then the available limited resource is newly optimally distributed. Figure 6 shows the evolution of the control inputs for the different scenarios. Additionally, the equivalence between the results obtained with Scenarios 2 and 3 as stated in Proposition 4 can be seen. Besides, notice that this equivalence between the solutions can be modified by selecting different prioritization in the distributed evolutionary-games-based algorithm, which is in charge of projecting the local controllers' solutions onto the feasible set. This modified behavior can be seen in Scenario 4. Regarding the Scenario 5, it can be seen that once the two sub-systems are unplugged from the system, then the resource that they were using is distributed throughout the other seven sub-systems, and afterwards, when the two sub-systems are plugged into the system, then the other seven sub-systems share the resource back again. Figure 7 shows the evolution of the total resource, where the satisfaction of the coupled constraint for all the scenarios is achieved.

Table 1 shows the steady-state error for the different scenarios considering the coupled constraint. The equivalence between the CMPC and the population-games-based DMPC can be seen (by comparing Scenarios 2 and 3). Besides, when weights ω_i , for all $i \in \mathcal{S}$ (see Proposition 4), are selected to be the current error, e.g.,

Table 1 Steady-state error for Scenarios 2, 3, and 4.

CSTR	Scenario 2	Scenario 3	Scenario 4
1	24.3 %	24.2 %	17.6 %
2	22.1 %	22.0 %	17.6 %
3	20.7 %	20.5 %	17.6 %
4	19.2 %	19.2 %	17.6 %
5	18.1 %	18.0 %	17.6 %
6	17.0 %	17.1 %	17.7 %
7	16.0 %	16.0 %	17.5 %
8	15.0 %	15.2 %	17.7 %
9	14.2 %	14.3 %	17.6 %
10	13.5 %	13.7 %	17.7 %

time-varying weights for the population-games-based DMPC, then an evenhanded distribution of the resource is performed achieving a stationary state where all the percentages of error are the same (see the behavior of Scenario 4). Consequently, the general distributed scheme can be tuned to the population game without modifying the local MPC controllers.

7 Conclusions

This work developed a novel methodology that use distributed population games to make an MPC controller perform under a distributed

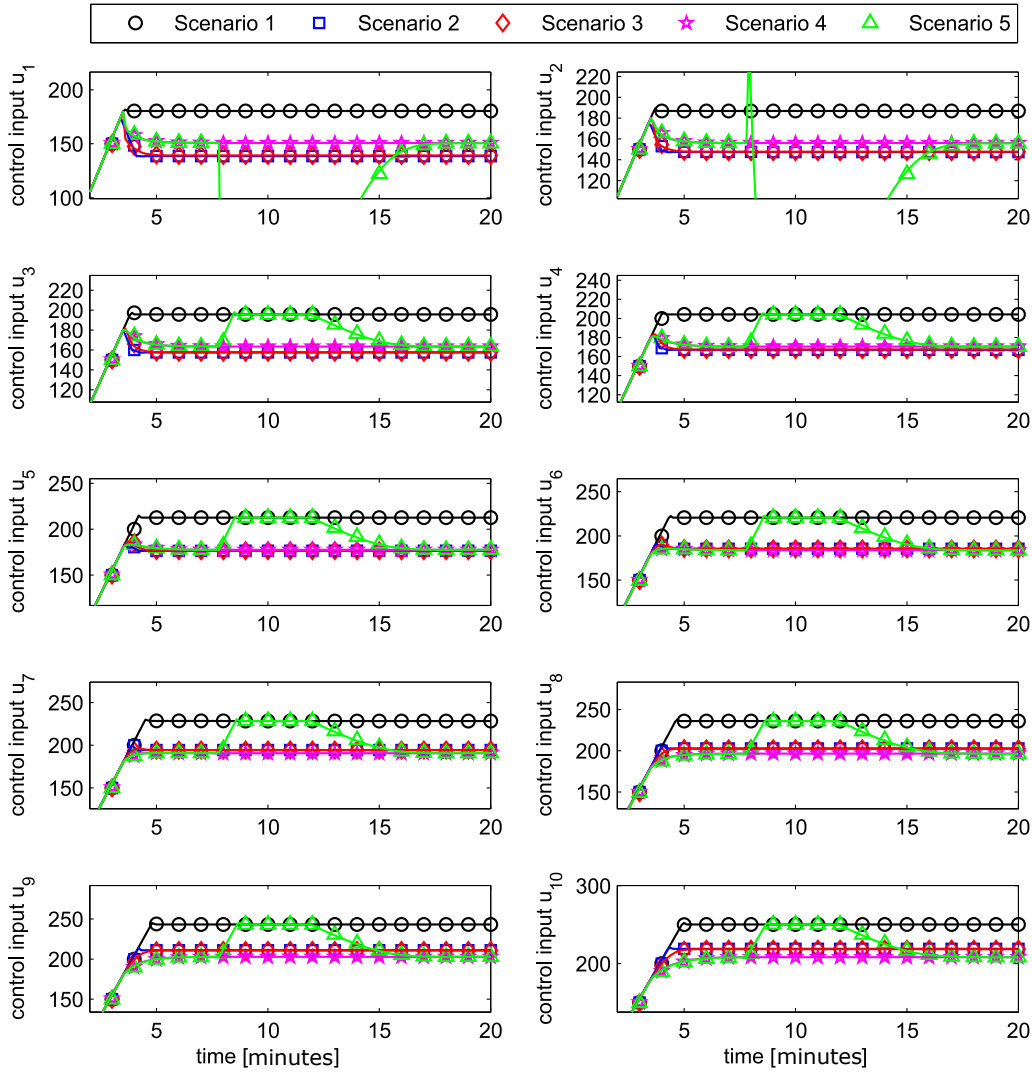


Fig. 6: Evolution of control inputs in [l/min] for the case study and five different scenarios.

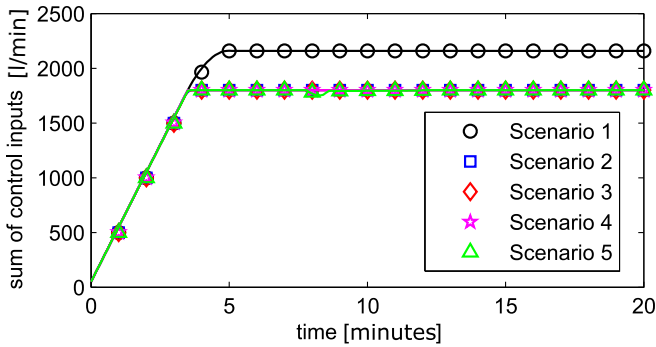


Fig. 7: Sum of control inputs for the five scenarios.

scheme with plug-and-play capabilities and considering a limited resource. Stability of the closed-loop system with the proposed DMPC is guaranteed. Results showed that the methodology satisfies the coupled constraint in a distributed way. Moreover, when selecting appropriately the potential-game weighting parameters, the same performance as a CMPC is obtained. In addition, it is shown that the same steady-state error can be achieved for all sub-systems by modifying the population-game weights and maintaining the design of the overall local MPC controllers. Furthermore, since this distributed methodology is composed by different local and independent MPC controllers, the addition or removal of sub-systems to/from the

entire system does not imply the modification of local MPC controllers. The new design is obtained by modifying only the number of strategies in the DSD and guaranteeing the communication-graph connectivity.

8 References

- 1 Y. Zheng, H. Qiu, and S. Y. Li. Networked coordination-based distributed model predictive control for large-scale system. *IEEE Transactions on Control System Technology*, 21(3):991–998, 2013.
- 2 X.J Liu, X.B. Kong, and K. Y. Lee. Distributed model predictive control for load frequency control with dynamic fuzzy valve position modeling for hydro-thermal power system. *IET Control Theory & Applications*, 10(14):1653–1664, 2016.
- 3 X.J Liu, Y. Zhang, and K. Y. Lee. Coordinated distributed mpc for load frequency control of power system with wind farms. *IEEE Transactions on Industrial Electronics*, 64(6):5140–5150, 2017.
- 4 J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Dynamical tuning for MPC using population games: A water supply network application. *ISA Transactions*, 69(2017):175–186, 2017.
- 5 J. Marden and J. S. Shamma. Game theory and distributed control. In H.P. Young and S. Zamir, editors, *Handbook of Game Theory Vol. 4*, pages 861–900. North Holland, 2014.
- 6 N. Li and J. R. Marden. Designing games for distributed optimization. *IEEE Journal of Selected Topics in Signal Processing*, 7(2):230–242, 2013. Special issue on adaptation and learning over complex networks.
- 7 Gesualdo Scutari, Daniel P Palomar, Francisco Facchinei, and Jong-shi Pang. Convex optimization, game theory, and variational inequality theory. *IEEE Signal Processing Magazine*, 27(3):35–49, 2010.
- 8 I. Bomze, M. Pelillo, and V. Stix. Approximating the maximum weight clique using replicator dynamics. *IEEE Transactions on Neural Networks*, 11(6):1228–1241, 2000.
- 9 A Pantoja, G Obando, and N Quijano. Distributed optimization with information-constrained population dynamics. *to appear in the Journal of the Franklin Institute*,

- 2018.
- 10 J. Barreiro-Gomez, N. Quijano, and C. Ocampo-Martinez. Constrained distributed optimization: A population dynamics approach. *Automatica*, 69:101–116, 2016.
 - 11 G. Obando, A. Pantoja, and N. Quijano. Building temperature control based on population dynamics. *IEEE Transactions on Control Systems Technology*, 22(1):404–412, 2014.
 - 12 J. Barreiro-Gomez and H. Tembine. Constrained evolutionary games by using a mixture of imitation dynamics. *Automatica*, 97(2018):254–262, 2018.
 - 13 J. Sanchez-Soriano. An overview on game theory applications to engineering. *International Game Theory Review*, 15(3):1340019, 2013.
 - 14 J. Barreiro-Gomez, G. Obando, and N. Quijano. Distributed population dynamics: Optimization and control applications. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(2):304–314, 2017.
 - 15 G. Bacci, S. Lasaulce, W. Saad, and L. Sanguinetti. Game theory for networks: A tutorial on game-theoretic tools for emerging signal processing applications. *IEEE Signal Processing Magazine*, 33(1):94–119, 2016.
 - 16 Yang Yang, Francisco Rubio, Gesualdo Scutari, and Daniel P Palomar. Multi-portfolio optimization: A potential game approach. *IEEE Transactions on Signal Processing*, 61(22):5590–5602, 2013.
 - 17 N. Quijano, C. Ocampo-Martinez, J. Barreiro-Gomez, G. Obando, A. Pantoja, and E. Mojica-Nava. The role of population games and evolutionary dynamics in distributed control systems. *IEEE Control Systems*, 37(1):70–97, 2017.
 - 18 J. M. Maestre and R. R. Negenborn. *Distributed model predictive control made easy*, volume 69. Springer Science & Business Media, 2013.
 - 19 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 and Chemical Engineering*, 51:21–41, 2013.
 - 20 S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends in Machine Learning*, 3(1):1–122, 2011.
 - 21 Dimitri P Bertsekas. *Nonlinear programming*. Athena scientific Belmont, 1999.
 - 22 Zheming Wang and Chong Jin Ong. Distributed model predictive control of linear discrete-time systems with local and global constraints. *Automatica*, 81:184–195, 2017.
 - 23 Arthur Richards and Jonathan P How. Robust distributed model predictive control. *International Journal of Control*, 80(9):1517–1531, 2007.
 - 24 J. Mota, J. Xavier, P. Aguiar, and M. Puschel. D-ADMM: A communication-efficient distributed algorithm for separable optimization. *IEEE Transactions on Signal processing*, 61(10):2718–2723, 2013.
 - 25 P. Trodden and A. Richards. Cooperative distributed MPC of linear systems with coupled constraints. *Automatica*, 49(2):479–487, 2013.
 - 26 A. Ferramosca, D. Limón, I. Alvarado, and E. F. Camacho. Cooperative distributed MPC for tracking. *Automatica*, 49(4):906–914, 2013.
 - 27 S. Rivero, M. Farina, and G. Ferrari-Trecate. Plug-and-play decentralized model predictive control for linear systems. *IEEE Transactions on Automatic Control*, 58(10):2608–2614, 2013.
 - 28 S. Rivero, M. Farina, and G. Ferrari-Trecate. Plug-and-play model predictive control based on robust control invariant sets. *Automatica*, 50(8):2179–2186, 2014.
 - 29 M. N. Zeilinger, Y-Pu, S. Rivero, G. Ferrari-Trecate, and C. N. Jones. Plug and play distributed model predictive control based on distributed invariance and optimization. In *Proceedings of the 52nd IEEE Conference on Decision and Control (CDC)*, pages 5770–5776, Florence, Italy, 2013.
 - 30 S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
 - 31 B. Hu and A. Linnemann. Towards infinite-horizon optimality in nonlinear model predictive control. *IEEE Transactions on Automatic Control*, 47(4):679–682, 2002.
 - 32 D. Limon, T. Alamo, and E. F. Camacho. Enlarging the domain of attraction of MPC controllers. *Automatica*, 41(4):629–635, 2005.
 - 33 D. Mayne, J. Rawlings, C. Rao, and P. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.
 - 34 W. H. Sandholm. *Population games and evolutionary dynamics*. Cambridge, Mass. MIT Press, 2010.
 - 35 J. Barreiro-Gomez, G. Obando, C. Ocampo-Martinez, and N. Quijano. Making non-centralized a model predictive control scheme by using distributed smith dynamics. In *Proceedings of the 5th IFAC Conference on Nonlinear Model Predictive Control*, pages 501–506, Seville, Spain, 2015.
 - 36 J. Maciejowski. *Predictive control: with constraints*. Pearson education, 2002.

Appendix

Proof of Proposition 4: This proof is divided into two parts. First, it is shown that there exists a CMPC whose solution is equivalent to the solution obtained with the proposed DMPC. Then, the equivalence allows to derive stability conditions of the closed-loop system controlled by using population-games-based DMPC.

First part: It is shown that there exists a relationship between the prioritization of the MPC controller in the cost function (2a), i.e., values of \mathbf{Q}_i , \mathbf{V}_i , and \mathbf{R}_i , with $i \in \tilde{S}$, and the prioritization of the DSD in the potential function, i.e., ω_i , for all $i \in \tilde{S}$ (one weight ω_i for each control input vector since $\mathbf{u}_i \in \mathbb{R}^{n_{u,i}}$) such that the optimal control input \mathbf{u}^* obtained from the solution of the centralized optimization problem \mathbf{P}_{MPC} is the same as the optimal control input \mathbf{p}^* obtained from the multi-stage distributed strategy through the

optimization problems \mathbf{P}_{LMPC} and \mathbf{P}_{DSD} . For simplicity and without loss of generality, a constant reference for each sub-system is considered, i.e., $\mathbf{r}_{i,k} = \mathbf{r}_i$, for all k .

Consider the vectors

$$\mathbf{X}_{i,k} = \begin{bmatrix} \mathbf{x}_{i,k+1|k}^\top & \mathbf{x}_{i,k+2|k}^\top & \cdots & \mathbf{x}_{i,k+H_p|k}^\top \end{bmatrix}^\top \in \mathbb{R}^{n_{x,i}H_p},$$

and

$$\mathbf{U}_{i,k} = \begin{bmatrix} \mathbf{u}_{i,k|k}^\top & \mathbf{u}_{i,k+1|k}^\top & \cdots & \mathbf{u}_{i,k+H_p-1|k}^\top \end{bmatrix}^\top \in \mathbb{R}^{n_{u,i}H_p}.$$

Then, the prediction model can be written as $\mathbf{X}_{i,k} = \Psi_i \mathbf{x}_{i,k|k} + \Theta_i \mathbf{U}_{i,k}$, and constraints of the form $\underline{\mathbf{x}}_i \leq \mathbf{x}_i \leq \bar{\mathbf{x}}_i$, and $\underline{\mathbf{u}}_i \leq \mathbf{u}_i \leq \bar{\mathbf{u}}_i$, may be compacted as $\Xi_i \mathbf{U}_{i,k} \leq \xi_{i,k}$. In order to determine the cost function in its matricial form, consider $\Phi_i = [\mathbf{r}_i^\top \cdots \mathbf{r}_i^\top]^\top$, i.e.,

$$J_k = \sum_{i=1}^{\tilde{m}} J_{i,k},$$

$$J_{i,k} = (\mathbf{X}_{i,k} - \Phi_i)^\top \tilde{\mathbf{Q}}_i (\mathbf{X}_{i,k} - \Phi_i) + \mathbf{U}_{i,k}^\top \tilde{\mathbf{R}}_i \mathbf{U}_{i,k}.$$

Defining $\mathbf{E}_{i,k} = \Phi_i - \Psi_i \mathbf{x}_{i,k|k}$, $\mathbf{G}_{i,k} = 2\Theta_i^\top \tilde{\mathbf{Q}}_i \mathbf{E}_{i,k}$, and $\mathbf{H}_i = \Theta_i^\top \tilde{\mathbf{Q}}_i \Theta_i + \tilde{\mathbf{R}}_i$, the cost function is written as*

$$J_{i,k} = \mathbf{U}_{i,k}^\top \mathbf{H}_i \mathbf{U}_{i,k} - \mathbf{U}_{i,k}^\top \mathbf{G}_{i,k} + \mathbf{E}_{i,k}^\top \tilde{\mathbf{Q}}_i \mathbf{E}_{i,k}.$$

Besides, consider the expression

$$\underbrace{\begin{bmatrix} \mathbb{1}_{n_{u,i}}^\top & & \mathbf{0} \\ & \ddots & \\ \mathbf{0} & & \mathbb{1}_{n_{u,i}}^\top \end{bmatrix}}_{\boldsymbol{\alpha}_i} \mathbf{U}_{i,k} = \begin{bmatrix} \mathbb{1}_{n_{u,i}}^\top \mathbf{u}_{i,k|k} \\ \vdots \\ \mathbb{1}_{n_{u,i}}^\top \mathbf{u}_{i,k+H_p-1|k} \end{bmatrix}.$$

1) The optimization problem \mathbf{P}_{MPC} behind the MPC controller is stated as follows:

$$\min_{\mathbf{U}_{i,k}} J_k = \sum_{i=1}^{\tilde{m}} \left\{ \mathbf{U}_{i,k}^\top \mathbf{H}_i \mathbf{U}_{i,k} - \mathbf{U}_{i,k}^\top \mathbf{G}_{i,k} + \mathbf{E}_{i,k}^\top \tilde{\mathbf{Q}}_i \mathbf{E}_{i,k} \right\}, \quad (13a)$$

subject to

$$\Xi_i \mathbf{U}_{i,k} \leq \xi_{i,k}, \quad \forall i \in \tilde{S}, \quad (13b)$$

$$\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{U}_{j,k} \leq \mathbb{1}_{[H_p]} \pi. \quad (13c)$$

For the optimization problem in (13), the following Lagrangian function is defined:

$$\bar{J}_k = \sum_{i=1}^{\tilde{m}} \left\{ J_{i,k} + \boldsymbol{\mu}_{i,k}^\top \left[\Xi_i \mathbf{U}_{i,k} - \xi_{i,k} \right] \right\} + \boldsymbol{\varepsilon}_k^\top \left[\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{U}_{j,k} - \mathbb{1}_{[H_p]} \pi \right],$$

*Notice that $\mathbf{E}_{i,k}$ is a known and constant value at each iteration since Φ_i and Ψ_i are constant and the current system state $\mathbf{x}_{i,k|k}$ is also known for all $i \in \tilde{S}$. Therefore, $\mathbf{G}_{i,k}$ is also constant at each time instant.

where $\boldsymbol{\mu}_i$, and $\boldsymbol{\varepsilon}$ are associated to the Lagrange multipliers. Then, the corresponding KKT conditions for each sub-system are written as follows:

$$2\mathbf{H}_i\mathbf{U}_{i,k}^* - \mathbf{G}_{i,k} + \boldsymbol{\mu}_{i,k}^\top \boldsymbol{\Xi}_i + \boldsymbol{\varepsilon}_k^\top \boldsymbol{\alpha}_i = \mathbf{0}, \quad (14a)$$

$$\boldsymbol{\Xi}_i \mathbf{U}_{i,k}^* \leq \boldsymbol{\xi}_{i,k}, \quad (14b)$$

$$\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{U}_{j,k}^* \leq \mathbb{1}_{[H_p]} \pi, \quad (14c)$$

$$\boldsymbol{\mu}_{i,k}^\top [\boldsymbol{\Xi}_i \mathbf{U}_{i,k}^* - \boldsymbol{\xi}_{i,k}] = \mathbf{0}, \quad (14d)$$

$$\boldsymbol{\varepsilon}_k^\top \left[\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{U}_{j,k}^* - \mathbb{1}_{[H_p]} \pi \right] = \mathbf{0}, \quad (14e)$$

$$\boldsymbol{\mu}_{i,k}, \boldsymbol{\varepsilon}_k \geq \mathbf{0}. \quad (14f)$$

2) When the minimization of the costs functions for the sub-systems considers the coupled constraint, more importance is assigned to those sub-systems with higher errors and with more prioritization weights. However, when the coupled constraint is neglected, the prioritization in the multi-objective cost function is lost. In order to take into account this effect, an auxiliary weight ψ_i^{-1} with $\psi_i > 0$ is considered at each decoupled $J_{i,k}$ for maintaining the original prioritization throughout the proof. Therefore, the cost function corresponding to the local MPC controller is given by

$$J_{i,k} = (\mathbf{X}_{i,k} - \boldsymbol{\Phi}_i)^\top \psi_i^{-1} \tilde{\mathbf{Q}}_i (\mathbf{X}_{i,k} - \boldsymbol{\Phi}_i) + \mathbf{U}_{i,k}^\top \psi_i^{-1} \tilde{\mathbf{R}}_i \mathbf{U}_{i,k}.$$

Notice that the addition of ψ_i^{-1} does not modify the optimal solution of the optimization problem whose cost function is $J_{i,k}$. Then, the optimization problem behind the local MPC controllers, \mathbf{P}_{LMPC} , are stated as follows:

$$\min_{\hat{\mathbf{U}}_{i,k}} J_{i,k} = \hat{\mathbf{U}}_{i,k}^\top \hat{\mathbf{H}}_i \hat{\mathbf{U}}_{i,k} - \hat{\mathbf{U}}_{i,k}^\top \hat{\mathbf{G}}_{i,k} + \mathbf{E}_{i,k}^\top \tilde{\mathbf{Q}}_i \mathbf{E}_{i,k}, \quad (15a)$$

subject to

$$\boldsymbol{\Xi}_i \hat{\mathbf{U}}_{i,k} \leq \boldsymbol{\xi}_{i,k}, \quad (15b)$$

where $\hat{\mathbf{G}}_{i,k} = 2\boldsymbol{\Theta}_i^\top \psi_i^{-1} \tilde{\mathbf{Q}}_i \mathbf{E}_{i,k}$, and $\hat{\mathbf{H}}_i = \boldsymbol{\Theta}_i^\top \psi_i^{-1} \tilde{\mathbf{Q}}_i \boldsymbol{\Theta}_i + \psi_i^{-1} \tilde{\mathbf{R}}_i$ (matrices $\tilde{\mathbf{Q}}_i$, and $\tilde{\mathbf{R}}_i$ are selected in a way that $\boldsymbol{\Theta}_i^\top \psi_i^{-1} \tilde{\mathbf{Q}}_i \boldsymbol{\Theta}_i + \psi_i^{-1} \tilde{\mathbf{R}}_i > 0$ to ensure that $\hat{\mathbf{H}}_i^{-1}$ exists [36]). In order to avoid confusion between the control inputs from the CMPC controller and those from the local MPC controllers, the optimal output of the local MPC controller is denoted by $\hat{\mathbf{U}}_{i,k}^*$. The related Lagrangian function is written as follows:

$$\bar{J}_{i,k} = J_{i,k} + \boldsymbol{\lambda}_{i,k}^\top [\boldsymbol{\Xi}_i \hat{\mathbf{U}}_{i,k} - \boldsymbol{\xi}_{i,k}],$$

where $\boldsymbol{\lambda}_i$ are associated to the Lagrange multipliers. Then, the corresponding KKT conditions are written as follows:

$$2\hat{\mathbf{H}}_i \hat{\mathbf{U}}_{i,k}^* - \hat{\mathbf{G}}_{i,k} + \boldsymbol{\lambda}_{i,k}^\top \boldsymbol{\Xi}_i = \mathbf{0}, \quad (16a)$$

$$\boldsymbol{\Xi}_i \hat{\mathbf{U}}_{i,k}^* \leq \boldsymbol{\xi}_{i,k}, \quad (16b)$$

$$\boldsymbol{\lambda}_{i,k}^\top [\boldsymbol{\Xi}_i \hat{\mathbf{U}}_{i,k}^* - \boldsymbol{\xi}_{i,k}] = \mathbf{0}, \quad (16c)$$

$$\boldsymbol{\lambda}_i(k) \geq \mathbf{0}. \quad (16d)$$

3) The variables of the optimization problem in the population dynamics are denoted by \mathbf{q}_k . The related problem is solved by using the DSD, i.e., \mathbf{q}_k^* is the Nash equilibrium of the full-potential population game.

Consider that $\mathbf{P}_{i,k} = [\mathbf{q}_{i,k|k}^\top \quad \mathbf{q}_{i,k+1|k}^\top \quad \cdots \quad \mathbf{q}_{i,k+H_p-1|k}^\top]^\top$. Besides, due to the fact that the proportion of agents is constrained by $\mathbf{u}_{i,k}^{\min} \leq \mathbf{q}_{i,k} \leq \mathbf{u}_{i,k}^{\max}$, then the optimization problem given by the population dynamics, \mathbf{P}_{DSD} , is stated as follows:

$$\min_{\mathbf{P}_{i,k}} f_k = \sum_{i=1}^{\tilde{m}} \left(\mathbf{P}_{i,k} - \hat{\mathbf{U}}_{i,k}^* \right)^\top \mathbf{W}_i \left(\mathbf{P}_{i,k} - \hat{\mathbf{U}}_{i,k}^* \right), \quad (17a)$$

subject to

$$\boldsymbol{\Xi}_i \mathbf{P}_{i,k} \leq \boldsymbol{\xi}_{i,k}, \quad \forall i \in \tilde{\mathcal{S}}, \quad (17b)$$

$$\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{P}_{j,k} \leq \mathbb{1}_{[H_p]} \pi, \quad (17c)$$

where the matrix \mathbf{W}_i , for all $i \in \tilde{\mathcal{S}}$, is non-singular. For the optimization problem in (17), the following Lagrangian function is defined:

$$\bar{f}_k = \sum_{i=1}^{\tilde{m}} \left\{ \left(\mathbf{P}_{i,k} - \hat{\mathbf{U}}_{i,k}^* \right)^\top \mathbf{W}_i \left(\mathbf{P}_{i,k} - \hat{\mathbf{U}}_{i,k}^* \right) + \boldsymbol{\theta}_{i,k}^\top [\boldsymbol{\Xi}_i \mathbf{P}_{i,k} - \boldsymbol{\xi}_{i,k}] \right\} + \boldsymbol{\beta}_k^\top \left[\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{P}_{j,k} - \mathbb{1}_{[H_p]} \pi \right],$$

where $\boldsymbol{\theta}_i$, and $\boldsymbol{\beta}$ are associated to the Lagrange multipliers. Then, the corresponding KKT conditions, for all sub-systems $i \in \tilde{\mathcal{S}}$, are written as follows:

$$2\mathbf{W}_i \mathbf{P}_i^* - 2\mathbf{W}_i \hat{\mathbf{U}}_{i,k}^* + \boldsymbol{\theta}_{i,k}^\top \boldsymbol{\Xi}_i + \boldsymbol{\beta}_k^\top \boldsymbol{\alpha}_i = \mathbf{0}, \quad (18a)$$

$$\boldsymbol{\Xi}_i \mathbf{P}_{i,k}^* \leq \boldsymbol{\xi}_{i,k}, \quad (18b)$$

$$\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{P}_{j,k}^* \leq \mathbb{1}_{[H_p]} \pi, \quad (18c)$$

$$\boldsymbol{\theta}_{i,k}^\top [\boldsymbol{\Xi}_i \mathbf{P}_{i,k}^* - \boldsymbol{\xi}_{i,k}] = \mathbf{0}, \quad (18d)$$

$$\boldsymbol{\beta}_k^\top \left[\sum_{j=1}^{\tilde{m}} \boldsymbol{\alpha}_j \mathbf{P}_{j,k}^* - \mathbb{1}_{[H_p]} \pi \right] = \mathbf{0}, \quad (18e)$$

$$\boldsymbol{\theta}_{i,k}, \boldsymbol{\beta}_k \geq \mathbf{0}. \quad (18f)$$

From (18a), it is obtained that

$$\hat{\mathbf{U}}_{i,k}^* = \mathbf{P}_i^* + \frac{1}{2} \mathbf{W}_i^{-1} \boldsymbol{\theta}_{i,k}^\top \boldsymbol{\Xi}_i + \frac{1}{2} \mathbf{W}_i^{-1} \boldsymbol{\beta}_k^\top \boldsymbol{\alpha}_i. \quad (19)$$

Now, replacing (19) in (16a), it is obtained

$$2\hat{\mathbf{H}}_i \mathbf{P}_i^* - \hat{\mathbf{G}}_{i,k} + \left(\hat{\mathbf{H}}_i \mathbf{W}_i^{-1} \boldsymbol{\theta}_{i,k}^\top + \boldsymbol{\lambda}_{i,k}^\top \right) \boldsymbol{\Xi}_i + \hat{\mathbf{H}}_i \mathbf{W}_i^{-1} \boldsymbol{\beta}_k^\top \boldsymbol{\alpha}_i = \mathbf{0}.$$

Notice that if the weights for the local MPC controllers (denoted by $\tilde{\mathbf{Q}}_i$, $\tilde{\mathbf{R}}_i$, and $\tilde{\mathbf{V}}_i$) in the optimization problem are selected to be $\tilde{\mathbf{Q}}_i = \psi_i \mathbf{Q}_i$, $\tilde{\mathbf{R}}_i = \psi_i \mathbf{R}_i$, and $\tilde{\mathbf{V}}_i = \psi_i \mathbf{V}_i$, for all $i \in \tilde{\mathcal{S}}$, then $\hat{\mathbf{H}}_i = \mathbf{H}_i$, and $\hat{\mathbf{G}}_{i,k} = \mathbf{G}_{i,k}$. In addition, if \mathbf{W}_i is selected to be $\hat{\mathbf{H}}_i$, for all $i \in \tilde{\mathcal{S}}$, then it follows that

$$2\mathbf{H}_i \mathbf{P}_i^* - \mathbf{G}_{i,k} + (\boldsymbol{\theta}_{i,k} + \boldsymbol{\lambda}_{i,k})^\top \boldsymbol{\Xi}_i + \boldsymbol{\beta}_k^\top \boldsymbol{\alpha}_i = \mathbf{0},$$

this condition is the same as (14a), and \mathbf{P}_i^* satisfies constraints (18b), and (18c). Therefore, the equivalence with the solution of the centralized optimization problem \mathbf{P}_{MPC} is shown. Due to the fact that only the first control input can be applied to the system, then it yields

$\omega_i = \mathbf{B}_{d,i}^\top \hat{\mathbf{Q}}_{i,k} \mathbf{B}_{d,i} + \hat{\mathbf{R}}_{i,k}$, for all $i \in \tilde{\mathcal{S}}$.

Second part: Now, it is considered the optimal cost function of the CMPC controller as a Lyapunov function and it is proceeded as in [31]. The cost is denoted by $J_k = \sum_{i=1}^{\tilde{m}} J_{i,k}$, and the optimal cost is denoted by $J_k^* = \sum_{i=1}^{\tilde{m}} J_{i,k}^*$. At time instant k , $\tilde{\mathbf{u}}_{i,k}^* = (\mathbf{u}_{i,k|k}^*, \dots, \mathbf{u}_{i,k+H_p-1|k}^*)$ is the optimal control sequence for the i^{th} sub-system. Similarly, the optimal control sequence at time instant $k+1$ for the i^{th} sub-system is given by $\tilde{\mathbf{u}}_{i,k+1}^* = (\mathbf{u}_{i,k+1|k+1}^*, \dots, \mathbf{u}_{i,k+H_p|k+1}^*)$. Furthermore, there are feasible control sequences given by

$$\tilde{\mathbf{u}}_{i,k} = (\mathbf{u}_{i,k|k}^*, \mathbf{u}_{i,k+1|k+1}^*, \dots, \mathbf{u}_{i,k+H_p-1|k+1}^*), \text{ and,}$$

$$\tilde{\mathbf{u}}_{i,k+1} = (\mathbf{u}_{i,k+1|k}^*, \mathbf{u}_{i,k+2|k}^*, \dots, \mathbf{u}_{i,k+H_p-1|k}^*, \mathbf{u}_{i\text{MPC},k+H_p|k}^*).$$

The four previously introduced control sequences generate the costs J_k^* , J_{k+1}^* , J_k , and J_{k+1} , respectively. Then,

$$J_{i,k}^* = J_i^f(\mathbf{x}_{i,k+H_p|k}) + \sum_{j=0}^{H_p-1} J_i^\ell(\mathbf{x}_{i,k+j|k}, \mathbf{u}_{i,k+j|k}^*),$$

$$J_{i,k+1}^* = J_i^f(\mathbf{x}_{i,k+H_p+1|k+1}) + \sum_{j=1}^{H_p} J_i^\ell(\mathbf{x}_{i,k+j|k+1}, \mathbf{u}_{i,k+j|k+1}^*),$$

$$J_{i,k} = J_i^f(\mathbf{x}_{i,k+H_p|k+1}) + J_i^\ell(\mathbf{x}_{i,k|k}, \mathbf{u}_{i,k|k}^*)$$

$$+ \sum_{j=1}^{H_p-1} J_i^\ell(\mathbf{x}_{i,k+j|k+1}, \mathbf{u}_{i,k+j|k+1}^*),$$

$$J_{i,k+1} = J_i^f(\mathbf{x}_{i,k+H_p+1|k}) + \sum_{j=1}^{H_p-1} J_i^\ell(\mathbf{x}_{i,k+j|k}, \mathbf{u}_{i,k+j|k}^*)$$

$$+ J_i^\ell(\mathbf{x}_{i,k+H_p|k}, \mathbf{u}_{i\text{MPC},k+H_p|k}^*).$$

Notice that $J_k^* \leq J_k$, and $J_{k+1}^* \leq J_{k+1}$. Consequently $J_{k+1}^* + J_k^* \leq J_{k+1} + J_k$, and

$$J_{k+1}^* - J_k \leq J_{k+1} - J_k^*. \quad (20)$$

The terms in (20) are

$$J_{k+1}^* - J_k = \sum_{i=1}^{\tilde{m}} \left\{ J_i^\ell(\mathbf{x}_{i,k+H_p|k+1}, \mathbf{u}_{i,k+H_p|k+1}^*) - J_i^\ell(\mathbf{x}_{i,k|k}, \mathbf{u}_{i,k|k}^*) + J_i^f(\mathbf{x}_{i,k+H_p+1|k+1}) - J_i^f(\mathbf{x}_{i,k+H_p|k+1}) \right\},$$

$$J_{k+1} - J_k^* = \sum_{i=1}^{\tilde{m}} \left\{ J_i^\ell(\mathbf{x}_{i,k+H_p|k}, \mathbf{u}_{i\text{MPC},k+H_p|k}^*) - J_i^\ell(\mathbf{x}_{i,k|k}, \mathbf{u}_{i,k|k}^*) + J_i^f(\mathbf{x}_{i,k+H_p+1|k}) - J_i^f(\mathbf{x}_{i,k+H_p|k}) \right\}.$$

Replacing in (20) and removing the term $J_i^\ell(\mathbf{x}_{i,k|k}, \mathbf{u}_{i,k|k}^*)$ at both sides from (20), it follows that:

$$\sum_{i=1}^{\tilde{m}} \left\{ J_i^\ell(\mathbf{x}_{i,k+H_p|k+1}, \mathbf{u}_{i,k+H_p|k+1}^*) + \right.$$

$$\left. J_i^f(\mathbf{x}_{i,k+H_p+1|k+1}) - J_i^f(\mathbf{x}_{i,k+H_p|k+1}) \right\}$$

$$\leq \sum_{i=1}^{\tilde{m}} \left\{ J_i^\ell(\mathbf{x}_{i,k+H_p|k}, \mathbf{u}_{i\text{MPC},k+H_p|k}^*) + J_i^f(\mathbf{x}_{i,k+H_p+1|k}) - J_i^f(\mathbf{x}_{i,k+H_p|k}) \right\} \quad (21)$$

$$\leq 0, \text{ (according to Assumption 1).}$$

The following steps of the proof are developed following the analysis presented in [31]. Suppose first that $\mathbf{x}_{i,k+H_p|k+1} = \mathbf{x}_i^s$, then the optimality implies that $\mathbf{x}_{i,k+H_p+1|k+1} = \mathbf{x}_i^s$ and it is concluded that $\mathbf{x}_{i,k+H_p+1|k+1} \in \mathcal{X}_i^f$. The other option occurs when $\mathbf{x}_{i,k+H_p|k+1} \neq \mathbf{x}_i^s$, in which case, using (21) and the fact that $J_i^\ell(\mathbf{x}_{i,k+H_p|k}, \mathbf{u}_{i,k+H_p|k}^*) > 0$, then

$$J_i^f(\mathbf{x}_{i,k+H_p+1|k+1}) < J_i^f(\mathbf{x}_{i,k+H_p|k+1}), \quad \forall i \in \tilde{\mathcal{S}}.$$

According to (4), the only possibility to satisfy the strict inequality is that $J_i^f(\mathbf{x}_{i,k+H_p+1|k+1}) = J_i^c < \pi_i^c$, and $J_i^f(\mathbf{x}_{i,k+H_p|k+1}) = \pi_i^c$, for all $i \in \tilde{\mathcal{S}}$. This fact implies that $\mathbf{x}_{i,k+1|k+1} \in \mathcal{X}_i^A$, where

$$\mathcal{X}_i^A = \left\{ \mathbf{x}_{i,k} \in \mathcal{X}_i : J_i^c(\mathbf{x}_{i,k+H_p|k}) \leq \pi_i^c \right\}.$$

Hence, it is concluded that \mathcal{X}_i^A is an invariant set. In order to prove the stability, it is recalled the fact that $J_{k+1}^* \leq J_{k+1}$, which implies that $J_{k+1}^* - J_k^* \leq J_{k+1} - J_k^*$, i.e.,

$$J_{i,k+1}^* - J_{i,k}^* \leq J_i^f(\mathbf{x}_{i,k+H_p+1|k}) - J_i^f(\mathbf{x}_{i,k+H_p|k}) + J_i^\ell(\mathbf{x}_{i,k+H_p|k}, \mathbf{u}_{i\text{MPC},k+H_p|k}^*) - J_i^\ell(\mathbf{x}_{i,k|k}, \mathbf{u}_{i,k|k}^*), \quad \forall i \in \tilde{\mathcal{S}}.$$

Using Assumption 1, it is concluded that

$$J_{k+1}^* - J_k^* \leq - \sum_{i=1}^{\tilde{m}} J_i^\ell(\mathbf{x}_{i,k|k}, \mathbf{u}_{i,k|k}^*).$$

The cost function J_k^* is a decaying sequence. Hence the system (1) controlled by using a CMPC controller is stable. As a conclusion, according to the first part of the proof, there exists a CMPC controller whose solution is equal to the solution computed by the proposed population-games-based DMPC controller. Moreover, since the system controlled with the equivalent CMPC controller is stable according to the second part of the proof, then the system controlled with population-games-based DMPC is also stable. \square