Dynamical Tuning for MPC using Population Games: A Water Supply Network Application

Julian Barreiro-Gomez [a,b,⋆], Carlos Ocampo-Martinez [a], and Nicanor Quijano [b]

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

[b] *Departamento de Ingeniería Eléctrica y Electrónica, Universidad de los Andes, Carrera 1 No 18A-10, Bogotá, Colombia*

Abstract

Model predictive control (MPC) is a suitable strategy for the control of large-scale systems that have multiple design requirements, e.g., multiple physical and operational constraints. Besides, an MPC controller is able to deal with multiple control objectives considering them within the cost function, which implies to determine a proper prioritization for each of the objectives. Furthermore, when the system has time-varying parameters and/or disturbances, the appropriate prioritization might vary along the time as well. This situation leads to the need of a dynamical tuning methodology. This paper addresses the dynamical tuning issue by using evolutionary game theory. The advantages of the proposed method are highlighted and tested over a large-scale water supply network with periodic time-varying disturbances. Finally, results are analyzed with respect to a multi-objective MPC controller that uses static tuning.

Key words: Dynamical tuning, model predictive control, game theory, large-scale systems, water supply networks

1 Introduction

Model predictive control (MPC) is one of the most used control techniques in industrial applications because of its versatility to deal with multiple design requirements. The MPC controller is an optimization-based technique that computes an optimal control sequence that minimizes a multi-objective cost function subject to physical and/or operational constraints. However, multiple control objectives imply to assign a prioritization weight for each objective. The task of finding the appropriate set of the aforementioned weights is known as the MPC tuning problem. In many cases, the tuning procedure is determined intuitively depending on the engineering application, or the adequate weights are found by a trial-and-error procedure. Furthermore, applications of large-scale nature, the consideration of a large number of constraints, and/or the need of including several control objectives make even more complex to determine the appropriate values for the MPC tuning weights. Therefore, the necessity of developing self-tuning methodologies has arisen. Additionally, when having time-varying parameters, disturbances and/or nominal conditions within the system, the appropriate tuning may also vary through time.

The tuning problem has been discussed by many authors and by using different approaches. A general review about different on-line and off-line tuning approaches for MPC controllers is presented in [6]. An alternative to determine the appropriate tuning of MPC controllers is by matching the MPC performance with the performance of a pre-established controller. For instance, in [5] the tuning of an MPC controller is computed based on a matching to a desired reference controller, then weights are adjusted in order to obtain a behavior close to the performance of the mentioned reference controller. Afterwards, an extension of this approach has been presented in [30]. In [27],

⋆ Corresponding author
Email addresses: j.barreiro135@uniandes.edu.co - jbarreiro@iri.upc.edu (Julian Barreiro-Gomez), cocampo@iri.upc.edu (Carlos Ocampo-Martinez), nquijano@uniandes.edu.co (Nicanor Quijano).

the matching to a linear controller is also used to determine the values of the MPC parameters for multiple-input-multiple-output systems. Authors in [20] present a tuning methodology for the weights of an MPC controller in the frequency domain using also control matching. In [35], an automatic tuning strategy is proposed consisting of a controller and a state observer. In [1], a tuning strategy is studied with an optimization algorithm, which uses an approximation between both a closed-loop predicted output and the parameters that can be adjusted in the MPC controller, and in [26] an optimal tuning of MPC policies with simultaneous perturbation stochastic approximation is presented. Other perspectives to solve the problem without the use of a reference model have emerged. For instance, in [29] it is proposed to compute several points of the Pareto front associated to the cost function in a multi-objective MPC controller. Then, a pre-established management point allows to determine the desired value within the Pareto front from which the appropriate tuning weights are determined. In [40], the system output is controlled to maintain it within a region instead of achieving a reference point. Therefore, weights are selected to penalize the output error with respect to a zone for a crude distillation unit. Besides, heuristic directions have also been used to determine the appropriate tuning in an MPC controller as in [38]. Moreover, in [32] and [8] the authors use neural networks and fuzzy-based decision making to establish a tuning in an MPC controller, illustrating examples for a mixing tank and for water networks, respectively. Further methods have been explored in the tuning task. In [36], a two-step off-set free tuning procedure is proposed. At first stage, the setup of a nominal MPC loop is made, and then the second step is in charge of adapting the external reference. In [31], a systematic tuning procedure is presented by using multi-objective optimization methods; in [10], a robust tuning problem for a two-degree-of-freedom MPC is presented for single-input-single-output system; and authors in [15] have presented a self-tuning of the terminal cost in an economic MPC controller.

On the other hand, game theory has gotten special importance in the last years for the design of control and decision-making algorithms. A general view about the role of game theory in distributed control is presented in [13]. It is shown that game theory is quite suitable to achieve global objectives by setting local rules. Furthermore, evolutionary-game theory allows to model the evolution of agents when they interact strategically in a population [37], [25]. In the evolution process of the population, each rational agent makes rational decisions in order to pursue an improvement over its benefits until reaching a scenario where it is not possible to obtain an enhancement by unilaterally making a decision (this situation is given by a Nash equilibrium). Besides, evolutionary game theory allows to design systems that guarantee convergence to a Nash equilibrium. Additionally, there is a close relationship between the Nash equilibrium with a maximum point in a concave constrained optimization problem due to the fact that under certain conditions the Nash equilibrium satisfies the Karush-Kuhn-Tucker (KKT) first-order conditions [25], making evolutionary-game theory a powerful tool to address optimization-based control design. For instance, in [2], [4], [12], [17], [21], [22], [23], [24], and [28], a game-theoretical approach has been presented for optimization and/or control purposes. Given the suitability of game theory in control applications and its relationship with optimization, this paper proposes a dynamical tuning methodology based on evolutionary game theory.

The contribution of this paper is a novel methodology for the on-line dynamical tuning of a multi-objective MPC controller based on evolutionary game theory. The method consists of a normalization of the cost function associated to the optimization problem that the MPC controller solves to determine the optimal control inputs at each time instant, and a population game that fixes the appropriate set of prioritization weights according to a desired region over the Pareto front known as *management region*. Furthermore, the method establishes a weighted sum, i.e., the sum of all weights should be equal to one [7]. The population game is solved by using a discrete version of the projection dynamics, which converge to a Nash equilibrium. It is shown that the projection dynamics satisfy the constraint given by the weighted sum, and the stability analysis of the Nash equilibrium under the discrete projection dynamics is formally presented. Some of the aforementioned previous works related to the tuning problem require either a reference controller or an observer, e.g., [5],[27], and [35]. Differently, the proposed method, based on population dynamics, do not require a reference controller. Moreover, other control strategies need to compute several points in the Pareto front in order to select an appropriate prioritization, which implies a high computational burden, e.g., [29]. As an advantage, the proposed method does not require to generate multiple points within the Pareto front associated to the multi-objective cost function in an MPC controller. Furthermore, most of the tuning techniques are static and performed off-line as part of a design procedure. Nevertheless, the proposed tuning methodology can continuously adjust the prioritization of the control objectives to maintain the system operating within the desired *management region*. In order to illustrate the enhancement over the performance of an MPC controller using the dynamical population-games-based tuning, the proposed methodology is applied to a large-scale water supply network. The results are analyzed and compared with respect to a multi-objective MPC controller with static tuning.

The remainder of this paper is organized as follows. Section 2 introduces the background associated to multi-objective predictive control and population games. This section also introduces a discrete version of the projection dynamics and formally presents their properties. Section 3 presents the proposed dynamical tuning based on population games, explaining in detail its different steps (normalization and dynamical weighting procedure). Then, Section 4 introduces the water supply network application, its control objectives, and motivates the necessity for implementing a dynamical tuning strategy. Furthermore, this section compares the results of a predictive controller with standard static tuning with respect to results when implementing the proposed dynamical tuning. Simulation results are analyzed and discussed highlighting the enhancement of the performance when adopting the dynamical tuning based on population games. Finally, concluding remarks are drawn in Section 5.

Notation

All column vectors are denoted by bold style, e.g., $\mathbf{x}$. Matrices are denoted by bold upper case, e.g., $\mathbf{A}$. In contrast, scalars are denoted by non-bold style, e.g., n. The sets are denoted by calligraphic upper case, e.g., $\mathcal{S}$. The norm $\|\mathbf{x}\|$ of the vector $\mathbf{x} \in \mathbb{R}^{n_x}$ is defined as $\|\mathbf{x}\| = \sqrt{\mathbf{x}^\top \mathbf{x}}$. The identity matrix of size $n \times n$ is denoted by $\mathbb{I}_n$, $\mathbb{1}_n$ is the column vector with n unitary entries, i.e., $\mathbb{1}_n = [1 \quad \dots \quad 1]^\top \in \mathbb{R}^n$, the vector of null entries and suitable dimensions is denoted by $\mathbf{0}$, and $\text{diag}(\mathbf{p})$ is the diagonal matrix of the vector $\mathbf{p}$. Finally, real numbers are denoted by $\mathbb{R}$, all the non-negative numbers are denoted by $\mathbb{R}_{\geq 0}$, and all the strictly positive real numbers are denoted by $\mathbb{R}_{>0}$. Similarly, the integer numbers, non-negative integer numbers, and the strictly positive integer numbers are denoted by $\mathbb{Z}$, $\mathbb{Z}_{\geq 0}$, and $\mathbb{Z}_{>0}$, respectively. Throughout this document, both continuous- and discrete-time systems are treated. Therefore, $k \in \mathbb{Z}_{\geq 0}$ denotes that the system is described in discrete time, whereas the use of time denoted by t in the continuous-time expressions is mostly omitted in order to simplify the notation. Regarding the discrete time notation for the MPC controller, $\mathbf{x}(k+j|k)$ denotes the prediction made at time k of the vector $\mathbf{x}$ for time $k+j$, where $k, j \in \mathbb{Z}_{\geq 0}$, i.e., in the argument $(k+j|k)$, the first element $k+j$ indicates discrete time for prediction, whereas the second element k indicates the actual discrete time.

2 Background

Prior to presenting the proposed population-games-based dynamical tuning, it is necessary to introduce some preliminary concepts that are used throughout the paper. First, some preliminaries related to the multi-objective MPC design and its corresponding optimization problem statement are introduced. Moreover, the mathematical formalism associated to the population dynamics, the discrete version of the projection dynamics, and their properties are shown and analyzed.

2.1 Multi-objective model predictive control

Consider a system whose dynamics are represented by the following discrete-time state-space model:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{B}_d\mathbf{d}(k), \tag{1}$$

where $k \in \mathbb{Z}_{\geq 0}$ denotes the discrete time. The vector $\mathbf{x} \in \mathbb{R}^{n_x}$ denotes the system states, $\mathbf{u} \in \mathbb{R}^{n_u}$ denotes the vector of control inputs, $\mathbf{d} \in \mathbb{R}^{n_d}$ corresponds to the vector of disturbances affecting the system, and $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{B}_d$ are the system matrices of suitable dimensions. System states and control inputs are constrained because of physical and/or desired operational limits. These constraints are established by defining the following feasible sets:

$$\mathcal{X} \triangleq \{\mathbf{x} \in \mathbb{R}^{n_x} : \mathbf{G}\mathbf{x} \leq \mathbf{g}\}, \tag{2a}$$

$$\mathcal{U} \triangleq \{\mathbf{u} \in \mathbb{R}^{n_u} : \mathbf{H}\mathbf{u} \leq \mathbf{h}\}, \tag{2b}$$

where $\mathbf{G}, \mathbf{g}, \mathbf{H}$, and $\mathbf{h}$ are matrices and vectors of suitable dimensions to represent the constraints for the system states and control inputs, respectively. Let $\hat{\mathbf{u}}(k)$ be a sequence of feasible control inputs within a pre-establish prediction horizon denoted by $H_p \in \mathbb{Z}_{>0}$. Similarly, let $\hat{\mathbf{x}}(k)$ be the sequence of feasible system states when applying the control input sequence $\hat{\mathbf{u}}(k)$ to the system (1). Finally, let $\hat{\mathbf{d}}(k)$ be the forecasting of the disturbances as in [9],[33], and [34]. Hence,

$$\hat{\mathbf{u}}(k) \triangleq \{\mathbf{u}(k|k), \mathbf{u}(k+1|k), ..., \mathbf{u}(k+H_p-1|k)\}, \tag{3a}$$

$$\hat{\mathbf{x}}(k) \triangleq \{\mathbf{x}(k+1|k), \mathbf{x}(k+2|k), ..., \mathbf{x}(k+H_p|k)\}, \tag{3b}$$

$$\hat{\mathbf{d}}(k) \triangleq \{\mathbf{d}(k|k), \mathbf{d}(k+1|k), ..., \mathbf{d}(k+H_p-1|k)\}. \tag{3c}$$

The system (1) is controlled by a multi-objective MPC controller with $n \geq 2$ control objectives. The optimization problem behind the MPC controller is as follows:

$$\underset{\hat{\mathbf{u}}}{\text{minimize}}\, J\left(\mathbf{x}(k), \mathbf{d}(k), \mathbf{u}\right) = \sum_{j=1}^{n} \gamma_j J_j(\mathbf{x}(k), \mathbf{d}(k), \mathbf{u}), \tag{4a}$$

subject to:

$$\mathbf{x}(k+i+1|k) = \mathbf{A}\mathbf{x}(k+i|k) + \mathbf{B}\mathbf{u}(k+i|k) + \mathbf{B}_l\mathbf{d}(k+i|k), \;\; i \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}, \tag{4b}$$

$$\mathbf{u}(k+i|k) \in \mathcal{U}, \;\; i \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}, \tag{4c}$$

$$\mathbf{x}(k+i|k) \in \mathcal{X}, \;\; i \in [1, H_p] \cap \mathbb{Z}_{\geq 0}, \tag{4d}$$

where $\mathbf{x}(k|k) \in \mathbb{R}^{n_x}$ is the current measured state, and $\gamma_j \in \mathbb{R}_{\geq 0}$, with $j = 1, \ldots, n$, are the n prioritization weights in the cost function $J(\mathbf{x}(k), \mathbf{u})$ satisfying that $\sum_{j=1}^{n} \gamma_j = 1$. Assuming that the optimization problem (4) is feasible, its solution is an optimal control input sequence denoted by $\hat{\mathbf{u}}^*(k)$, i.e.,

$$\hat{\mathbf{u}}^*(k) \triangleq \{\mathbf{u}^*(k|k), \mathbf{u}^*(k+1|k), ..., \mathbf{u}^*(k+H_p-1|k)\}.$$

Therefore, it follows that the controller may only apply the first control input from the optimal sequence, which is given by $\mathbf{u}^*(k) \triangleq \mathbf{u}^*(k|k)$. Then, after having applied the optimal control input to the system (1), a new state $\mathbf{x}(k+1)$ is measured and the procedure is repeated in order to determine the optimal sequence $\hat{\mathbf{u}}^*(k+1)$ from which the control input $\mathbf{u}^*(k+1)$ is obtained.

2.2 Population games

Consider a large and finite number of rational agents pursuing an improvement of their benefit within a population. It is assumed that each agent has the chance to select from a set of $n \geq 2$ available strategies from the set $\mathcal{S} = \{1, \ldots, n\}$. Making the analogy with the optimization problem behind the MPC controller (4), each control objective is associated to a strategy. The scalar $p_i \in \mathbb{R}_{\geq 0}$ represents the proportion of agents selecting the strategy $i \in \mathcal{S}$, and the vector $\mathbf{p} \in \mathbb{R}^n_{\geq 0}$ represents the strategic distribution of agents, i.e., $\mathbf{p} = [p_1 \;\; \ldots \;\; p_n]^\top$. Since each p_i, for all $i \in \mathcal{S}$, represents a proportion of agents, then it should be satisfied that $\sum_{j \in \mathcal{S}} p_j = 1$. Therefore, all the possible strategic distributions in the population are given by a simplex set

$$\Delta = \left\{ \mathbf{p} \in \mathbb{R}^n_{\geq 0} : \sum_{j \in \mathcal{S}} p_j = 1 \right\}. \tag{5}$$

The incentives that agents have in order to switch from one strategy to another one are determined by a fitness function whose mapping is $f_i : \Delta \mapsto \mathbb{R}$, for all $i \in \mathcal{S}$, i.e., $f_i(\mathbf{p})$ receives a strategic distribution of the population, and returns the benefits that the proportion of agents p_i obtains for selecting strategy $i \in \mathcal{S}$. Therefore, the vector of fitness functions $\mathbf{f} : \Delta \mapsto \mathbb{R}^n$ is a function that receives strategic distribution and returns the benefits for all the proportions in the population, i.e., $\mathbf{f}(\mathbf{p}) = [f_1(\mathbf{p}) \;\; \ldots \;\; f_n(\mathbf{p})]^\top$. In this regard, notice that agents stop switching among strategies once they do not have more incentives to do so. This situation is achieved at an equilibrium point known as the Nash equilibrium introduced in Definition 1 [25].

Definition 1 *A population state $\mathbf{p}^* \in \Delta$ is a Nash equilibrium if each used strategy entails the maximum benefit for the proportion of agents that chooses it. Equivalently, the Nash equilibrium $\mathbf{p}^* \in \Delta$ is given by the condition that $p_i^* > 0 \Rightarrow f_i(\mathbf{p}^*) \geq f_j(\mathbf{p}^*)$, for all $i, j \in \mathcal{S}$.*

The framework for the population games in this paper is given by full-potential and stable games. These two classes of population games allow to guarantee the stability of the Nash equilibrium under population dynamics. Full-potential and stable games are introduced in Definitions 2 and 3, respectively [25].

Definition 2 *The game $\mathbf{f}(\mathbf{p})$ is a full-potential game if there exists a continuous differentiable function $V(\mathbf{p})$, known as potential function, satisfying that $\frac{\partial V(\mathbf{p})}{\partial p_i} = f_i(\mathbf{p})$, for all $i \in \mathcal{S}$, $\mathbf{p} \in \Delta$. Then, a full-potential game is generated from a known potential differentiable function $V(\mathbf{p})$.* ◊

Definition 3 *The population game* $\mathbf{f} : \Delta \mapsto \mathbb{R}^n$ *is a stable game if*

$$(\mathbf{p} - \mathbf{q})^\top (\mathbf{f}(\mathbf{p}) - \mathbf{f}(\mathbf{q})) \leq 0, \text{ for all } \mathbf{p}, \mathbf{q} \in \Delta. \tag{6}$$

This condition is equivalent to the condition that $D\mathbf{f}(\mathbf{p})$ *is negative semidefinite, where* $[D\mathbf{f}(\mathbf{p})]_{ij} = \frac{\partial f_i(\mathbf{p})}{\partial p_j}$. ◊

2.2.1 Projection dynamics

The projection dynamics are one of the six fundamental population dynamics [2][23][25], which have been introduced in [16]. These dynamics are given by the following differential equation:

$$\frac{d}{dt} p_i(t) = f_i(\mathbf{p}) - \frac{1}{n} \sum_{j=1}^{n} f_j(\mathbf{p}), \text{ for all } i \in \mathcal{S}. \tag{7}$$

Then, according to (7), the proportion of agents p_i grows as the fitness function $f_i(\mathbf{p})$ is greater than the average of fitness functions $\frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{p})$, and decreases otherwise. Alternatively, the projection dynamic in (7) can be re-written as follows:

$$\begin{aligned}
\frac{d}{dt} p_i(t) &= \frac{1}{n} \sum_{j=1}^{n} f_i(\mathbf{p}(t)) - \frac{1}{n} \sum_{j=1}^{n} f_j(\mathbf{p}(t)), \text{ for all } i \in \mathcal{S}, \\
\frac{d}{dt} p_i(t) &= \frac{1}{n} \sum_{j=1}^{n} \left(f_i(\mathbf{p}(t)) - f_j(\mathbf{p}(t))\right), \text{ for all } i \in \mathcal{S}, \\
\frac{d}{dt} \mathbf{p}(t) &= \frac{1}{n} \mathbf{L}\mathbf{f}(\mathbf{p}(t)),
\end{aligned}$$

where $\mathbf{L}$ corresponds to the Laplacian matrix of a complete graph [14]. The equilibrium point of the projection dynamics (7) is achieved when $f_i(\mathbf{p}^*) = \frac{1}{n}\sum_{j=1}^{n} f_j(\mathbf{p}^*)$, for all $i \in \mathcal{S}$. This fact implies that at the equilibrium of (7), $f_i(\mathbf{p}^*) = f_j(\mathbf{p}^*)$, for all $i, j \in \mathcal{S}$, and therefore $\mathbf{p}^* \in \Delta$ is a Nash equilibrium according to Definition 1.

For the population-games-based dynamical tuning for multi-objective MPC controllers, it is proposed to use the discrete version of the projection dynamics, which is obtained by using the Euler approximation for a sampling time $\tau \in \mathbb{R}_{>0}$, i.e.,

$$\frac{d}{dt} p_i(t) \approx \frac{\left(p_i(k+1) - p_i(k)\right)}{\tau}.$$

Then,

$$p_i(k+1) = \tau \left(f_i(\mathbf{p}) - \frac{1}{n} \sum_{j=1}^{n} f_j(\mathbf{p}) \right) + p_i(k),$$

for all $i \in \mathcal{S}$. Notice that the projection dynamics can be re-written in a compacted manner as follows:

$$\mathbf{p}(k+1) = \tau \left(\mathbb{I}_n - \frac{1}{n} \mathbb{1}_n \mathbb{1}_n^\top \right) \mathbf{f}(\mathbf{p}) + \mathbf{p}(k), \tag{8}$$

$$\mathbf{p}(k+1) = \frac{\tau}{n} \mathbf{L}\, \mathbf{f}(\mathbf{p}) + \mathbf{p}(k). \tag{9}$$

The equilibrium of (8) is the same as the equilibrium of (7). Then, the equilibrium of (8) implies that $f_i(\mathbf{p}^*) = f_j(\mathbf{p}^*)$, for all $i, j \in \mathcal{S}$. Prior making the stability analysis of the equilibrium point $\mathbf{p}^* \in \Delta$, it is shown in Proposition 1 that the set of population states Δ is invariant under the discrete projection dynamics (8).

Proposition 1 *The simplex* Δ *is an invariant set under the discrete projection dynamics (8), i.e., being* $\mathbf{p}(0)$ *the initial condition of the population state, if* $\mathbf{p}(0) \in \Delta$*, then* $\mathbf{p}(k) \in \Delta$*, for all* $k \in \mathbb{Z}_{\geq 0}$.

Proof. It is desired to prove that $\mathbb{1}_n^\top \mathbf{p}(k+1) = \mathbb{1}_n^\top \mathbf{p}(k)$. Then

$$\begin{aligned}\mathbb{1}_n^\top \mathbf{p}(k+1) &= \tau \mathbb{1}_n^\top \left(\mathbb{I}_n - \frac{1}{n}\mathbb{1}_n \mathbb{1}_n^\top\right)\mathbf{f}(\mathbf{p}) + \mathbb{1}_n^\top \mathbf{p}(k),\\ &= \tau \mathbb{1}_n^\top \left(\mathbf{f}(\mathbf{p}) - \frac{1}{n}\mathbb{1}_n \mathbb{1}_n^\top \mathbf{f}(\mathbf{p})\right) + \mathbb{1}_n^\top \mathbf{p}(k),\\ &= \tau \left(\mathbb{1}_n^\top \mathbf{f}(\mathbf{p}) - \frac{1}{n}\mathbb{1}_n^\top \mathbb{1}_n \mathbb{1}_n^\top \mathbf{f}(\mathbf{p})\right) + \mathbb{1}_n^\top \mathbf{p}(k).\end{aligned}$$

Since $\frac{1}{n}\mathbb{1}_n^\top \mathbb{1}_n = 1$, it is obtained that

$$\mathbb{1}_n^\top \mathbf{p}(k+1) = \tau\left(\mathbb{1}_n^\top \mathbf{f}(\mathbf{p}) - \mathbb{1}_n^\top \mathbf{f}(\mathbf{p})\right) + \mathbb{1}_n^\top \mathbf{p}(k).$$

Finally, $\mathbb{1}_n^\top \mathbf{p}(k+1) = \mathbb{1}_n^\top \mathbf{p}(k)$, which completes the proof. ∎

The equilibrium point $\mathbf{p}^* \in \Delta$ is asymptotically stable under the discrete projection dynamics (8) by selecting appropriately the sampling time τ as stated in Proposition 2.

Proposition 2 *Let* $\mathbf{f}$ *be a potential and stable game with potential function* $V(\mathbf{p})$*, then the equilibrium point* $\mathbf{p}^* \in \Delta$ *is asymptotically stable under the discrete projection dynamics (8) if the sampling time* τ *is selected such that the matrix* $\Xi(\tau) = \Psi + \frac{\tau}{2}\Psi^\top D\mathbf{f}(\mathbf{p})\Psi$ *is positive definite, where* $\Psi = \left(\mathbb{I}_n - \frac{1}{n}\mathbb{1}_n\mathbb{1}_n^\top\right) = \frac{1}{n}\mathbf{L}$.

Proof. Since $\mathbf{f}(\mathbf{p}) = \nabla V(\mathbf{p})$, and $\mathbf{f}$ is a stable game, then $V(\mathbf{p})$ is a concave function. Consider the following Lyapunov function candidate:

$$E_v(k) = \frac{V(\mathbf{p}^*) - V(\mathbf{p}(k))}{\tau},$$

where $E_v > 0$, for all $\mathbf{p} \neq \mathbf{p}*$, and $E_v = 0$ for $\mathbf{p} = \mathbf{p}^*$. It is necessary to show that $\Delta E_v = E_v(k+1) - E_v(k) \leq 0$, i.e.,

$$\begin{aligned}\Delta E_v &= \frac{V(\mathbf{p}^*) - V(\mathbf{p}(k+1)) - V(\mathbf{p}^*) + V(\mathbf{p}(k))}{\tau},\\ &= \frac{-V(\mathbf{p}(k+1)) + V(\mathbf{p}(k))}{\tau}.\end{aligned}$$

As in [39], the Taylor expression of $V(\mathbf{p})$ at $\mathbf{p}$ yields

$$V(\mathbf{p}(k+1)) = V(\mathbf{p}(k)) + \nabla V(\mathbf{p}(k))^\top \Delta\mathbf{p}(k) + \frac{1}{2}\Delta\mathbf{p}^\top \nabla^2 V(\mathbf{z}(k))\Delta\mathbf{p}(k),$$

where $\Delta\mathbf{p}(k) = \mathbf{p}(k+1) - \mathbf{p}(k)$, and $\mathbf{z}(k)$ is a value between $\mathbf{p}(k)$, and $\mathbf{p}(k+1)$. It follows that

$$\Delta E_v = -\frac{1}{\tau}\nabla V(\mathbf{p}(k))^\top \Delta\mathbf{p}(k) - \frac{1}{2\tau}\Delta\mathbf{p}^\top \nabla^2 V(\mathbf{z}(k))\Delta\mathbf{p}(k). \tag{10}$$

Then, replacing from (8) the term $\Delta\mathbf{p}$ in (10) yields

$$\begin{aligned}\Delta E_v &= -\nabla V(\mathbf{p}(k))^\top \Psi \nabla V(\mathbf{p}) - \frac{\tau}{2}\nabla V(\mathbf{p})^\top \Psi^\top \nabla^2 V(\mathbf{z}(k))\Psi\nabla V(\mathbf{p}),\\ &= -\nabla V(\mathbf{p}(k))^\top \left(\Psi + \frac{\tau}{2}\Psi^\top D\mathbf{f}(k)\Psi\right)\nabla V(\mathbf{p}).\end{aligned}$$

In conclusion, the equilibrium point $\mathbf{p}^* \in \Delta$ is asymptotically stable if $\Xi(\tau) = \Psi + \frac{\tau}{2}\Psi^\top D\mathbf{f}(k)\Psi$ is positive definite. In addition, notice that there exists a $\tau \in \mathbb{R}_{>0}$. To verify this fact, ΔE_v is expressed in terms of the Laplacian $\mathbf{L}$, i.e.,

$$\Delta E_v = \underbrace{-\frac{1}{n}\nabla V(\mathbf{p}(k))^\top \mathbf{L}\nabla V(\mathbf{p})}_{\Delta E_v^1} \underbrace{-\frac{\tau}{2n^2}\nabla V(\mathbf{p}(k))^\top \mathbf{L}^\top D\mathbf{f}(k)\mathbf{L}\nabla V(\mathbf{p})}_{\Delta E_v^2},$$

where the term $\Delta E_v^1 \leq 0$ since it is a quadratic form and $\mathbf{L}$ is positive definite [14], and $\Delta E_v^2 \geq 0$ since it is a quadratic form and $D\mathbf{f}(k)$ is negative semidefinite according to Definition 3. Therefore, there exists a sufficiently small $\tau \in \mathbb{R}_{>0}$ such that $|\Delta E_v^1| \geq |\Delta E_v^2|$. ■

Proposition 2 requires that the game $\mathbf{f}$ was full potential. Nevertheless, the discrete projection dynamics (8) can also be implemented for other types of games. Therefore, Proposition 3 presents the stability proof for a game that does not require that the game is full potential, but still stable. Afterwards, it is shown that both results are equivalent for full-potential games.

Proposition 3 *Let* $\mathbf{f}$ *be a stable game, then there exists a sampling time* $\tau \in \mathbb{R}_{>0}$ *such that the equilibrium point* $\mathbf{p}^* \in \Delta$ *is asymptotically stable under the discrete projection dynamics (8). The sampling time* τ *is selected such that* $|2\left(\mathbf{p}(k) - \mathbf{p}^*\right)^\top \mathbf{f}(\mathbf{p})| > |\tau \mathbf{f}(\mathbf{p})^\top \Psi^\top \Psi \mathbf{f}(\mathbf{p})|$ *is satisfied.*

Proof. Consider the Lyapunov function $E(k) = \frac{1}{\tau}\sum_{i=1}^{n}\left(p_i(k) - p_i^*\right)^2$, where $E(k) > 0$ for all $\mathbf{p} \neq \mathbf{p}^*$, and $E(k) = 0$ for $\mathbf{p} = \mathbf{p}^*$. It is necessary to show that $\Delta E_v = E_v(k+1) - E_v(k) \leq 0$, i.e.,

$$
\begin{aligned}
\Delta E_v &= \frac{1}{\tau}\sum_{i=1}^{n}\left\{p_i^2(k+1) - 2p_i(k+1)p_i^* + p_i^{*2} - p_i^2(k) + 2p_i(k)p_i^* - p_i^{*2}\right\}, \\
&= \frac{1}{\tau}\sum_{i=1}^{n}\left\{-2p_i(k+1)p_i^* + 2p_i(k)p_i^* + p_i^2(k+1) - p_i^2(k)\right\}, \\
&= \frac{1}{\tau}\sum_{i=1}^{n}\left\{-2p_i(k+1)p_i^* + 2p_i(k)p_i^*\right\} + \frac{1}{\tau}\sum_{i=1}^{n}\left\{p_i^2(k+1) - p_i^2(k)\right\}, \\
&= \frac{1}{\tau}\sum_{i=1}^{n} -2p_i^*\left(p_i(k+1) - p_i(k)\right) + \frac{1}{\tau}\sum_{i=1}^{n}\left(p_i(k+1) - p_i(k)\right)^2 + \frac{1}{\tau}\sum_{i=1}^{n} 2p_i(k)\left(p_i(k+1) - p_i(k)\right), \\
&= \frac{1}{\tau}\sum_{i=1}^{n} 2\left(p_i(k) - p_i^*\right)\left(p_i(k+1) - p_i(k)\right) + \frac{1}{\tau}\sum_{i=1}^{n}\left(p_i(k+1) - p_i(k)\right)^2.
\end{aligned}
$$

Replacing the projection dynamics, it follows that

$$
\Delta E_v = \underbrace{2(\mathbf{p}(k) - \mathbf{p}^*)^\top \Psi \mathbf{f}(\mathbf{p})}_{\Delta E_v^1} + \underbrace{\tau \mathbf{f}(\mathbf{p})^\top \Psi^\top \Psi \mathbf{f}(\mathbf{p})}_{\Delta E_v^2}.
$$

The first term ΔE_v^1 is re-written as follows:

$$
\begin{aligned}
\Delta E_v^1 &= 2(\mathbf{p}(k) - \mathbf{p}^*)^\top \left(\mathbb{I}_n - \frac{1}{n}\mathbb{1}_n\mathbb{1}_n^\top\right)\mathbf{f}(\mathbf{p}) \\
&= 2(\mathbf{p}(k) - \mathbf{p}^*)^\top \mathbf{f}(\mathbf{p}) - \frac{2}{n}(\mathbf{p}(k) - \mathbf{p}^*)^\top \mathbb{1}_n\mathbb{1}_n^\top \mathbf{f}(\mathbf{p}) \\
&= 2(\mathbf{p}(k) - \mathbf{p}^*)^\top \mathbf{f}(\mathbf{p}) - \frac{2}{n}\underbrace{\left(\mathbf{p}(k)^\top \mathbb{1}_n - \mathbf{p}^{*\top}\mathbb{1}_n\right)}_{0}\mathbb{1}_n^\top \mathbf{f}(\mathbf{p}) \\
&= 2(\mathbf{p}(k) - \mathbf{p}^*)^\top \mathbf{f}(\mathbf{p}),
\end{aligned}
$$

then it is concluded that $\Delta E_v^1 \leq 0$ since $\mathbf{f}$ is stable. On the other hand, $\Delta E_v^2 = \frac{\tau}{n^2}\mathbf{f}(\mathbf{p})^\top \mathbf{L}^\top \mathbf{L}\mathbf{f}(\mathbf{p})$, and it is concluded that $\Delta E_v^2 \geq 0$. Finally, there exists a sampling time $\tau \in \mathbb{R}_{>0}$ such that $|\Delta E_v^1| \geq |\Delta E_v^2|$. ■

2.2.2 Finding the sampling time: A potential-game example

Consider the coordination game given by the following potential function:

$$
V(\mathbf{p}) = -\frac{p_1^2}{2} - p_2^2 - \frac{3p_3^2}{2},
$$

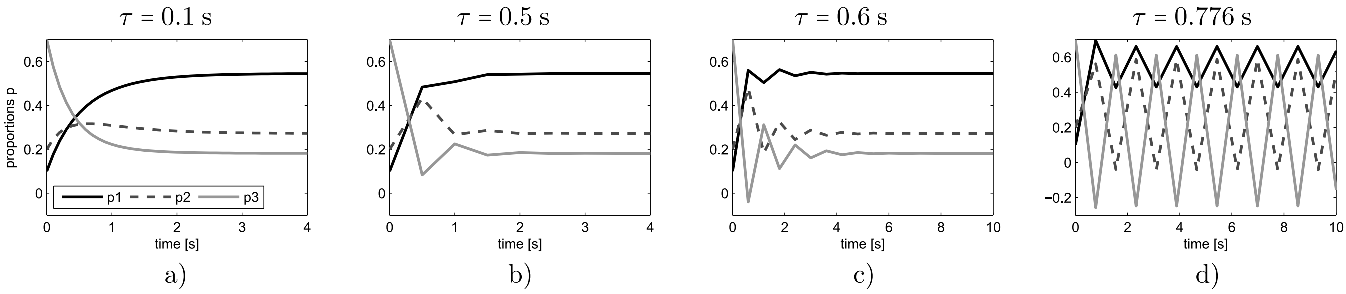

Fig. 1. Evolution of proportion of agents for the coordination game under the discrete projection dynamics for four different values of τ. a) stable with $\tau = 0.1 < 0.776$, b) stable with $\tau = 0.5 < 0.776$, c) stable with $\tau = 0.6 < 0.776$, and d) marginally stable with $\tau = 0.776$.

then, $D\mathbf{f}(\mathbf{p}) = \mathrm{diag}([-1 \quad -2 \quad -3])$. According to Proposition 2, the condition for asymptotic stability of the equilibrium point $\mathbf{p}^* \in \Delta$ is given by

$$\Xi(\tau) = \begin{bmatrix} \frac{2}{3} - \frac{\tau}{2} & \frac{\tau}{6} - \frac{1}{3} & \frac{\tau}{3} - \frac{1}{3} \\ \frac{\tau}{6} - \frac{1}{3} & \frac{2}{3} - \frac{2\tau}{3} & \frac{\tau}{2} - \frac{1}{3} \\ \frac{\tau}{3} - \frac{1}{3} & \frac{\tau}{2} - \frac{1}{3} & \frac{2}{3} - \frac{5\tau}{6} \end{bmatrix}.$$

The conditions over τ to make $\Xi(\tau)$ positive definite are:

$$\frac{2}{3} - \frac{1}{2}\tau > 0, \text{ and } \frac{11}{36}\tau^2 - \frac{2}{3}\tau + \frac{1}{3} > 0.$$

It follows that $\Xi(\tau)$ is positive definite for any $\tau < 0.776$ s, which is the condition to have asymptotic stability of the equilibrium point $\mathbf{p}^* \in \Delta$ under the discrete projection dynamics (8). Figure 1 shows the evolution of the proportion of agents $\mathbf{p} \in \Delta$ for the coordination game under the discrete projection dynamics using different sampling times. It can be seen that the system is marginally stable when $\tau = 0.776$ s, validating the condition over τ to have asymptotic stability. Considering Proposition 3, it is also possible to find the conditions over the sampling time τ by solving the following problem $\min_{\tau \in \mathbb{R}_{>0}, \mathbf{p} \in \Delta} \tau$, subject to $0 \leq 2(\mathbf{p}(k) - \mathbf{p}^*)^\top \mathbf{f}(\mathbf{p}) + \frac{\tau}{n^2} \mathbf{f}(\mathbf{p})^\top \mathbf{L}^\top \mathbf{L} \mathbf{f}(\mathbf{p})$, i.e., the minimum τ such that stability condition is not satisfied with a $\mathbf{p} \in \Delta$. When solving this optimization problem with $\mathbf{f}(\mathbf{p}) = \mathrm{diag}([-1 \quad -2 \quad -3])\mathbf{p}$, it is found that $\tau_c = 0.7762$ is the critical sampling time with $\mathbf{p} = [0.5941 \quad 0.4058 \quad 0]^\top$. This example validates the equivalence between the conditions for τ in Propositions 2 and 3.

3 Proposed dynamical tuning methodology

The proposed dynamical tuning methodology based on population games consists of two different stages. First, it is necessary to normalize the multi-objective cost function, and then the discrete projection dynamics assign permanently the appropriate weights p_i for each one of the control objectives $J_i(\mathbf{x}(k), \mathbf{u})$, for all $i \in \mathcal{S}$. These two main steps of the dynamical tuning methodology are explained next.

3.1 Normalization

The cost function (4a) has several control objectives, which might depend on different parameters, e.g., one objective depending on the system states in contrast with another objective in function of the control inputs. Furthermore, several objectives (even if they involve the same variables) might have different order of magnitude. Therefore, it is necessary to perform a normalization procedure in order to make a fair comparison among all the control objectives.

Let $\mathbf{x}_i^*, \mathbf{u}_i^*$ be the optimal solution of the optimization problem (4) considering only the function $J_i(\mathbf{x}(k), \mathbf{u})$, i.e., the solution of (4) with weights $\gamma_i = 1$, and $\gamma_j = 0$, for all $j \in \mathcal{S} \backslash \{i\}$. Then, the Utopia point denoted by $\mathbf{J}^{\text{utopia}} = [J_1^{\text{utopia}} \quad \dots \quad J_n^{\text{utopia}}]^\top$ is computed as in [11], i.e.,

$$\mathbf{J}^{\text{utopia}} = \left[J_1(\mathbf{x}_1^*, \mathbf{u}_1^*) \quad J_2(\mathbf{x}_2^*, \mathbf{u}_2^*) \quad \cdots \quad J_n(\mathbf{x}_n^*, \mathbf{u}_n^*) \right]^\top. \tag{11}$$

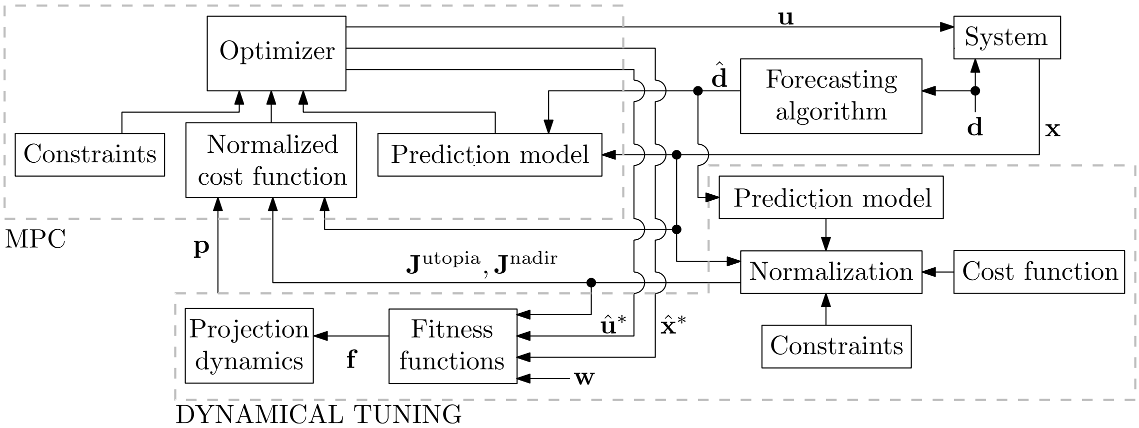

Fig. 2. General scheme of the proposed dynamical tuning.

On the other hand, the i^{th} Nadir value is computed as in [11], i.e.,

$$J_i^{\text{nadir}} = \max\left(J_i(\mathbf{x}_1^*, \mathbf{u}_1^*), J_i(\mathbf{x}_2^*, \mathbf{u}_2^*), \cdots, J_i(\mathbf{x}_n^*, \mathbf{u}_n^*)\right), \tag{12}$$

where the Nadir point $\mathbf{J}^{\text{nadir}}$ is given by

$$\mathbf{J}^{\text{nadir}} = \begin{bmatrix} J_1^{\text{nadir}} & J_2^{\text{nadir}} & \cdots & J_n^{\text{nadir}} \end{bmatrix}^\top. \tag{13}$$

Finally, the normalized multi-objective cost function denoted by $\tilde{J}(\mathbf{x}(k), \mathbf{u})$ has the form

$$\tilde{J}(\mathbf{x}(k), \mathbf{u}) = \sum_{i=1}^{n} \tilde{J}_i(\mathbf{x}(k), \mathbf{u}),$$

where each normalized objective is given by

$$\tilde{J}_i(\mathbf{x}(k), \mathbf{u}) = \frac{J_i(\mathbf{x}(k), \mathbf{u}) - J_i^{\text{utopia}}}{J_i^{\text{nadir}} - J_i^{\text{utopia}}}.$$

Having normalized the cost function $J(\mathbf{x}(k), \mathbf{u})$, then the established weights assign a prioritization without being affected by the order of magnitude of each objective. This procedure is illustrated in Figure 2, receiving information from the cost function, prediction model, and constraints.

3.2 Dynamical weighting procedure

Once the cost function has been normalized, it is considered that the prioritization weights at each control objective $J_i(\mathbf{x}(k), \mathbf{u})$ are given by a time-varying parameter $p_i(k)$, for all $i \in \mathcal{S}$. Hence, the normalized optimization problem behind the MPC controller is formulated as follows:

$$\underset{\hat{\mathbf{u}}}{\text{minimize}} \sum_{i=1}^{n} p_i(k) \tilde{J}_i(\mathbf{x}(0), \mathbf{u}), \tag{14a}$$

subject to:

$$\mathbf{x}(k+j+1|k) = \mathbf{A}\mathbf{x}(k+j|k) + \mathbf{B}\mathbf{u}(k+j|k) + \mathbf{B}_l\mathbf{d}(k+j|k), \quad j \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}, \tag{14b}$$

$$\mathbf{u}(k+j|k) \in \mathcal{U}, \quad j \in [0, H_p - 1] \cap \mathbb{Z}_{\geq 0}, \tag{14c}$$

$$\mathbf{x}(k+j|k) \in \mathcal{X}, \quad j \in [1, H_p] \cap \mathbb{Z}_{\geq 0}, \tag{14d}$$

where $\mathbf{p}(k) = \begin{bmatrix} p_1(k) & \cdots & p_n(k) \end{bmatrix}^\top$, satisfying the constraint $\sum_{i=1}^{n} p_i(k) = 1$. The unitary value in the constraint of weights is associated to the population mass that defines the simplex set Δ in the population game. Notice that weights should vary dynamically since the disturbances in the system (1) also vary along the time. To overcome this issue, the discrete projection dynamics (8) are implemented. The fitness functions $f_i(p_i(k))$, for all $i \in \mathcal{S}$, are chosen to be dependent of the current value of each control objective $\tilde{J}_i(\hat{\mathbf{x}}^*(k), \hat{\mathbf{u}}^*(k))$ representing each strategy, i.e.,

$$f_i(p_i(k)) = w_i \tilde{J}_i(\hat{\mathbf{x}}^*(k), \hat{\mathbf{u}}^*(k)), \tag{15}$$

where w_i, for all $i \in \mathcal{S}$, assigns a prioritization that defines a *management region* in the Pareto front as has been presented in [3]. Besides, these terms w_i, for all $i \in \mathcal{S}$, do not appear in the optimization problem of the MPC, and should not be confused with the weights of the cost function (14a) in the MPC controller, which are denoted by p_i, for all $i \in \mathcal{S}$.

Assumption 1 *The fitness function $f_i(p_i)$ is a decreasing function with respect to p_i. It is expected that the value of the objective $\tilde{J}_i(\hat{\mathbf{x}}^*(k), \hat{\mathbf{u}}^*(k))$ decreases as bigger weight $p_i(k)$ is assigned to it when solving the corresponding optimization problem.* ◇

Remark 1 *Propositions 2 and 3 have shown that there exists a sampling time $\tau \in \mathbb{R}_{>0}$ such that the equilibrium point $\mathbf{p}^* \in \Delta$ is asymptotically stable under the discrete projection dynamics. Moreover, in order to find the critical τ_c, it is necessary either to compute the Jacobian $D\mathbf{f}(\mathbf{p})$ or to know the equilibrium point $\mathbf{p}^* \in \Delta$. For the dynamical tuning application, none of these data is available since there is not a function describing the Pareto front depending on the assigned prioritization in the cost function (14(a)), and the equilibrium point varies along the time because of the time-varying disturbance affecting the system. However, there exists a sufficiently small τ to guarantee stability according to Proposition 3 since the game $\mathbf{f}$ is stable. For the tuning application, we have selected $\tau = 0.15 < \tau_c$.* ◇

The dynamical adjustment of the weights is presented in Figure 2. The fitness functions are determined by using information from the normalized cost function and the weights that determine the *management region*. Thus, the discrete projection dynamics compute the appropriate prioritization of the normalized cost function in the MPC controller. A detailed procedure to implement the population-games-based dynamical tuning for multi-objective MPC is presented in Algorithm 1.

Algorithm 1 Dynamical tuning based on population games for multi-objective MPC.

$H_s \leftarrow$ simulation length
$H_p \leftarrow$ prediction horizon
$n \leftarrow$ number of objectives
$\mathbf{x}(k) \leftarrow \mathbf{x}(0) \in \mathbb{R}^{n_x}$ states initial condition
$\mathbf{p}(k) \leftarrow \mathbf{p} \in \mathbb{R}^n_{\geq 0}$ proportion initial condition
for $k = 1 : H_s$ **do**
  for $i = 1 : n$ **do**
    $\mathbf{u}_i^* \leftarrow \arg\min_{\hat{\mathbf{u}}} J_i(\mathbf{x}, \mathbf{u})$ with constraints
    $J_i^{\text{utopia}} \leftarrow J_i(\mathbf{x}_i^*, \mathbf{u}_i^*)$
  end for
  for $j = 1 : n$ **do**
    $J_j^{\text{nadir}} \leftarrow \max\left(J_j(\mathbf{x}_1^*, \mathbf{u}_1^*), \cdots, J_j(\mathbf{x}_n^*, \mathbf{u}_n^*)\right)$
  end for
  $\hat{\mathbf{u}}^*(k) \leftarrow \arg\min_{\hat{\mathbf{u}}} \sum_{i=1}^{n} p_i(k) \tilde{J}_i(\mathbf{x}, \mathbf{u})$ with constraints
  $\hat{\mathbf{x}}^*(k) \leftarrow$ using $\hat{\mathbf{u}}^*(k)$ and (14b)
  $\mathbf{u}^*(k) \leftarrow \mathbf{u}^*(k|k) \in \mathbb{R}^{n_u}$ optimal control input
  for $i = 1 : n$ **do**
    $f_i(p_i) \triangleq f_i(p_i(k)) \leftarrow w_i \tilde{J}_i(\hat{\mathbf{x}}^*(k), \hat{\mathbf{u}}^*(k))$
  end for
  $\mathbf{p}(k+1) = \tau\left(\mathbb{I}_n - \frac{1}{n}\mathbb{1}_n\mathbb{1}_n^\top\right)\mathbf{f}(\mathbf{p}) + \mathbf{p}(k).$
  $\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}^*(k) + \mathbf{B}_l\mathbf{d}(k)$
end for

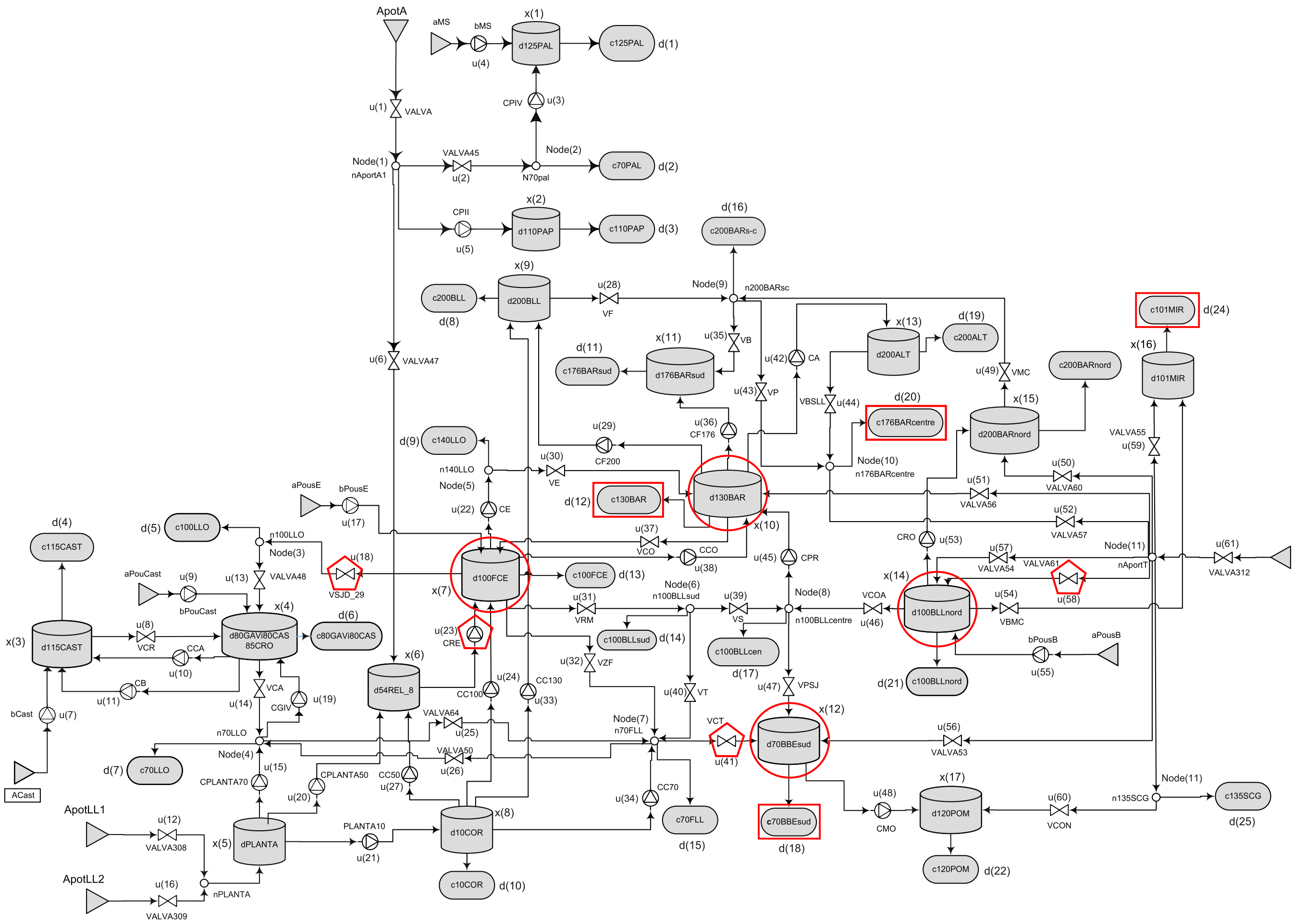

Fig. 3. Case study. Topology of the 17 tanks BWSN. Circles correspond to states 7, 10, 12, and 14. Hexagons correspond to control inputs 18, 23, 41, and 58. Rectangles correspond to demands 12, 18, 20, and 24.

4 Water supply network application

In order to illustrate the performance of a multi-objective MPC controller with a dynamical tuning based on population games, the proposed on-line tuning methodology is implemented in a large-scale water supply network. Furthermore, the performance of the MPC controller with dynamical tuning is compared to the performance obtained by using a conventional static tuning. Figure 3 shows a representative portion of the Barcelona water supply network (BWSN) that is composed of 17 tanks, 26 pumps, 35 valves, nine water sources, 25 water demands, and 11 mass-balance nodes. The dynamical model of the system is given by the following expressions:

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{B}_d\mathbf{d}(k), \tag{16a}$$

$$\mathbf{0} = \mathbf{E}_u\mathbf{u}(k) + \mathbf{E}_d\mathbf{d}(k), \tag{16b}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ is the vector of $n_x = 17$ system states corresponding to the tank volumes, $\mathbf{u} \in \mathbb{R}^{n_u}$ is the vector of $n_u = 61$ control inputs, and $\mathbf{d} \in \mathbb{R}^{n_d}$ is the vector of $n_d = 25$ time-varying water demands. The water demands are considered to be disturbances to the system, which have a periodicity of 24 hours with a mean value, and a nominal amplitude [33]. The constraints given by the 11 mass-balance nodes are described by (16b). Matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{B}_d$, $\mathbf{E}_u$, and $\mathbf{E}_d$ are obtained according to the control-oriented modeling described in [18]. See [19] for further details regarding this case study.

4.1 Management criteria

The MPC controller is designed considering a cost function with multiple objectives. These objectives for the BWSN are established by a *management criteria* considering the following three aspects:

- Economic operation, i.e., $J_1(\mathbf{u}(k)) \triangleq \left|(\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2(k))^\top \mathbf{u}(k)\right|$, where $\boldsymbol{\alpha}_1$ represents the time-invariant costs associated to the water resource, and $\boldsymbol{\alpha}_2$ represents the time-variant costs associated to the operation of valves and pumps.
- Smoothness operation, i.e, $J_2(\mathbf{u}(k)) \triangleq \|\Delta\mathbf{u}(k)\|^2$, where $\Delta\mathbf{u}(k) \; = \; \mathbf{u}(k) - \mathbf{u}(k-1)$.
- Safety operation, i.e., considering the constraint $\mathbf{x}(k) \geq \mathbf{x}_s - \boldsymbol{\xi}(k)$, for all k, with $\mathbf{x}_s \in \mathbb{R}^{n_x}$ being the vector of safety volumes for all the tanks. The third objective is given by $J_3(\boldsymbol{\xi}(k)) \triangleq \|\boldsymbol{\xi}(k)\|^2$.

It is important to clarify that the prioritization of objectives, which is determined by the company in charge of the management of the network, is already known. In fact, the prioritization of these aforementioned objectives is commonly used in the design of controllers using a static tuning [9], [19]. In this particular case study, and according to the company in charge of the system, the most important objective is the minimization of the economical costs, i.e., $J_1(\mathbf{u}(k))$. Followed by the objective related to the safety volumes, i.e., $J_3(\boldsymbol{\xi}(k))$. Finally, the *less* important control objective is related to the smooth operation, i.e., $J_2(\mathbf{u}(k))$. This prioritization order should be satisfied in case of both static and dynamical tuning.

4.2 Optimization problem of the predictive controller

The cost function of the optimization problem behind the MPC controller is determined considering the system management criteria. Therefore, the cost function is composed of three control objectives, i.e., J_1, J_2, and J_3. The cost function of the optimization problem (4) has $n = 3$ control objectives. Hence, following the procedure presented in Section 3.1, a normalized optimization problem of the form as in (14) is obtained, i.e.,

$$\underset{\hat{\mathbf{u}},\hat{\boldsymbol{\xi}}}{\text{minimize}}\, J(\mathbf{u},\boldsymbol{\xi}) = \sum_{i=0}^{H_p-1} p_1(k)\tilde{J}_1(\mathbf{u}(k+i)) + \sum_{i=0}^{H_p-1} p_2(k)\tilde{J}_2(\mathbf{u}(k+i)) + \sum_{i=0}^{H_p-1} p_3(k)\tilde{J}_3(\boldsymbol{\xi}(k+i)),$$

subject to:

$$\begin{aligned}
\mathbf{x}(k+j+1|k) &= \mathbf{A}\mathbf{x}(k+j|k) + \mathbf{B}\mathbf{u}(k+j|k) + \mathbf{B}_l\mathbf{d}(k+j|k), j \in [0, H_p-1] \cap \mathbb{Z}_{\geq 0},\\
\mathbf{0} &= \mathbf{E}_u\mathbf{u}(k+j|k) + \mathbf{E}_d\mathbf{d}(k+j|k), j \in [0, H_p-1] \cap \mathbb{Z}_{\geq 0},\\
\mathbf{u}(k+j|k) &\in \mathcal{U}, \;\; j \in [0, H_p-1] \cap \mathbb{Z}_{\geq 0},\\
\mathbf{x}(k+j|k) &\in \mathcal{X}, \;\; j \in [0, H_p] \cap \mathbb{Z}_{\geq 0},\\
\mathbf{x}(k+j|k) &\geq \mathbf{x}_s - \boldsymbol{\xi}(k+i|k), \;\; j \in [0, H_p] \cap \mathbb{Z}_{\geq 0},\\
\boldsymbol{\xi}(k+j|k) &\geq 0, \;\; j \in [0, H_p] \cap \mathbb{Z}_{\geq 0},
\end{aligned}$$

where the feasible sets for the control inputs $\mathcal{U}$ and the system states $\mathcal{X}$ are given by $\mathcal{U} = \{\mathbf{u} \in \mathbb{R}^{n_u} | \mathbf{u}_{min} \leq \mathbf{u} \leq \mathbf{u}_{max}\}$, and $\mathcal{X} = \{\mathbf{x} \in \mathbb{R}^{n_x} | \mathbf{x}_{min} \leq \mathbf{x} \leq \mathbf{x}_{max}\}$, respectively, being $\mathbf{u}_{min}$, and $\mathbf{u}_{max}$ the minimum and maximum limits for the control inputs, and $\mathbf{x}_{min}$, and $\mathbf{x}_{max}$ the minimum and maximum limits for the system states. Finally, similarly as in (3), $\hat{\boldsymbol{\xi}}$ is a sequence along H_p.

Figure 4 shows the trend of the normalized functions $\tilde{J}_i(\hat{\mathbf{x}}^*(k), \hat{\mathbf{u}}^*(k))$, for all $i = 1, 2, 3$. It can be seen that these functions are decreasing with respect to the weight p_i. This is because it is expected to get a smaller value from the minimization problem as more prioritization is assigned (see Assumption 1).

4.3 Scenarios

In order to illustrate the enhancement of the control performance when adopting the population-games-based dynamical tuning methodology, the performance obtained with the dynamical tuning is compared to the performance

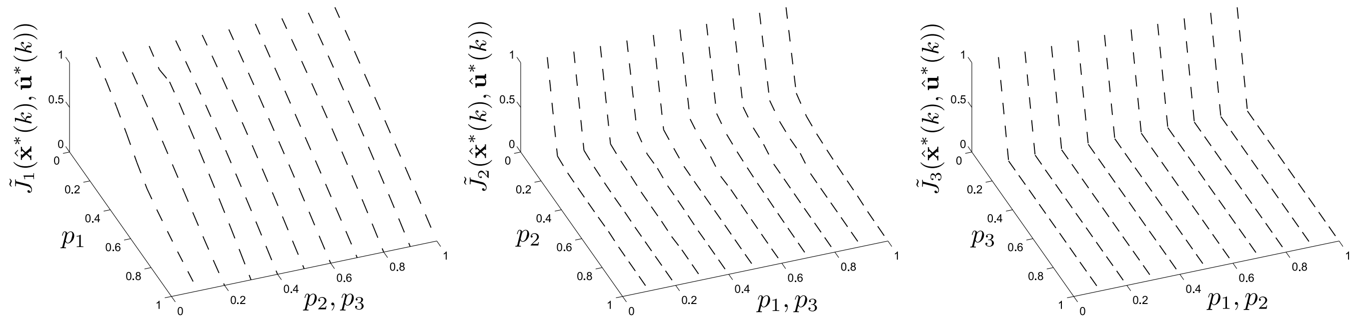

Fig. 4. Behavior of the trend of the normalized functions $\tilde{J}_i(\hat{\mathbf{x}}^*(k), \hat{\mathbf{u}}^*(k))$, for all $i = 1, 2, 3$.

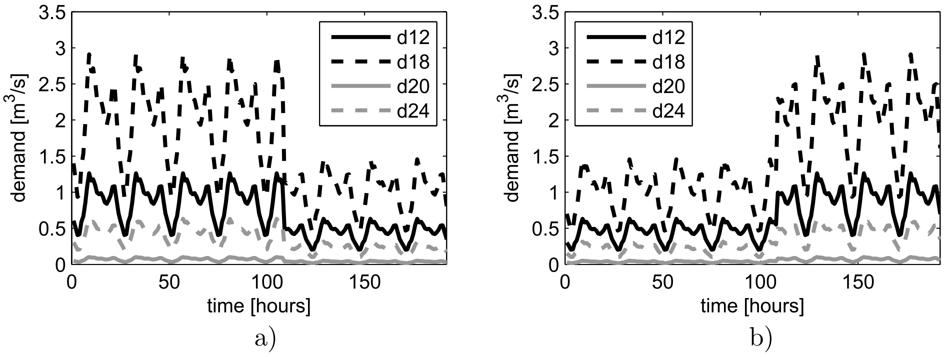

Fig. 5. Demand profile for: a) Scenario 1 and b) Scenario 2. The demands within the network can be seen in Figure 3 signed with squares, and correspond to disturbances in the model (1).

when static weights are established to the objectives in the cost function. Besides, two different scenarios are proposed. In general, the water demand profiles have a periodic behavior (daily), remaining a constant mean value, and maintaining a regular amplitude. Nevertheless, it is considered the event in which the periodic demand changes unexpectedly along the time, i.e., when the demand varies its mean value and its regular amplitude. The purpose is to assess the automatic adjustment of the weights when conditions over the system suffer a modification along the time, improving the performance with respect to an MPC with static tuning.

The performance when the demand suffers a decrement, and when demand has a sudden increment are analyzed. These two possible scenarios are presented in Figure 5, i.e.,

- **Scenario 1:** decrement of the mean value of the demand profiles (see Figure 5a)).
- **Scenario 2:** increment of the mean value of the demand profiles (see Figure 5b)).

The decrement and increment of the mean value of the disturbances is made arbitrarily at the end of the fourth day.

In order to make a fair comparison, the weights $\gamma_1, ..., \gamma_n$ for the cost function in problem (4) *for the static tuning case* and the weights for the *management region* $w_1, ..., w_n$ in (15) *for the dynamical tuning case* are selected to be the same, i.e., $w_i = \gamma_i$, for all $i \in \mathcal{S}$.

4.4 Results and discussion

The performance of the controllers is evaluated by using an economical *key performance index* denoted by C during the total number of simulation days (in this case eight days), i.e.,

$$C = \sum_{k=0}^{192} \left(\boldsymbol{\alpha}_1 + \boldsymbol{\alpha}_2(k)\right)^\top \mathbf{u}(k), \tag{18}$$

where $k \in \mathbb{Z}_{\geq 0}$ in given in hours. Furthermore, a sub-index is used to differentiate between the results with static tuning, and with the proposed dynamical tuning, i.e., C_S and C_D, respectively. For each scenario, six different cases corresponding to six *management regions* are tested:

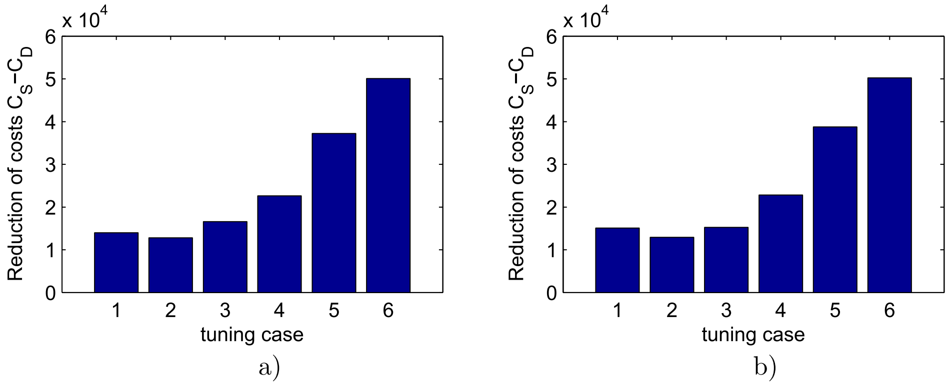

Fig. 6. Reduction of costs in 8 days for the six different tuning cases. a) Scenario 1, and b) Scenario 2.

- Tuning case 1: $[\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top = [0.8 \quad 0.05 \quad 0.15]^\top$,
- Tuning case 2: $[\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top = [0.7 \quad 0.1 \quad 0.2]^\top$,
- Tuning case 3: $[\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top = [0.6 \quad 0.15 \quad 0.25]^\top$,
- Tuning case 4: $[\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top = [0.5 \quad 0.2 \quad 0.3]^\top$,
- Tuning case 5: $[\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top = [0.4 \quad 0.25 \quad 0.35]^\top$,
- Tuning case 6: $[\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top = [0.35 \quad 0.3 \quad 0.35]^\top$,

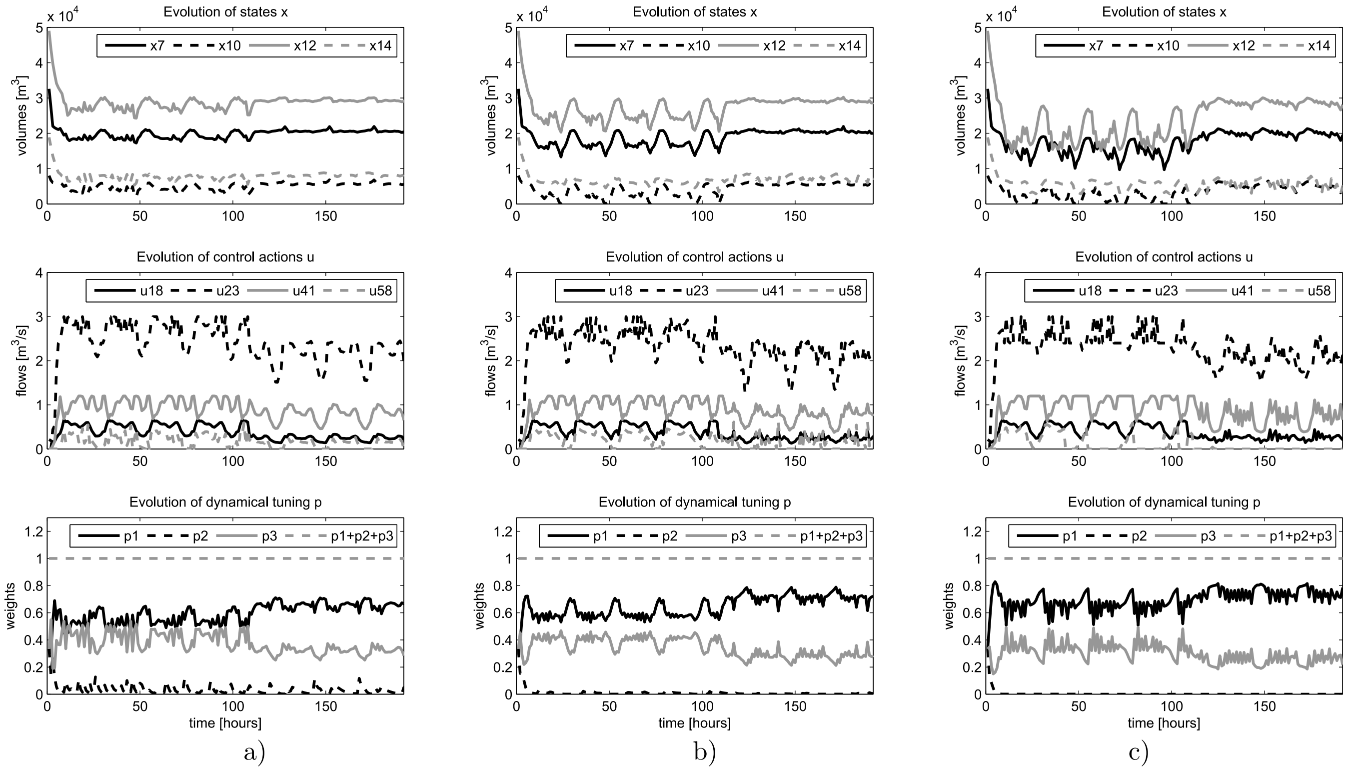

Fig. 7. Evolution of volumes, control inputs, and tuning weights for the population-games-based dynamical tuning in Scenario 1, for three different management points: a) First column $\mathbf{w} = [0.4 \quad 0.25 \quad 0.35]^\top$, b) second column $\mathbf{w} = [0.6 \quad 0.15 \quad 0.25]^\top$, and c) third column $\mathbf{w} = [0.8 \quad 0.05 \quad 0.15]^\top$. The corresponding states and control inputs within the network can be seen in Figure 3.

where $[w_1 \quad w_2 \quad w_3]^\top = [\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top$. Notice that all the proposed tuning cases satisfy the prioritization order presented in Section 4.1, i.e., $w_1 > w_3 > w_2$. Table 1 presents the comparison between the economic results obtained with a multi-objective MPC using a static and dynamical population-games-based tuning, and for the two different scenarios. Also, Table 1 shows the reduction of costs when adopting the proposed dynamical tuning, i.e., $C_S - C_D$. It can be seen that, for all the tested *management regions*, and for both scenarios, a reduction of costs is obtained

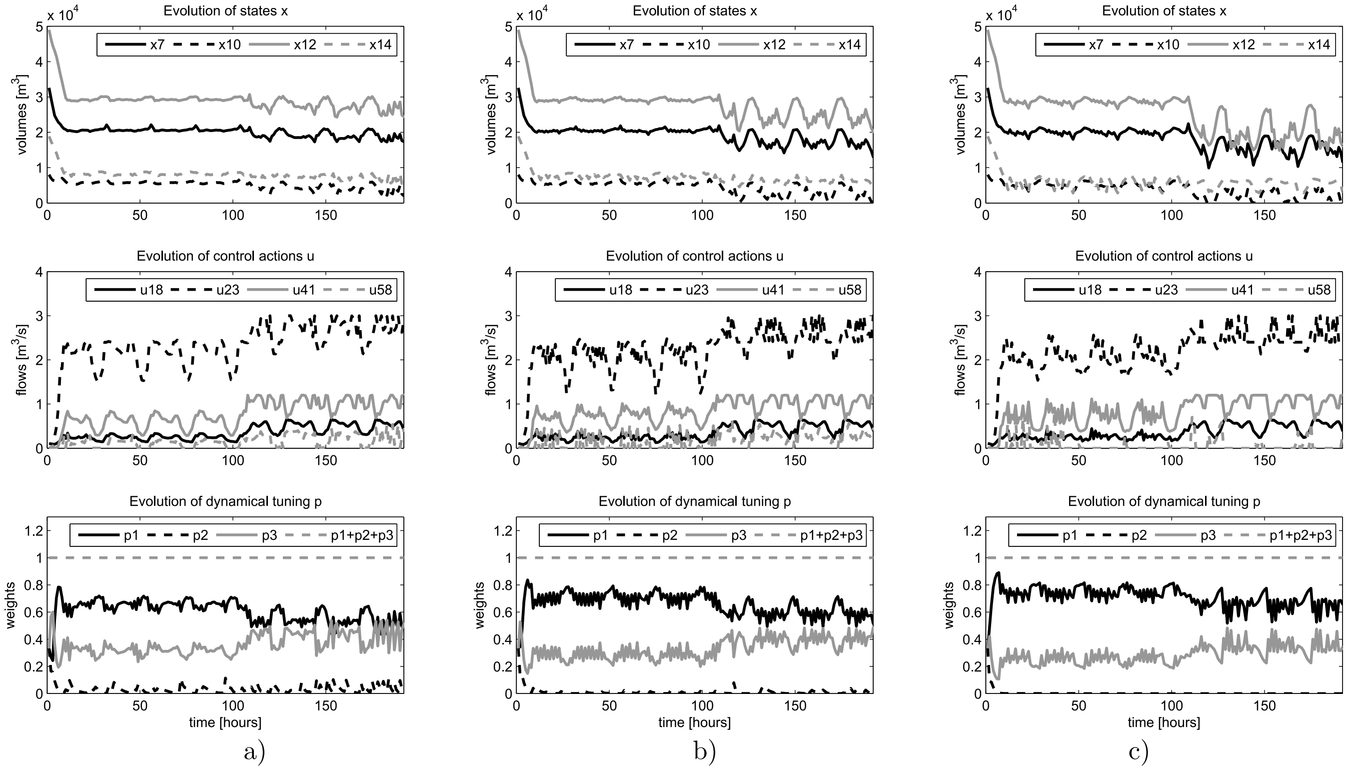

Fig. 8. Evolution of volumes, control inputs, and tuning weights for the population-games-based dynamical tuning in Scenario 2, for three different management points: a) First column $\mathbf{w} = [0.4 \quad 0.25 \quad 0.35]^\top$, b) second column $\mathbf{w} = [0.6 \quad 0.15 \quad 0.25]^\top$, and c) third column $\mathbf{w} = [0.8 \quad 0.05 \quad 0.15]^\top$. The corresponding states and control inputs within the network can be seen in Figure 3.

Table 1
Economic results for Scenario 1 and Scenario 2 in the case study. Notice that for the comparison of data the management region corresponds to the prioritization of the MPC controller with static tuning, i.e., $[w_1 \quad w_2 \quad w_3]^\top = [\gamma_1 \quad \gamma_2 \quad \gamma_3]^\top$.

	Tuning case	**Dynamical tuning costs C_D (e.u.)**	**Static tuning costs C_S (e.u.)**	**Reduction of costs $C_S - C_D$ (e.u.)**	**Percentage reduction $100(C_S - C_D)/C_S$ [%]**
Scenario 1	1	281475.4393	295465.0021	13989.5627	4.73
	2	282296.0113	295114.7771	12818.7657	4.34
	3	283592.6568	300172.7427	16580.0858	5.52
	4	289484.6672	312124.2979	22639.6307	7.25
	5	291048.2900	328267.0268	37218.7368	11.33
	6	291874.0282	341964.3402	50090.3120	14.64
Scenario 2	1	251003.7369	266079.8919	15076.1550	5.66
	2	252147.3533	265056.5038	12909.1505	4.87
	3	255457.0784	270722.0341	15264.9556	5.63
	4	259626.8908	282454.0561	22827.1652	8.08
	5	261713.5740	300459.4927	38745.9187	12.89
	6	263114.2332	313364.1354	50249.9022	16.03

when implementing a dynamical tuning with respect to the costs with a standard static tuning. Figure 6 presents a summary of the reduction of costs for both scenarios and all the tested combination of weights for the control objectives. Cost reductions from 13989.56 to 50090.31 e.u., and from 15076.15 to 50249.90 e.u., are obtained for the first and second scenario in eight days, respectively.

Figures 7 and 8 show the evolution of system states, control inputs, and dynamic prioritization weights for the first and second scenario with *management regions* given by $\mathbf{w} = [0.4 \quad 0.25 \quad 0.35]^\top$, $\mathbf{w} = [0.6 \quad 0.15 \quad 0.25]^\top$, and $\mathbf{w} = [0.8 \quad 0.05 \quad 0.15]^\top$. The performance exhibits an oscillatory behavior for the adjustment of weights because of the disturbances in the system. In fact, it can be seen that the periodicity of the oscillation in the weights adjustment corresponds to the diary periodicity of the demands (see Figure 5). In addition, it can be seen in Figures 7 and 8 that the dynamical tuning suffers an abrupt change at the end of the fourth day adjusting weights appropriately. This fact occurs since, at that point, the decrement or increment of the mean value for the demand profiles is applied.

5 Concluding remarks

A novel dynamical tuning methodology for multi-objective MPC controllers has been presented. The dynamical tuning methodology requires to normalize the cost function of the optimization problem behind the MPC controller. Therefore, a population game is solved with a discrete version of the projection dynamics, which update the appropriate tuning by using information about the current value of the normalized control objectives. The proposed dynamical tuning does not require to generate multiple points of the Pareto front, which implies that it is not computationally costly with respect to other reported on-line approaches. The proposed tuning has been established to be a weighting sum, for which it is required that the sum of all the weights is equal one. It has been shown that the discrete version of the projection dynamics satisfies this constraint throughout the evolution of their variables. Furthermore, the stability analysis of the Nash equilibrium under the discrete projection dynamics has been made, and it is guaranteed as long as the control objectives decrease as more priority is assigned to them (Assumption 1).

Finally, the dynamical tuning methodology is implemented to a large-scale water supply network. Results have shown a reduction of costs when adopting the proposed population-games-based dynamical tuning. The reduction of costs is achieved for all the six tested tuning cases, and for two different scenarios for demand abrupt changes (one scenario considering a decrement of demand, and another considering an increment of demand). It is worth to point out that these achieved cost reductions have been presented for a period of eight days, and that these reductions are maintained along the time. Therefore, the proposed dynamical tuning strategy, according to the results obtained during a week, might represent a bigger reduction of costs in a larger period of time, e.g., a month or a year.

Acknowledgement

Authors would like to thank COLCIENCIAS (grant 6172) and Agència de Gestió d'Ajust Universitaris i de Recerca AGAUR (FI-2014) for supporting J. Barreiro-Gomez. This work has been partially supported by the projects "Drenaje urbano y cambio climático: Hacia los sistemas de alcantarillados del futuro, fase II. COLCIENCIAS", and ECOCIS (Ref. DPI2013-48243-C2-1-R).

References

[1] A. Al-Ghazzawi, E. Ali, A. Nouh, and E. Zafiriou. On-line tuning strategy for model predictive controllers. *Journal of Process Control*, 11:265–284, 2001.

[2] 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.

[3] J. Barreiro-Gomez, C. Ocampo-Martinez, and N. Quijano. Evolutionary-game-based dynamical tuning for multi-objective model predictive control. In S. Olaru, A. Grancharova, and F. Lobo Pereira, editors, *Developments in Model-Based Optimization and Control*, pages 115–138. Springer Verlag, 2015.

[4] J. Barreiro-Gomez, N. Quijano, and C. Ocampo-Martinez. Constrained distributed optimization: A population dynamics approach. *Automatica*, 69:101–116, 2016.

[5] S. Di Cairano and A. Bemporad. Model predictive control tuning by controller matching. *IEEE Transactions on Automatic Control*, 55:185–190, 2010.

[6] J.L. Garriga and M. Soroush. Model predictive control tuning methods: A review. *Industrial & Engineering Chemistry Research (I&EC)*, 49:3505–3515, 2010.

[7] O. Grodzevich and O. Romanko. Normalization and other topics in multi-objective optimization. In *Proceedings of the Fields-MITACS Industrial Problems Workshop*, pages 89–101, Toronto, Ontario, 2006.

[8] J.M. Grosso, C. Ocampo-Martinez, and V. Puig. Learning-based tuning of supervisory model predictive control for drinking water networks. *Engineering Applications of Artificial Intelligence*, 26:1741–1750, 2013.

[9] J.M. Grosso, C. Ocampo-Martinez, V. Puig, and B. Joseph. Chance-constrained model predictive control for drinking water networks. *Journal of Process Control*, 24:504–516, 2014.

[10] N. He, D. Shi, J. Wang, M. Forbes, J. Backstrom, and T. Chen. User friendly robust mpc tuning of uncertain paper-making processes. *Proceedings of the 9th IFAC Symposium on Advanced Control of Chemical Processes (ADCHEM)*, 48:1021–1026, 2015.

[11] I.Y. Kim and O. L. de Weck. Adaptive weighted sum method for multiobjective optimization: a new method for pareto front generation. *Structural and Multidisciplinary Optimization*, 31:105–116, 2006.

[12] S. Lasaulce and H. Tembine. *Game theory and learning for wireless networks: fundamentals and applications.* Academic Press, 2011.

[13] J. Marden and J. Shamma. Game theory and distributed control. *Handbook of Game Theory*, 4:861–899, 2014.

[14] Mehran Mesbahi and Magnus Egerstedt. *Graph theoretic methods in multiagent networks.* Princeton series in applied mathematics. Princeton University Press, Princeton (N.J.), 2010.

[15] M.A. Müller, D. Angeli, and F. Allgöwer. On the performance of economic model predictive control with self-tuning terminal cost. *Journal of Process Control*, 24:1179–1186, 2014.

[16] A. Nagurney and D. Zhang. Projected dynamical systems in the formulation, stability analysis, and computation of fixed demand traffic network equilibria. *Transportation Science*, 31:147–158, 1997.

[17] 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.

[18] C. Ocampo-Martinez, D. Barcelli, V. Puig, and Bemporad A. Hierarchical and decentralised model predictive control of drinking water networks: application to Barcelona case study. *IET Control Theory and Applications*, 6(1):62–71, 2012.

[19] C. Ocampo-Martinez, V. Puig, G. Cembrano, and J. Quevedo. Application of predictive control strategies to the management of complex networks in the urban water cycle. *IEEE Control Systems Magazine*, 33(1):15–41, 2013.

[20] L. Ozkan, J. Meijs, and A.C.P.M. Backx. A frequency domain approach for MPC tuning. In *Proceedings of the Symposium on Process Systems Engineering*, pages 15–19, Singapore, 2012.

[21] S. Parsons and M. Wooldridge. Game theory and decision theory in multi-agent systems. *Autonomous Agents and Multi-Agent Systems*, 5(3):243–254, 2002.

[22] L. Pavel. *Game theory for control of optical networks.* Springer Science & Business Media, 2012.

[23] 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.

[24] E. Ramirez-Llanos and N. Quijano. A population dynamics approach for the water distribution problem. *International Journal of Control*, 83:1947–1964, 2010.

[25] W. H. Sandholm. *Population games and evolutionary dynamics.* Cambridge, Mass. MIT Press, 2010.

[26] J.D. Schwartz and D.E. Rivera. Simulation-based optimal tuning of model predictive control policies for supply chain management using simultenuous perturbation stochastic approximation. In *Proceedings of the American Control Conference (ACC)*, pages 14–16, Minneapolis, Minnesota, USA, 2006.

[27] G. Shah and S. Engell. Tuning MPC for desired closed-loop performance for MIMO systems. In *Proceedings of the American Control Conference (ACC)*, pages 4404–4409, San Francisco, USA, 2011.

[28] H. Tembine, E. Altman, R. El-Azouzi, and Y. Hayel. Evolutionary games in wireless networks. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 40(3):634–646, 2010.

[29] R. Toro, C. Ocampo-Martinez, F. Logist, J.V. Impe, and V. Puig. Tuning of predictive controllers for drinking water networked systems. In *Proceedings of the 18th IFAC World Congress*, pages 14507–14512, Milan, Italy, 2011.

[30] Q.N. Tran, R. Octaviano, L. Özkan, and A.C.P.M. Backx. Generalized predictive control tuning by controller matching. In *Proceedings of the American Control Conference (ACC)*, pages 4889–4894, Portland, USA, 2014.

[31] M. Vallerio, J.V. Impe, and F. Logist. Tuning of NMPC controllers via multi-objective optimisation. *Computers and Chemical Engineering*, 61:38–50, 2014.

[32] J.H. van der Lee, W.Y. Svrcek, and B.R. Young. A tuning algorithm for model predictive controllers based on genetic algorithms and fuzzy decision making. *ISA Transactions*, 47:53–59, 2008.

[33] Y. Wang, C. Ocampo-Martinez, and V. Puig. Stochastic model predictive control based on Gaussian processes applied to drinking water networks. *IET Control Theory and Applications*, 10:947–955, 2016.

[34] Y. Wang, C. Ocampo-Martinez, V. Puig, and J. Quevedo. Gaussian-process-based demand forecasting for predictive control of drinking water networks. In C. G. Panayiotou, G. Ellinas, E. Kyriakides, and M. M. Polycarpou, editors, *Critical Information Infrastructures Security. Lecture Notes in Computer Science 8985: CRITIS 2014*, pages 69–80. Springer International Publishing, 2016.

[35] H. Waschl and L. Alberer, D.and del Re. *Automatic Tuning Methods for MPC Environments*, pages 41–48. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012.

[36] H. Waschl, J.B. Jogensen, J.K. Huusom, and L. del Re. A tuning approach for offset-free MPC with conditional reference adaptation. In *Proceedings of the 19th World Congress*, pages 24–29, Cape Town, South Africa, 2014.

[37] J. W. Weibull. *Evolutionary Game Theory.* The MIT Press, London, England, 1997.

[38] W. Wojsznis, J. Gudaz, T. Blevins, and A. Mehta. Practical approach to tuning MPC. *ISA Transactions*, 42:149–162, 2003.

[39] L. Xiao and S. Boyd. Optimal scaling of a gradient method for distributed resource allocation. *Journal of Optimization Theory and Applications*, 129:469–488, 2006.

[40] A.S. Yamashita, A.C. Zanin, and D. Odloak. Tuning the model predictive control of a crude distillation unit. *ISA Transactions*, 60:178–190, 2016.