Distributed Adaptive Control of a Water Delivery Canal

Tiago José Ribeiro Ricardo

Abstract:
Water Delivery Canals are important structures that contribute to efficient water management. They are characterized for having several sensors and actuators spread across large areas and by a time-variant dynamic behaviour influenced by external factors and disturbances. This provides motivation to distributed adaptive control strategies. The algorithms considered in this study result in the combination of two main steps: a recursive identification algorithm that estimates the system parameters using input and output data retrieved from the plant and a distributed control algorithm that relies on coordination processes based on Game Theory concepts or optimization with augmented Lagrangian. The identification algorithm considered is the Recursive Least Squares (RLS) whereas the control algorithms are based in Linear-Quadratic Gaussian (LQG) and Model Predictive Control (MPC) theory.

Keywords: RLS, LQG, MPC, Distributed Control, Adaptive Control, Water Canal.

1. INTRODUCTION

With the growth in the world population, immediate action is required to manage water in a sustainable way (FAO [2012]). The increasing demand of water and the issue of water scarcity have become critical in several regions worldwide, representing a severe impact in the economy. Therefore it is important to address these issues with the efficient resource management. Water canals play an important role by allowing water management and are characterized for their large dimensions and for their highly dynamic behaviour due to variations in the water levels and disturbances that can be either related with external factors (physical problems, mud accumulation and vegetation growing) or with water consumption (Rijo et al. [2009]). To address these issues, and to allow the system to respond in a both stable and robust way it is important to develop appropriate control strategies.

A water canal may be envisaged as a series of different subsystems that are connected and interact among each other, and due to its characteristics several control algorithms have been developed throughout the years (Mareels et al. [2005]; Negenborn et al. [2009]; Malaterre and Baume [1998]) with the main objective concerning the water level regulation to optimize the resource distribution (Mareels et al. [2005]). These control techniques range from centralized to decentralized and distributed approaches (Negenborn et al. [2009]), with several adaptive strategies being taken into account (Rijo et al. [2009]; Lemos [2009]).

Considering distributed strategies is a possible solution to improve the efficiency of the system and to tackle communication issues that may occur in centralized and decentralized strategies (Costa [2013]). The distributed control algorithms take into account the interactions between subsystems in coordination processes where the local controllers negotiate with each other with the objective of reaching a consensus (Costa [2013]; Pinto [2011]; Sampaio [2012]; Igreja et al. [2011]). The distributed strategies considered in more detail in this study are defined in (Costa [2013]; Pinto [2011]; Igreja et al. [2011]; Lemos and Pinto [2012]) and they are based in optimal control LQG and in predictive control. In (Pinto [2011]; Igreja et al. [2011]) the coordination procedures rely on Game Theory concepts in which the local control agents negotiate until reaching a consensus situation where none of them benefits from changing his manipulated variable, a Nash equilibrium (Nash [1950]), whereas in (Costa [2013]) the distributed optimization algorithm uses augmented Lagrangian. Adding adaptation has the advantage of increasing the performance of the system, by contributing to its stability when facing variations in its dynamics due to external factors or disturbances such as water extraction.

Three adaptive distributed control algorithms are considered in this study. The first algorithm is a LQG controller whereas the second and third algorithms are predictive controllers. The coordination procedures were described above, and are based in Game Theory concepts or dual optimization. The identification algorithm considered is the Recursive Least Squares (RLS) that provides estimates of the system parameters using incremental input and output data retrieved from the system, that represent the variations of the gate positions and water levels with respect to the operating points.

2. SYSTEM IDENTIFICATION

This section is dedicated to the linear model identification, with a description of the water canal considered in this study followed by the definition of the model identification algorithms used and linear models considered.

2.1 Water Delivery Canal

The water canal considered in this dissertation belongs to Núcleo de Hidráulica e Controlo de Canais (NuHCC) of Univerisity of Évora, Portugal. In (Lemos et al. [2010]) a more extensive description of the canal is presented. It consists of a series of four pools separated by four vertical gates that are actuated by electrical motors. The first three pools are terminated by an undershot gate, whereas the fourth pool is terminated by an overshot gate.

In order to measure the upstream (M_i), centre (C_i) and downstream (J_i) water level, three sensors were placed along each

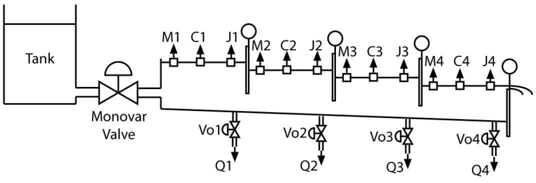

Fig. 1. Schematic representation of the NuHCC automatic canal.

pool i, as shown in figure 1, although only the last ones are considered in this study. The variables Q_i represent the side flow of off-take i, controlled by valves placed at the canal bottom, with control signals denoted by V_{Oi}. These variables allow the generation of disturbances in the system, being associated, in a practical situation, to water consumption by users.

The dynamics of the water flow in the canal is described by the non-linear hyperbolic Saint-Venant equations (Malaterre and Baume [1998]), obtained using the mass and momentum conservation principles and whose solution is obtained using numerical methods for solving Partial Differential Equations (Lemos et al. [2010]). In this study only the undershot gates are considered, and although the irrigation canal is a non-linear system with infinite order, in this study it is approximated by a finite-dimension linear model, and thus the control algorithms developed will address only linear finite dimensional State-Space (SS) models.

2.2 *Linear Incremental Models*

The target system considered is the SIMULINK non-linear canal model described in (Lemos et al. [2010]). The objective is to obtain finite-dimension models that describe the system dynamics. This requires the establishment of the model orders and structure. Since previous studies indicate that AutoRegressive with eXogenous input (ARX) models are suitable for this system (Pinto [2011]; Sampaio [2012]), the linear incremental model is described by

$$A(q^{-1})\Delta y(t) = B(q^{-1})\Delta u(t - n_K) + e(t), \qquad (1)$$

in which $\Delta y(t) \in \mathbb{R}$ represents the incremental output of the system, $\Delta u(t) \in \mathbb{R}$ the incremental input, $e(t) \in \mathbb{R}$ a Gaussian white noise sequence, q^{-1} is the unit delay operator, and $A(q^{-1})$ and $B(q^{-1})$ are polynomials in the unit delay operator, described by

$$A(q^{-1}) = 1 + a_1 q^{-1} + ... + a_{n_A} q^{-n_A}, \qquad (2)$$

$$B(q^{-1}) = b_0 + b_1 q^{-1} + ... + b_{n_B} q^{-n_B}, \qquad (3)$$

in which positive integers n_K, n_A and n_B represent the pure delay, and the number of zeros and poles. The identification of linear incremental models from the non-linear SIMULINK model requires the definition of an equilibrium point.

2.3 *MIMO Model Identification*

The multi-variable linear model defined in this section takes into account the first three pools and corresponding gates of the water canal, while the fourth gate is kept in its equilibrium position. The outputs and inputs of each subsystem are defined as the downstream water levels and gate positions, respectively. Instead of having $A(q^{-1})$ and $B(q^{-1})$ defined as polynomials that represent the zeros and poles of the system, these are now matrices with polynomial entries, with the underlying assumption that the water level in each pool depends only on the water level in the same pool in previous time instants and that the inputs only influence the water levels of neighbouring systems.

The signal chosen to excite the system input is the PRBS, since it is characterized for being a variable signal that excites the system along a wide range of frequencies. The function used to identify the model was *pem*, which implements the prediction error algorithm over the simulated data. In order to avoid the excitation of high-frequency modes and non-linearities, the data was filtered using a third-order low-pass Butterworth filter with a cut-off frequency of $0.1 rad/s$, as described in (Sampaio [2012]), and the mean and the initial transient of the signal were also removed. During the simulations that were conducted it was possible to verify that higher amplitude variations in the input signal led to poor identification results, with manifestations of non-linearities and that the period of the input signal had also influence in those results. The frequency has to be within a range of values that is sufficiently exciting to provide good estimates, and suitable to the slow system response, in order to allow the water level to stabilize. Using the gate positions as input signals did not produce satisfactory results, and therefore a different solution proposed in (Sampaio [2012]) was considered, that consisted in using a variable proportional to the flow drawn by each gate $q(t)$ as input signal. This new variable, $v(t)$, is related with the flow across the gate by the discharge coefficient C_{ds} as

$$v(t) = \frac{1}{C_{ds}} q(t) = u(t) * W * \sqrt{2g(h_u - h_d)}. \qquad (4)$$

The linear MIMO incremental model structure is given by

$$x(t+1) = Ax(t) + Bv(t) + e(t), \quad \Delta y(t) = Cx(t) \qquad (5)$$

$$A = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix}, \quad B \begin{bmatrix} B_{11} & B_{12} & 0 \\ B_{21} & B_{22} & B_{23} \\ 0 & B_{32} & B_{33} \end{bmatrix}, \qquad (6)$$

$$\Delta y(t) = \begin{bmatrix} C_{11} & 0 & 0 \\ 0 & C_{22} & 0 \\ 0 & 0 & C_{33} \end{bmatrix} x(t), \qquad (7)$$

with $v = [v_1 \; v_2 \; v_3]^T$ and $x = [x_1 \; x_2 \; x_3]^T$. The incremental gate positions $\Delta u_i(t)$ are computed using the relationship between $u_i(t)$ and the variable proportional to the flow drawn by each gate v_i

$$\Delta u_i(t) = \frac{v_i(t)}{W * \sqrt{2g(h_u - h_d)}}. \qquad (8)$$

The model structure (5) reflects the assumption that the different canal stretches interact only through the manipulated variables (gate positions) of the adjacent gates. This assumption leads to a good fit of the model and allows the application of the control methods described in subsequent sections.

2.4 *The effect of side takes*

The D-LQG strategy introduced in section 4 requires a model composed by several subsystems (pools) connected to their neighbours that interact with their manipulated variables and flow of lateral off-takes (Pinto [2011]; Sampaio [2012]). Each subsystem Σ_i, with i denoting the i-th pool, is seen as a MISO linear incremental model in which its output is the water level y_i of the corresponding pool and its manipulated variable is the position of the respective gate v_i. The interactions between neighbouring subsystems are assumed to be described by the respective gate positions v_{i-1}, v_{i+1}, considered later in the feed-forward control term, and the flows of the lateral off-takes Q_{i-1}, Q_i, Q_{i+1} are handled as accessible disturbances.

The identification procedure was divided into two different parts, in which in the first half, PRBS signals were applied simultaneously representing variations in the gate positions and flows of the lateral off-takes. In the second half of the experiment, gate positions were kept constant at their equilibrium position while varying the flows of lateral off-takes. It was assumed that the side off-take valves were open in the equilibrium, and thus a new operating point was considered.

The linear incremental model structure is described by

$$x(t+1) = Ax(t) + Bv(t) + \Phi_v v(t) + \Gamma_d Q(t) + e(t), \quad (9)$$

$$A = \begin{bmatrix} A_{11} & 0 & 0 \\ 0 & A_{22} & 0 \\ 0 & 0 & A_{33} \end{bmatrix}, \quad B = \begin{bmatrix} B_{11} & 0 & 0 \\ 0 & B_{22} & 0 \\ 0 & 0 & B_{33} \end{bmatrix}, \quad (10)$$

$$\Phi_v = \begin{bmatrix} 0 & B_{12} & 0 \\ B_{21} & 0 & B_{23} \\ 0 & B_{32} & 0 \end{bmatrix}, \quad \Gamma_d = \begin{bmatrix} \Gamma_{11} & \Gamma_{12} & 0 \\ \Gamma_{21} & \Gamma_{22} & \Gamma_{23} \\ 0 & \Gamma_{32} & \Gamma_{33} \end{bmatrix}, \quad (11)$$

3. RECURSIVE LEAST SQUARES

To identify the system dynamic or, in other words, to estimate its parameters, the solution needs to take into account that several observations are required and that memory management should be efficient. This leads us to a recursive solution, which in this dissertation will be the RLS algorithm that uses both incremental input and output data to estimate the parameters (Franklin et al. [1998]). Considering the generic transfer function defined for an ARX model in (1), it is possible to write an equivalent difference equation with delayed samples for an incremental model

$$\Delta y(t) = -\sum_{i=1}^{n_A} a_i \Delta y(t-i) + \sum_{i=0}^{n_B} b_i \Delta u(t-n_K-i) + e(t) \quad (12)$$

in which $t \geq 0$ is an integer that represents discrete-time, $\Delta u \in \mathbb{R}$ is the incremental manipulated variable, $\Delta y \in \mathbb{R}$ the incremental system output, with increments defined with respect to the operating point, $e \in \mathbb{R}$ represents white Gaussian noise, n_A is the pole order, n_B the zero order and n_K is the system delay. Taking into account expression (12), the regressor φ is defined as a vector with the delayed input and output data and the vector of the parameters to be estimated, Θ, is given by

$$\Theta' = [a_1 \;\; a_2 \;\; ... \;\; a_{n_A} \;\; b_0 \;\; ... \;\; b_{n_B}]\,. \quad (13)$$

For each observation, the model is described by

$$\Delta y(t) = \varphi'(t-1)\Theta + e(t). \quad (14)$$

Given N observations, the estimation of the vector of parameters Θ by $\hat{\Theta}$ is made by minimizing the following cost function:

$$J(\Theta) = \frac{1}{2N} \sum_{t=1}^{N} [\Delta y(t) - \Theta' \varphi(t-1)]^2. \quad (15)$$

To obtain the estimate by combining the previous estimates with new data, a recursive estimator is required, with the following elements:

- Vector of estimates $\hat{\Theta}(t-1)$ and previous auxiliary variables $P(t-1)$;
- New data $\Delta y(t)$, $\varphi(t-1)$.

With the combination of these elements it is possible to compute the new estimates $\hat{\Theta}(t)$ and the new auxiliary variables $P(t)$, in which P is the covariance matrix, symmetric and positive semi-definite. One may also consider defining a forgetting factor, λ, with values between 0 and 1 so that the algorithm weights less data from past. With smaller values of λ, the algorithm tracks better changes in the state and the convergence is faster since it retains less data. With larger values of λ, the algorithm becomes progressively slower to follow changes in the system dynamic since it retains more data but with less variations in the estimates.

With a fixed forgetting factor λ, the results obtained in the experiments were unsatisfactory and thus an alternative version of the RLS algorithm with variable forgetting factor, introduced in (Sanoff and Wellstead [1983]) is considered. The value of λ depends on the information available and on the current estimates. In order to do so, one must define a new variable ε that denotes the prediction error. One of the required parameters is the mean value of the prediction error, denoted as ε_0. In the experiments conducted with the adaptive control algorithms, the parameters of the three subsystems were defined as

$$\lambda_0 = \lambda_{min} = 0.98, \; P_0 = 0.01 \times I_{p\times p}, \; \varepsilon_0 = 5 \times 10^{-3}, \quad (16)$$

in which λ_0 and P_0 are the initial values of the forgetting factor and covariance matrix and p is the number of model parameters.

Algorithm 1 Recursive Least Squares (RLS) with variable exponential forgetting factor

```
Require: $\Delta y(t)$, $\varepsilon_0$ $P(t-1)$, $\lambda(t-1)$, and $\hat{\Theta}(t-1)$
function RLS($\Delta y, \hat{\Theta}, P, \varepsilon$)
    for $t$=1:$T$
        $\varepsilon(t) = \Delta y(t) - \varphi'(t-1)\hat{\Theta}(t-1)$
        $K(t) = \frac{P(t-1)\varphi(t-1)}{\lambda(t-1)+\varphi'(t-1)P(t-1)\varphi(t-1)}$
        $\hat{\Theta}(t) = \hat{\Theta}(t-1) + K(t)\varepsilon(t)$
        $\lambda(t) = 1 - [1 - \varphi'(t-1)K(t)]\varepsilon^2(t)/\varepsilon_0$
        If $\lambda(t) < \lambda_{min} \rightarrow \lambda(t) = \lambda_{min}$
        $P(t) = [I - K(t)\varphi'(t-1)]\frac{P(t-1)}{\lambda(t)}$
return $\hat{\Theta}(t)$
```

4. ADAPTIVE DISTRIBUTED LQG CONTROL

4.1 LQG Control Law with integral action

The Linear-Quadratic Gaussian (LQG) controller results from the combination of a Linear-Quadratic Regulator (LQR) and a Linear-Quadratic Estimator (LQE). This observer-controller is required because in the system here considered, it is not possible to access the state for direct measure, and therefore an estimate of the state is considered in the control law. The state estimate is defined as

$$\hat{x}(t|t) = \Phi_E \hat{x}(t-1|t-1) + \Gamma_E \Delta u(t-1) - M(e(t)), \quad (17)$$

with

$$\Phi_E = A - MCA, \; \Gamma_E = B - MCB, \; e(t) = r(t) - \Delta y(t), \quad (18)$$

in which $r(t)$ is the reference signal, M is the optimal gain matrix obtained by minimizing a cost function that depends on the covariance matrices Q_E and R_E related with the process and measurement noises (Sampaio [2012]). M is given by

$$M = PC^T(CPC^T + R_E)^{-1}, \quad (19)$$

with P denoting the positive definite solution of the algebraic Riccati equation. In order to guarantee that the system response follows the reference signal, the control design requires the inclusion of an integrator, defined by

$$x_I(t+1) = x_I(t) + T_s e(t), \tag{20}$$

with $e(t)$ denoting the output error defined by $e(t) = r(t) - \Delta y(t)$, and T_s is the sampling time. The state-space model of the system is now described by

$$\bar{x}(t+1) = \bar{A}\bar{x}(t) + \bar{B}\Delta u(t), \tag{21}$$

in which

$$\bar{x} = \begin{bmatrix} x \\ x_I \end{bmatrix}, \quad \bar{A} = \begin{bmatrix} A & \underline{0} \\ -T_s C & I \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} B \\ \underline{0} \end{bmatrix}, \tag{22}$$

with the system output being written as

$$\Delta y(t) = \bar{C}\bar{x}(t), \quad \bar{C} = [C \ \underline{0}] \,. \tag{23}$$

Expressions (21) and (23) define the augmented state-space model of the system with integral action, which is used in the computation of manipulated variables. The LQR takes into account the augmented state-space matrices ($\bar{A}$, $\bar{B}$, and $\bar{C}$) to compute the manipulated variables, that can be written as

$$\Delta u(t) = -\left[K \ K_I\right] \begin{bmatrix} \hat{x}(t) \\ x_I(t) \end{bmatrix}, \tag{24}$$

with

$$[K \ K_I] = (I + \rho^{-1}\bar{B}^T \bar{P}\bar{B})^{-1}\rho^{-1}\bar{B}^T\bar{P}\bar{B}, \tag{25}$$

in which $\bar{P}$ denotes the algebraic solution of the Riccati equation using the augmented state-space model.

4.2 The Distributed Controller Structure

The state model considered in this section is the model defined by (9), in which $v(t)$ and $Q(t)$ denote the accessible disturbances regarding the manipulated variables of neighbouring subsystems and the flow of the lateral off-takes. By including an integrator in the multi-variable model defined by (9), its dynamics becomes described by

$$\bar{x}(t+1) = \bar{A}\bar{x}(t) + \bar{B}v(t) + \bar{\Gamma}d(t), \quad \Delta y(t) = \bar{C}\bar{x}(t), \tag{26}$$

with $\bar{\Gamma} = \left[[\Phi_v \ \Gamma_d] \ \underline{0}\right]^T$ and the disturbance vector, denoted by d, defined as

$$d(t) = \left[v_{i-1}(t) \ v_i(t) \ v_{i+1}(t) \ Q_{i-1}(t) \ Q_i(t) \ Q_{i+1}(t)\right]^T. \tag{27}$$

The controller structure is represented in figure 2, in which it is possible to verify how the local control agents communicate with each other.

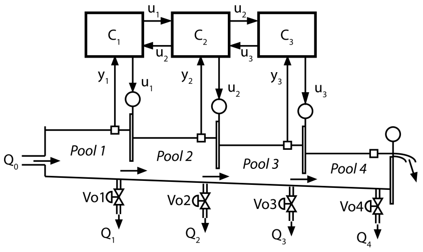

Fig. 2. Schematic representation of the distributed controller structure.

At each time instant t, each controller has access to the manipulated variables of its neighbours and to the flow of the lateral off-takes of the respective pool and of neighbouring susbsystems. This information exchange between local control agents is crucial to the coordination procedure. The minimization of the LQR quadratic cost function taking into account the augmented state model is accomplished using the discrete-time version of the Pontryagin Minimum Principle, as described in (Pinto [2011]) and (Sampaio [2012]). This solution leads to the following control law

$$v(t) = -\left[K \ K_I\right]\bar{x}(t) + v_{ff}(t), \tag{28}$$

in which v_{ff} is the vector of feedforward control variables and the gains $[K \ K_I]$ are computed using (25).

4.3 The Coordination Procedure

The distributed procedure considered in this study, defined in (Lemos et al. [2013] and Pinto [2011]), applied to serially chained systems, is an iterative procedure during which the local control agents communicate with each other to compute the corresponding manipulated variables. In order to define the algorithm, the letter j will be used to define the iterations, i to identify the subsystem and t is the discrete time. The procedure begins by initializing the gate positions v with the previous values, while the flow of the lateral off-takes Q is read directly from the sensors

$$d_{i,j=0}(t) = \left[v(t-1) \ Q(t)\right]^T. \tag{29}$$

During a number of predefined iterations n_I, the expression (28) is used to compute the new manipulated variables $v_{i,j}(t)$,

$$v_{i,j}(t) = -\left[K \ K_I\right]\bar{x}(t) + K_{ff}d_{i,j}(t). \tag{30}$$

Although the disturbance vector d is composed by accessible disturbances associated with the gate positions v and with the flows of lateral off-takes Q, the iterative procedure considers only the gate positions, and thus it can be written as

$$v_{i,j}(t) = \left[K \ K_I\right]\bar{x}(t) + \left[K_{ff,v} \ K_{ff,Q}\right]d_{i,j}(t), \tag{31}$$

$$d_{i,j}(t) = \begin{bmatrix} v_j(t) \\ Q(t) \end{bmatrix}. \tag{32}$$

Since the local control agents find their optimal manipulated variables with knowledge of the neighbors decisions, the procedure converges to the Nash Equilibrium (Lemos and Pinto [2012], Nash [1950]), a situation where no local controller benefits by changing only its manipulated variable. In order for the algorithm to reach convergence, the spectral radius of $K_{ff,v}$ needs to satisfy the condition

$$max|\lambda(K_{ff,v})| < 1, \tag{33}$$

where $\lambda(K_{ff,v})$ represents the eigenvalues of the matrix $K_{ff,v}$.

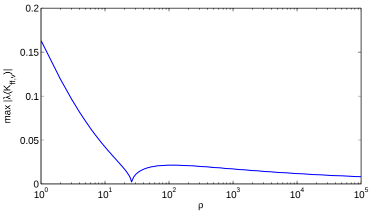

Fig. 3. Spectral radius of $K_{ff,v}$ as a function of the quadratic cost weights ρ_i. It is assumed that the cost weights have the same value.

The condition (33) was verified in experiments conducted with different values of quadratic weights ρ. In order to simplify the design it is assumed that the quadratic weights of the three pools, ρ_i, have the same value, and $n_I = 10$. The results of these experiments are represented in figure 3, where it is possible to see that the condition is valid for ρ between 1 and 1×10^5. However after conducting experiments with the SIMULINK canal model, with the same value of maximum iterations and with the observer parameters defined as $R_{E,i} = 1 \times 10^3$ and $Q_E = I_{3\times3}$, it was verified that the closed-loop response of the system was only acceptable for $\rho_i \geq 5 \times 10^3$. The value defined for the quadratic weight was $\rho_i = 1 \times 10^4$ and the closed-loop system response is represented in figures 4 and 5.

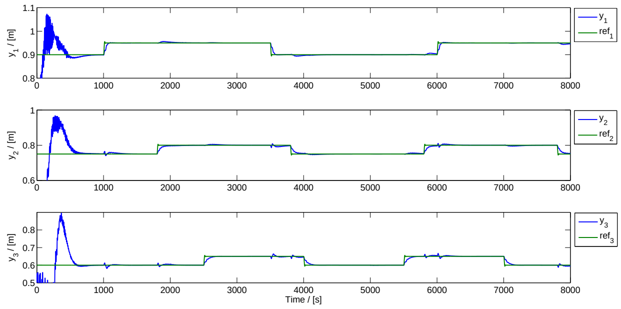

Fig. 4. Closed-loop response of the SIMULINK non-linear canal model, controlled by the D-LQG controller. (Output and Reference signals)

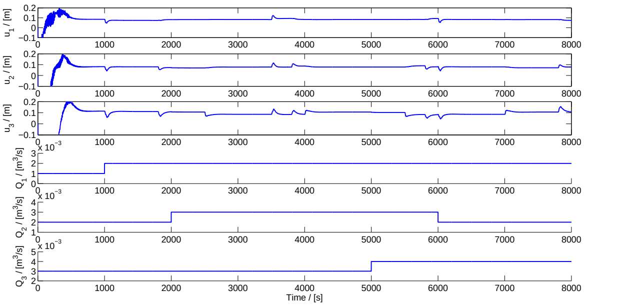

Fig. 5. Closed-loop response of the SIMULINK non-linear canal model, controlled by the D-LQG controller. (Input signals and flows of lateral off-takes)

4.4 Adaptive Controller

The adaptive strategy is based in a two-step sequence in which first the RLS estimates the system parameters using input and output data obtain from the plant, and after that, a model defined with those estimates is considered in the controller step, based on the D-LQG algorithm described above. During a pre-specified period of time t_I, only the identification step is working, in order to guarantee that the estimates are closer to convergence by the time the controller is activated. Without guaranteeing this, it is possible to occur either stability or identification issues. During this period of time, the system is excited by a PRBS signal of amplitude $0.001m$ around the equilibrium points, that remains active during the whole experiment to provide enough excitation to the system, preventing possible identification issues. The adaptive controller parameters were defined as $t_I = 4 \times 10^4 s, \rho_i = 5 \times 10^4, R_{E,i} = 1 \times 10^3$ and $Q_E = I_{3\times3}$.

Algorithm 2 Adaptive D-LQG

```
Initialization of parameter estimates $(\theta_1, \theta_2, \theta_3)$ and respective covariance matrices $(P_1, P_2, P_3)$.
for each time instant $t$
    Computation of $\theta_i(t)$, using algorithm 1
    Definition of augmented state models of each subsystem
        $\bar{x}_i(t+1) = \bar{A}_i\bar{x} + \bar{B}_i v_i(t) + \bar{\Gamma}_i d_i(t)$
        $\Delta y(t) = \bar{C}_i\bar{x}_i(t)$
    if $t > t_I$ then
        Regulator and Observer parameters: $\rho_i$, $R_{E,i}$, $Q_E$
        Number of iterations: $n_I$
        Computation of LQR gains: $[K_i \;\; K_{I,i}]$
        Computation of LQE gain: $M_i$
        Computation the state estimates $\hat{x}_i$
        $v_{fb,i}(t) = [K_i \;\; K_{I,i}] \begin{bmatrix} \hat{x}_i(t) \\ x_{I,i}(t) \end{bmatrix}$
        Initialize manipulated variables: $v_{i,j=0}(t) = v_i(t-1)$
        for $j = 1 : n_I$
            $v_{ff,i,j}(t) = [K_{ff,i,v} \;\; K_{ff,i,Q}]\, d_{i,j}(t)$
        end
        Computation of manipulated variables:
        $v_i(t) = v_{fb,i}(t) + v_{ff,i,n_I}(t)$
```

The results obtained during an experiment with algorithm 2 are represented in figures 6 and 7, where it is shown the open-loop and closed-loop response of the system controlled by the adaptive D-LQG algorithm. It is possible to verify how the outputs of the three pools converge to the reference signals. Comparing with the non-adaptive algorithm, the system response adaptive strategy appears to be a close approximation, even with an additional small amplitude PRBS signal in the input of the system. The quadratic cost weights ρ_i were also increased to 5×10^4 in order to improve the system response and prevent stability and oscillation issues.

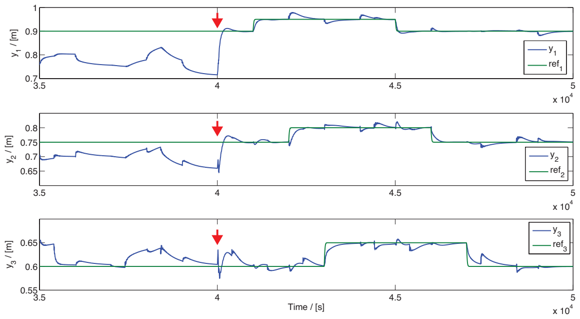

Fig. 6. Open-loop and closed-loop response of the SIMULINK non-linear canal model, controlled by the adaptive D-LQG controller. (Output and Reference signals)

5. ADAPTIVE DISTRIBUTED MPC

Model Predictive Control is a feedback strategy in which the manipulated variable is obtained using predictions of the system dynamics that take into account its model, with the advantage of handling both input and state constraints. Its control law consists in the optimization of a quadratic cost function that depends on the forecasts of the system behavior during a predefined finite horizon N. The algorithms developed take into account the linear incremental models identified around an equilibrium point of the system, as shown in section 2. Here, one considers a quadratic cost function associated to a

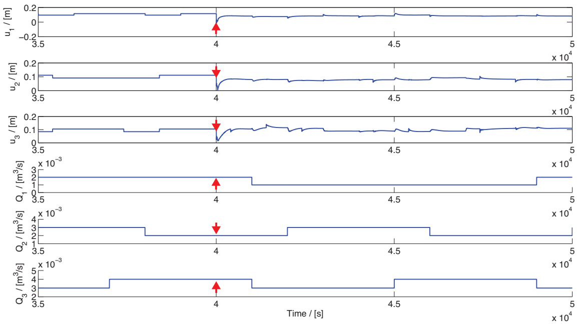

Fig. 7. Open-loop and closed-loop response of the SIMULINK non-linear canal model, controlled by the adaptive D-LQG controller. (Input signals and flows of lateral off-takes)

predefined horizon N (Rawlings and Mayne [2009]) that slides alongside with the current state estimates, while taking into account the constraints, in order to find a sequence of manipulated variables that minimizes

$$J_N = \sum_{i=1}^{N} (y(t+i)-r(t+i))^T (y(t+i)-r(t+i)) + \rho u^2(t+i-1). \tag{34}$$

In the strategies based on the MPC theory, an integrator was included in series with the controller, and thus taking into account (5), the state model of the i-th system, with integral effect is defined as

$$\begin{bmatrix} x_i(t+1) \\ z_{i-1}(t+1) \\ z_i(t+1) \\ z_{i+1}(t+1) \end{bmatrix} = \begin{bmatrix} A_i & B_i \\ \underline{0} & I \end{bmatrix} \begin{bmatrix} x_i(t) \\ z_{i-1}(t) \\ z_i(t) \\ z_{i+1}(t) \end{bmatrix} + \begin{bmatrix} \underline{0} \\ I \end{bmatrix} V_i(t), \tag{35}$$

$$V_i(t) = \left[v_{i-1}(t) \;\; v_i(t) \;\; v_{i+1}(t)\right]^T, \tag{36}$$

$$\Delta y(t) = [C \;\; \underline{0}] \begin{bmatrix} x_i(t) \\ z_{i-1}(t) \\ z_i(t) \\ z_{i+1}(t) \end{bmatrix}, \quad B_i = \left[B_{i,i-1} \;\; B_{i,i} \;\; B_{i,i+1}\right], \tag{37}$$

with

$$z_i(t) = \Delta u_i(t) + z(t). \tag{38}$$

The model can be written in a more compact form as

$$\bar{x}_i(t+1) = \bar{A}\bar{x}_i(t) + \bar{B}_{i,i-1}v_{i-1}(t) + \bar{B}_{i,i}v_i(t) + \bar{B}_{i,i+1}v_{i+1}(t), \tag{39}$$

$$\Delta y(t) = \bar{C}\bar{x}(t). \tag{40}$$

With the augmented state matrices it is possible to define the model predictor as

$$Y_i = \mathbf{\Pi}_i \begin{bmatrix} \hat{x}_i \\ z_{i-1}(t) \\ z_i(t) \\ z_{i+1}(t) \end{bmatrix} + \left[\mathbf{W}_{i,i-1} \;\; \mathbf{W}_{i,i} \;\; \mathbf{W}_{i,i+1}\right] \begin{bmatrix} \Delta U_{i-1} \\ \Delta U_i \\ \Delta U_{i+1} \end{bmatrix} \tag{41}$$

with

$$\mathbf{W}_{i,j} = \begin{bmatrix} \bar{C}_i\bar{B}_{i,j} & 0 & \dots & 0 \\ \bar{C}_i\bar{A}_i\bar{B}_{i,j} & \bar{C}_i\bar{B}_{i,j} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ \bar{C}_i\bar{A}_i^{i-1}\bar{B}_{i,j} & \bar{C}_i\bar{A}_i^{i-2}\bar{B}_{i,j} & \dots & \bar{C}_i\bar{B}_{i,j} \end{bmatrix}, \mathbf{\Pi} = \begin{bmatrix} \bar{C}_i\bar{A}_i \\ \bar{C}_i\bar{A}_i^2 \\ \dots \\ \bar{C}_i\bar{A}_i^i \end{bmatrix}, \tag{42}$$

$$Y_i = \begin{bmatrix} \Delta y_i(t+1) \\ \Delta y_i(t+2) \\ \dots \\ \Delta y_i(t+N) \end{bmatrix}, \quad \Delta U_i = \begin{bmatrix} \Delta u_i(t) \\ \Delta u_i(t+1) \\ \dots \\ \Delta u_i(t+N-1) \end{bmatrix}. \tag{43}$$

The local cost functions associated with each subsystem i are defined as

$$J_i = (Y_i - R_i)^T(Y_i - R_i) + \rho_i \Delta U_i^T \Delta U_i, \tag{44}$$

in which R_i is the reference vector, and Y can be written in a more compact notation as

$$Y_i = \mathbf{\Pi}_i \begin{bmatrix} \hat{x}_i \\ z_{i-1}(t) \\ z_i(t) \\ z_{i+1}(t) \end{bmatrix} + \mathbf{W}_i \bar{\Delta U}_i, \tag{45}$$

$$\mathbf{W}_i = \left[\mathbf{W}_{i,i-1} \;\; \mathbf{W}_{i,i} \;\; \mathbf{W}_{i,i+1}\right], \quad \bar{\Delta U}_i = \begin{bmatrix} \Delta U_{i-1} \\ \Delta U_i \\ \Delta U_{i+1} \end{bmatrix}. \tag{46}$$

5.1 *Distributed Model Predictive Control based on D-ADMM*

This first strategy is based on a distributed optimization algorithm named Distributed Alternating Direction Method of Multipliers (D-ADMM) that solves problems in networks of interconnected nodes, that represent the subsystems and that have a local cost function J_i associated with them (Mota et al. [2013]). The global cost function J is the sum of all local cost functions J_i. The network structure considered in the problem formulation of D-ADMM is represented in figure 2 in which a series of interconnected subsystems (nodes) Σ_i is associated to a local controller C_i with a local cost function that depends on the manipulated variable of the corresponding node and on copies of the manipulated variables of its neighbours $J_i(\Delta U_{i-1}, \Delta U_i, \Delta U_{i+1})$.

The implementation of the D-ADMM considered in this study was introduced in (Costa [2013]), where dual variables are associated to the edges (γ_1 and γ_2) and a cost weight ρ_A is required for the cooperation part of the algorithm.

With this being said, the D-ADMM cost functions associated with each subsystem i, in a compact notation, are defined as

$$J_{i,A} = \bar{\Delta U}_i^T \Psi_i \bar{\Delta U}_i + \bar{\Delta U}_i^T \Phi_i + \Upsilon_i, \tag{47}$$

where Υ_i represents the terms that do not depend on the respective manipulated variable, and Ψ_i and Φ_i are defined in detail in algorithm 3. The analytical minimization of cost functions $J_{i,A}$ is accomplished by computing the derivative in order to the respective manipulated variable and finding the value for which $\frac{\partial J_{i,A}}{\partial \bar{\Delta U}_i}$ is equal to 0. The derivative of the D-ADMM cost function $J_{i,A}$ (47) in order to $\bar{\Delta U}_i$ is given by

$$\frac{\partial J_{i,A}}{\partial \bar{\Delta U}_i} = 2\bar{\Delta U}_i \Psi_i + \Phi_i. \tag{48}$$

The values of the manipulated variables $\bar{\Delta U}_i^*$ that minimize the cost functions are given by expression

$$\frac{\partial J_{i,A}}{\partial \bar{\Delta U}_i} = 0 \;\; \Leftrightarrow \;\; \bar{\Delta U}_i^* = -\frac{1}{2}\Psi_i^{-1}\Phi_i. \tag{49}$$

In the distributed strategies here considered, no constraints were taken into account in the optimization problem. In this case, the introduction of constraints would made impossible to use the analytical solution. The D-MPC strategy based on D-ADMM is introduced in algorithm 3. There are two parameters regarding the D-ADMM algorithm, that need to be pre-defined before conducting experiments: the cost weight ρ_A, and the maximum number of iterations n_I. The values defined for the quadratic cost weights associated with each subsystem ρ_i were defined in previous experiments conducted with the centralized MPC controller.

Algorithm 3 D-MPC based on D-ADMM with edge-associated dual variables

```
  Initialization of manipulated and dual variables: $\gamma_1 = 0; \gamma_2 = 0; \bar{\Delta U}_1 = 0; \bar{\Delta U}_2 = 0; \bar{\Delta U}_3 = 0$.
  repeat
      $\Phi_1 = 2\mathbf{W}_1^T(\mathbf{\Pi}_1\bar{x}_1 - R_1) - \gamma_1 - \rho_A\bar{\Delta U}_2$
      $\Psi_1 = \mathbf{W}_1^T\mathbf{W}_1 + \bar{\rho}_1 + \frac{\rho_A}{2}I$
      $\bar{\Delta U}_1 = -\frac{1}{2}\Psi_1^{-1}\Phi_1$
      $\Phi_3 = 2\mathbf{W}_3^T(\mathbf{\Pi}_3\bar{x}_3 - R_3) + \gamma_2 - \rho_A\bar{\Delta U}_2$
      $\Psi_3 = \mathbf{W}_3^T\mathbf{W}_3 + \bar{\rho}_3 + \frac{\rho_A}{2}I$
      $\bar{\Delta U}_3 = -\frac{1}{2}\Psi_3^{-1}\Phi_3$
      $\Phi_2 = 2\mathbf{W}_2^T(\mathbf{\Pi}_2\bar{x}_2 - R_2) - (\gamma_1 - \gamma_2) - \rho_A(\bar{\Delta U}_1 + \bar{\Delta U}_3)$
      $\Psi_2 = \mathbf{W}_2^T\mathbf{W}_2 + \bar{\rho}_2 + 2\rho_A I$
      $\bar{\Delta U}_2 = -\frac{1}{2}\Psi_2^{-1}\Phi_2$
      $\gamma_1 = \gamma_1 - \rho_A(\bar{\Delta U}_1 - \bar{\Delta U}_2)$
      $\gamma_2 = \gamma_2 - \rho_A(\bar{\Delta U}_2 - \bar{\Delta U}_3)$
  until pre-defined maximum number of iterations $n_I$ reached or stopping criteria is met
```

In order to select values for the D-ADMM parameters several experiments were conducted with different combinations of values and the results are shown in figure 8. It is possible to verify that initially, with less iterations and with a lower weight ρ_A, the output error is higher. The value tends to decrease with more iterations, and with a higher value for ρ_A. From figure 8, the combination of values defined to be used in the experiments is $\rho_A = 80$ and $n_I = 20$, since the associated output error is close to the minimum and a smaller maximum number of iterations is better in terms of computational time.

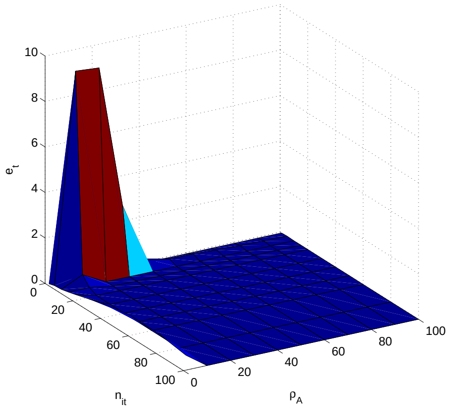

Fig. 8. Variation of the total output error e_t with different combinations of cost weight ρ_A and the maximum number of iterations n_I. The value of the quadratic cost weights were $\rho_1 = 200$, $\rho_2 = 600$, $\rho_3 = 400$ and $N = 35$.

The results obtained by applying the D-MPC algorithm based on D-ADMM with the parameters defined above are represented in figures 9 and 10. The system outputs converge towards the respective reference signals, but with a small amplitude output error even with integral action, that appears to be related with the interactions between pools, with the current water levels on each pool or with the position of the integrator. The effects of the interactions between subsystems are visible whenever occur variations in the water levels.

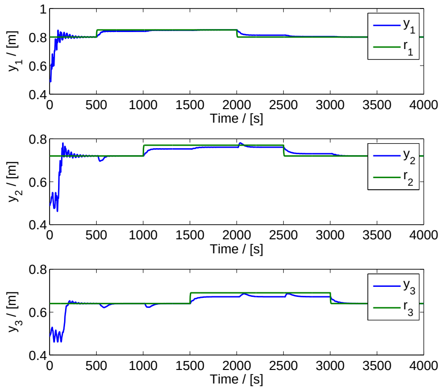

Fig. 9. Closed-loop response of the system of the SIMULINK non-linear canal model controlled by the D-MPC controller based on D-ADMM. (Output and Reference signals)

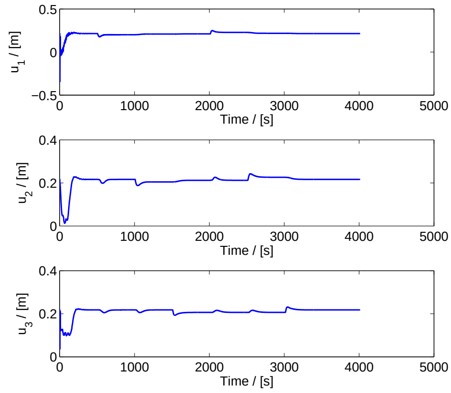

Fig. 10. Closed-loop response of the system of the SIMULINK non-linear canal model controlled by the D-MPC controller based on D-ADMM. (Input signals)

5.2 *Adaptive Controller*

Similar to the adaptive D-LQG strategy, the adaptive D-MPC controller based on D-ADMM is also divided into a two step identification/control sequential procedure. The identification step is accomplished with the RLS algorithm and during an initial period of time $t < t_I$, the control step is switched off. When $t \geq t_I$, the D-MPC algorithm is activated and the control strategy becomes a sequence of two main steps: an identification step and a control step. At each time instant t, after estimating the values of the parameters, a system model is defined followed the computation of the manipulated variables using algorithm 4. The D-ADMM parameters considered in the adaptive controller were defined in the previous section and the quadratic cost weights considered are $\rho_1 = 2000$, $\rho_2 = 1000$, and $\rho_3 = 2000$.

In this adaptive D-MPC approach, in order to guarantee that the parameters estimates are closer to convergence to prevent stability issues, the value defined for the time instant in which the controller is switched on is $t_I = 2 \times 10^5 s$. In figures 11 and 12 it is represented the system closed-loop response with the adaptive distributed controller. The higher value for the quadratic cost weights was again the considered solution

Algorithm 4 Adaptive D-MPC based on D-ADMM with edge-associated dual variables

```
Initialization of manipulated and dual variables: $\gamma_1 = 0; \gamma_2 = 0; \bar{\Delta U}_1 = 0; \bar{\Delta U}_2 = 0; \bar{\Delta U}_3 = 0$.
Initialization of parameter estimates $(\theta_1, \theta_2, \theta_3)$ and respective covariance matrices $(P_1, P_2, P_3)$.
for each time instant $t$
    Computation of parameters $\theta_i(t)$, using algorithm 1
    Define the augmented models of each subsystem
        $\bar{x}_i(t) = \bar{A}\bar{x}_i(t-1) + \bar{B}_i \Delta U_i$
        $\Delta y(t) = \bar{C}\bar{x}(t)$
    if $t > t_I$ then
        Predictor model:
            Computation of $\mathbf{\Pi}_i$ and $\mathbf{W}_i$
            $Y_i = \mathbf{\Pi}_i \bar{x} + \mathbf{W}_i \Delta U_i$
        Iterative procedure described in algorithm 3.
```

to handle with existing high-frequency oscillations, and despite the existing output error, mostly visible in the second pool response, the water levels converged to the reference signals. By the time instant the controller step is switched on, the parameter estimates are close to convergence.

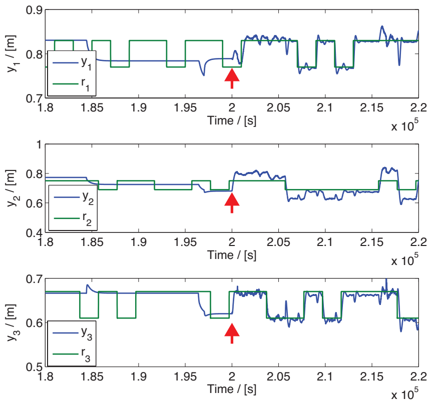

Fig. 11. System response of the SIMULINK non-linear canal model controlled by the adaptive D-MPC controller based on D-ADMM. (Output and Reference signals)

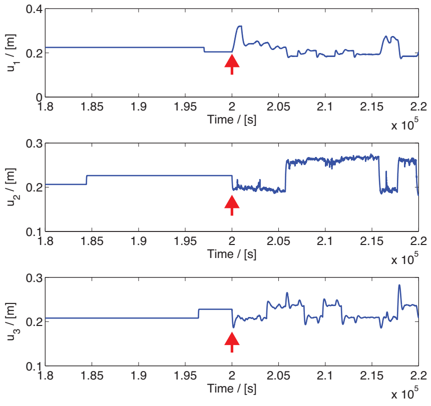

Fig. 12. System response of the SIMULINK non-linear canal model controlled by the adaptive D-MPC controller based on D-ADMM. (Input signals)

5.3 Distributed Model Predictive Control based on Game Theory concepts

In this approach, the D-MPC algorithm to be considered is based on Game Theory concepts, in the sense that each controller must optimize its control variable, taking into account the knowledge of the manipulated variables computed by its neighbours, similarly as the D-LQG algorithm in section 4. If each controller computes its manipulated variable with knowledge of the control inputs of the neighbouring subsystems, the goal is to reach a situation in which no controller benefits from changing the manipulated variable, the Nash equilibrium (Lemos and Pinto [2012]). This coordination process, similar to control strategies introduced in (Igreja et al. [2011]), is defined in this study as an alternative to the D-MPC algorithm based on D-ADMM. The model of each subsystem, with integral effect, is written as

$$\begin{bmatrix} x_i(t+1)\\ d_{i-1}(t+1)\\ z_i(t+1)\\ d_{i+1}(t+1)\end{bmatrix} = \begin{bmatrix} A_i & [\Gamma_{i,i-1} \;\; B_i \;\; \Gamma_{i,i+1}] \\ \underline{0} & I \end{bmatrix} \begin{bmatrix} x_i(t)\\ d_{i-1}(t)\\ z_i(t)\\ d_{i+1}(t)\end{bmatrix} + \begin{bmatrix}\underline{0}\\ I\end{bmatrix} V_i(t), \tag{50}$$

$$V_i(t) = [v_{i-1}(t) \;\; v_i(t) \;\; v_{i+1}(t)]^T, \tag{51}$$

$$\Delta y(t) = [C \;\; \underline{0}] \begin{bmatrix} x_i(t)\\ d_{i-1}(t)\\ z_i(t)\\ d_{i+1}(t)\end{bmatrix}, \tag{52}$$

with d denoting the manipulated variables of the neighbouring subsystems, treated as accessible disturbances. The model can be written in a more compact notation as

$$\bar{x}_i(t+1) = \bar{A}\bar{x}_i(t) + \left[\bar{\Gamma}_{i,i-1} \;\; \bar{B}_i \;\; \bar{\Gamma}_{i,i+1}\right] V_i(t), \tag{53}$$

$$\Delta y(t) = \bar{C}\bar{x}(t). \tag{54}$$

The predictor model is given by (41) and the local cost functions by (44). The Markov parameters matrices $\mathbf{W}_{i,j}$ for $j \neq i$ are similar to (42), but with $\bar{B}_{i,j}$ replaced by $\bar{\Gamma}_{i,j}$. The remaining predictor model matrices remain equal. The minimum of each local cost function is solved by computing its derivative in order to the corresponding manipulated variable and finding the solution of $\frac{\partial J_i}{\partial \Delta U_i} = 0$, which results in an iterative procedure described by

$$\bar{\Delta U} = -\frac{1}{2}M^{-1}\Psi - \frac{1}{2}M^{-1}\Phi\bar{\Delta U}, \tag{55}$$

with

$$M_i = \mathbf{W}_{i,i}^T \mathbf{W}_{i,i} + \rho_i I, \tag{56}$$

$$M = \begin{bmatrix} M_1 & \underline{0} & \underline{0} \\ \underline{0} & M_2 & \underline{0} \\ \underline{0} & \underline{0} & M_3 \end{bmatrix}, \tag{57}$$

$$\Phi = \begin{bmatrix} \underline{0} & 2\mathbf{W}_{1,1}^T\mathbf{W}_{1,2} & \underline{0} \\ 2\mathbf{W}_{2,2}^T\mathbf{W}_{2,1} & \underline{0} & 2\mathbf{W}_{2,2}^T\mathbf{W}_{2,3} \\ \underline{0} & 2\mathbf{W}_{3,3}^T\mathbf{W}_{3,2} & \underline{0} \end{bmatrix}, \tag{58}$$

$$\psi_i = 2\mathbf{W}_{i,i}^T \mathbf{\Pi}_i \bar{x}_i, \tag{59}$$

$$\Psi = [\psi_1 \;\; \psi_2 \;\; \psi_3]^T. \tag{60}$$

The procedure will converge if the spectral radius

$$\lambda_{max} := \max \lambda(M^{-1}\Phi) \tag{61}$$

verifies $|\lambda_{max}| < 1$ (Igreja et al. [2011]). As it is possible to verify in figure 13, where it is represented the influence of the quadratic cost weights ρ_i in the spectral radius, by fixating $\rho_1 = 200$, λ_{max} tends to decrease when the quadratic cost weights increase, which influences the rate of convergence of the iterative procedure. The selected values of the quadratic cost weights are $\rho_1 = 200$, $\rho_2 = 800$, and $\rho_3 = 100$, with the same value of maximum iterations $n_I = 20$.

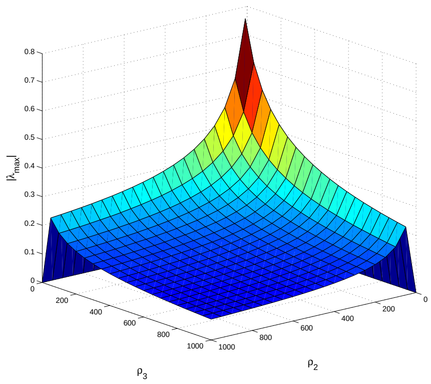

Fig. 13. Spectral radius λ_{max}.

The results of the experiment conducted in the SIMULINK non-linear canal model, after defining the controller parameters, in figures 14 and 15 are similar to the results obtained with the D-MPC algorithm based on D-ADMM, with the output tracking the reference with the same issue regarding the output error, even with integral action, and a similar system response. In terms of performance, the D-MPC algorithm based on Game Theory concepts, at least with an equal number of maximum iterations, appears to have a similar computational time and load when compared with the D-ADMM approach.

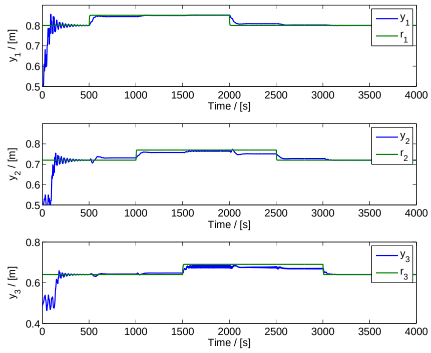

Fig. 14. Closed-loop response of the system of the SIMULINK non-linear canal model controlled by the D-MPC controller based on Game Theory concepts. (Output and Reference signals)

5.4 *Adaptive Controller*

Following the design of a D-MPC algorithm based on Game Theory concepts, comes the definition of a corresponding adaptive control strategy, similar to the one defined in section 4.4, with the only relevant change being the control algorithm considered, by using the iterative procedure introduced described in the section above. Regarding the parameters of the controller, the number of iterations and the finite-time horizon remain the same, whereas the quadratic weight costs considered were the ones used in section 4.4 ($\rho_1 = 2000$, $\rho_2 = 1000$, $\rho_3 = 2000$).

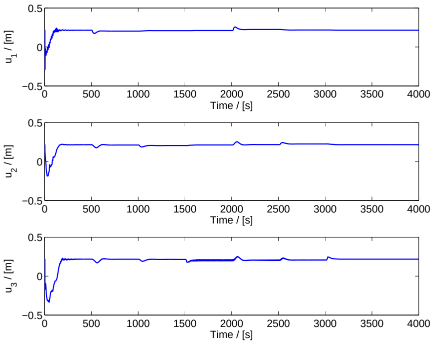

Fig. 15. Closed-loop response of the system of the SIMULINK non-linear canal model controlled by the D-MPC controller based on Game Theory concepts. (Input signals)

Algorithm 5 Adaptive D-MPC with neighbouring agent coordination

```
Initialization of manipulated variables: $\Delta\bar{U}_1 = 0; \Delta\bar{U}2 = 0; \Delta\bar{U}_3 = 0.$
$\Delta\bar{U} = \left[\Delta\bar{U}_1 \ \Delta\bar{U}_2 \ \Delta\bar{U}_3\right]^T$
Initialization of parameter estimates $(\theta_1, \theta_2, \theta_3)$ and respective covariance matrices $(P_1, P_2, P_3)$.
for each time instant $t$
    Computation of parameters $\theta_i(t)$, using algorithm 1
    Define the augmented models of each subsystem
        $\bar{x}_i(t+1) = \bar{A}\bar{x}_i(t) + \left[\bar{\Gamma}_{i,i-1} \ \bar{B}_i \ \bar{\Gamma}_{i,i+1}\right] V_i(t)$
        $\Delta y(t) = \bar{C}\bar{x}(t)$
    if $t > t_I$ then
    repeat
        $\Delta\bar{U} = -\frac{1}{2}M^{-1}\Psi - \frac{1}{2}M^{-1}\Phi\Delta\bar{U}$
    until pre-defined maximum number of iterations $n_I$ reached or stopping criteria is met
```

The time instant the control step is switched on remains as $t_I = 2 \times 10^5 s$.

The results of the experiment conducted in the SIMULINK non-linear canal model are represented in figures 16 and 16. Despite the output error and the oscillatory behavior, the results are practically similar to the ones obtained with the adaptive D-MPC algorithm based on D-ADMM, with a higher output error in the second pool, that appears to be related with the interactions between subsystems. As for the first and third pools, the outputs converge to the reference and appear to be less sensitive to variations in neighbouring subsystems. Regarding the parameter estimates, by the time the controller step is switched on, these are close to convergence.

6. CONCLUSIONS

It would be interesting to define adaptive strategies with a different identification algorithm such as the Recursive LASSO, and to define strategies that allow the parameter estimates initialization with a larger uncertainty. Comparing the adaptive and non-adaptive control strategies, the first ones resulted, in general, in more oscillatory system responses, but since the outputs followed the respective reference signals, the results were satisfactory. The higher computational load and time of the adaptive strategies is compensated by the adaptability to changes in the system dynamic.

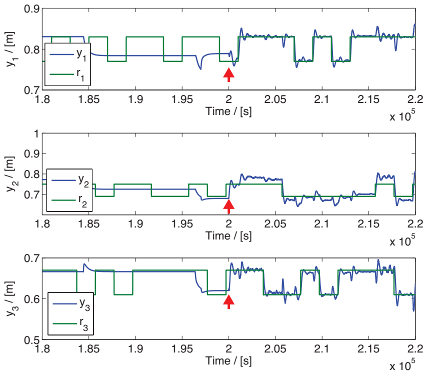

Fig. 16. System response of the system of the SIMULINK non-linear canal model controlled by the adaptive D-MPC controller based on Game Theory concepts. (Output and Reference signals)

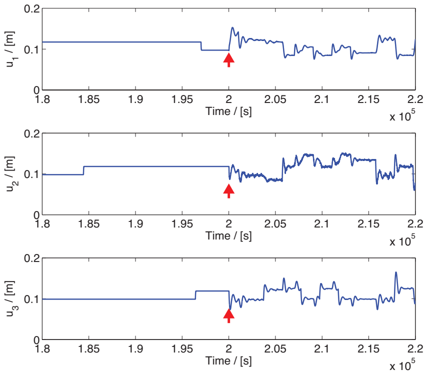

Fig. 17. System response of the system of the SIMULINK non-linear canal model controlled by the adaptive D-MPC controller based on Game Theory concepts. (Input signals)

It was interesting to see how the performance of the distributed techniques was close to the corresponding centralized solutions, with less computational load and communication steps required. In general, the adaptive distributed algorithms here considered have similar performances, with the D-LQG requiring less time for the RLS algorithm to work isolated. The algorithms based on Game Theory concepts have simpler negotiation/coordination strategies, with iterative procedures that require less operations and auxiliary variables. Since there is the possibility of the Nash equilibrium being far from the global minimum, an alternative strategy is considered based on Lagrangian optimization (Costa et al. [2014], Lemos and Pinto [2012]). The results obtained with the adaptive D-MPC algorithms are also similar, with a more oscillatory response than the one obtained with the adaptive D-LQG algorithm.

Although there is still work to be done with new ideas and solutions to explore regarding adaptive and distributed algorithms applied to water canals, three different algorithms were developed and studied with satisfactory results, complementing the work already developed and creating new challenges for future research.

REFERENCES

Costa, R.P. (2013). *Dual Optimization based Distributed Predictive Control of a Water Delivery Canal*. Master's thesis, Instituto Superior Técnico, Lisboa.

Costa, R.P., Lemos, J.M., Mota, J.F.C., and Xavier, J.M.F. (2014). D-ADMM based distributed MPC with input-output models. *2014 IEEE Conference on Control Applications, CCA 2014*, 699–704.

FAO (2012). *Coping with water scarcity: An action framework for agriculture and food security*. Food and Agriculture Organization of the United Nations.

Franklin, G.F., Workman, M.L., and Powell, J.D. (1998). *Digital Control of Dynamic Systems*. Wesley, 3 edition.

Igreja, J.M., Cadete, F.M., and Lemos, J.M. (2011). Application of distributed model predictive control to a water delivery canal. *2011 19th Mediterranean Conference on Control & Automation (MED)*, 682–687.

Lemos, J.M., Machado, F.C., Nogueira, N.M., and Shirley, P.O. (2010). Modelo SIMULINK de um canal piloto - Manual do utilizador. *INESC-ID, Relatório Técnico n^{o}35/2010*, (35).

Lemos, J.M. and Pinto, L.F. (2012). Distributed Linear-Quadratic Control of Serially Chained Systems - Application to a Water Delivery Canal. *IEEE Control Systems Magazine*, 32(6), 26–38.

Lemos, J.M. (2009). Distributed Adaptive Predictive Control of Water Delivery Networks. *IFAC Proceedings Volumes*, 42(20), 156–161.

Lemos, J.M., Pinto, L.F., Rato, L.M., and Rijo, M. (2013). Multivariable and Distributed LQG Control of a Water Delivery Canal. *Journal of Irrigation and Drainage Engineering*, 139(10), 855–863.

Malaterre, P. and Baume, J. (1998). Modeling and regulation of irrigation canals: existing applications and ongoing researches. *SMC'98 Conference Proceedings. 1998 IEEE International Conference on Systems, Man, and Cybernetics (Cat. No.98CH36218)*, 4, 3850–3855.

Mareels, I., Weyer, E., Ooi, S.K., Cantoni, M., Li, Y., and Nair, G. (2005). Systems engineering for irrigation systems: Successes and challenges. *Annual Reviews in Control*, 29(2), 191–204.

Mota, J.F.C., Xavier, J.M.F., Aguiar, P.M.Q., and Püschel, M. (2013). D-ADMM: A Communication-Efficient Distributed Algorithm For Separable Optimization. *IEEE Transactions on Signal Processing*, 61(10), 2718–2723.

Nash, J.F. (1950). Equilibrium Points in N-Person Games. *Mathematics*, 36, 1949–1950.

Negenborn, R.R., Overloop, P.J., Keviczky, T., and Schutter, B. (2009). Distributed model predictive control of irrigation canals. *Networks and Heterogeneous Media*, 4(2), 359–380.

Pinto, L.M. (2011). *Distributed LQG control of a Water Delivery Canal*. Master's thesis, Instituto Superior Técnico, Lisboa.

Rawlings, J.B. and Mayne, D.Q. (2009). *Model Predictive Control : Theory and Design*. Nob Hill Publishing.

Rijo, M., Rato, L., Nogueira, N., Machado, F., and Lemos, J. (2009). Adaptive and non-adaptive model predictive control of an irrigation channel. *Networks and Heterogeneous Media*, 4(2), 303–324.

Sampaio, I. (2012). *Fault Tolerant Control of a Water Delivery Canal*. Master's thesis, Instituto Superior Técnico, Lisboa.

Sanoff, S.P. and Wellstead, P.E. (1983). Comments on: 'implementation of self-tuning regulators with variable forgetting factors'. *Automatica*, 19(3), 345–346.