Actuator Fault Tolerant LQG Control of a Water Delivery Canal*

João M. Lemos¹, Inês Sampaio², Manuel Rijo³ and Luís M. Rato⁴

Abstract— The problem of reconfiguration of the control system to mitigate the effects of actuator faults in a water delivery canal is addressed in this paper. When a fault in an actuator is detected and isolated, the controller is reconfigured by changing the set of manipulated and process variables and using a different controller, associated to a different plant model, in a hybrid systems framework. In order to prevent instability that may be associated with switching among controllers, a dwell time condition is used. Both centralized and distributed LQG controllers are considered. In the case of distributed control, a game approach is followed to coordinate the different local controllers. Experimental results are presented.

I. INTRODUCTION

A. Motivation

Water delivery open canals used for irrigation [1] are large structures whose complexity, together with increasing requirements on reliability and quality of service provides a strong motivation to consider fault tolerant control methods [2]. In order to achieve fault tolerant features, the idea consists in exploring the redundancy in installed sensors and actuators to reconfigure the control system such as to allow the plant operation to continue, perhaps with some graceful degradation, when a sensor or actuator fails. Fault tolerance may be embedded either in centralized multivariable controllers, where a single controller receives the data from all the sensors and uses it to compute the value of all manipulated variables, or in distributed controller networks, where local controllers, each connected to a single gate, negotiate their moves with their neighbors in order to reach a consensus that allows coordinated action among them.

In general, distributed control is useful for water delivery canal since these are plants that may extend over wide areas, with the actuators (gates) separated by long distances, over which data communication systems may be unreliable. The use of distributed control has the advantage of already providing per se a certain degree of fault tolerance. If a local controller fails, the others can still ensure their own tasks. However, since the resulting interconnections may yield an unstable overall system, the fault tolerant mechanisms must ensure both a reconfiguration of the communication network and a redesign of the local controllers such as to keep the overall system stable. In [3], [4] a distributed LQG algorithm based on a game approach has been presented and compared to multivariable LQG control. In this paper we extend these algorithms to make them tolerant to faults in actuators.

B. Literature review

The concept of fault tolerant control (FTC) has been the subject of intense research in the last twenty years [5], [6], [7], in particular in what concerns reconfigurable fault tolerant control systems [8]. This activity yielded a rich bibliography that, of course, cannot be covered here and that comprises aspects such as fault detection and isolation and fault tolerant control design. In relation to distributed control, an important concept is “integrity”, namely the capacity of the system to continue in operation when some part of it fails [9]. Other type of approach models the failures as disturbances that are estimated and compensated by the controller [10]. In what concerns water delivery canal systems a topic that receives attention due to their immediate economic impact related to water saving is leak detection [11]. Other aspects found in the literature are control loop monitoring [12], and reconfiguration to mitigate fault effects [14], which is the issue considered in this work. Reconfiguring the controller in face of a plant fault falls in the realm of hybrid systems and raises issues related to stability that must be taken into account [13].

C. Contributions and paper structure

The contribution of this paper consists of the application of LQG centralized and distributed fault tolerant control to a water delivery canal in the presence of actuator faults. An algorithm based on controller reconfiguration with a dwell time logic is presented, together with a sufficient condition on the dwell time to ensure stability and experimental results. The paper is organized as follows: After the introduction in which the work is motivated, a short literature review is made and the main contributions are presented, the canal is described in section II, including a static nonlinearity compensation of the gate model. Centralized LQG control is described in section III and distributed LQG control is described in section IV, whereas actuator fault tolerant control is dwelt with in section V. Experimental results are presented in section VI. Finally, section VII draws conclusions.

II. THE CANAL SYSTEM

A. Canal description

The experimental work reported hereafter was performed at the large scale pilot canal of Núcleo de Hidráulica

— The problem of reconfiguration of the control system to mitigate the effects of actuator faults in a water delivery canal is addressed in this paper. When a fault in an actuator is detected and isolated, the controller is reconfigured by changing the set of manipulated and process variables and using a different controller, associated to a different plant model, in a hybrid systems framework. In order to prevent instability that may be associated with switching among controllers, a dwell time condition is used. Both centralized and distributed LQG controllers are considered. In the case of distributed control, a game approach is followed to coordinate the different local controllers. Experimental results are presented.
described in [4]. The canal has four pools with a length of 35m, separated by three undershoot gates, with the last pool ended by an overshoot gate. In this work, only the first three gates are used. The maximum nominal design flow is 0.09 m³s⁻¹. There are water off-takes downstream from each branch made of orifices in the canal walls, that are used to generate disturbances corresponding to water usage.

Water level sensors are installed downstream of each pool. The water level sensors allow to measure values between 0 mm and 900 mm, a value that corresponds to the canal bank. For pool number $i$, $i = 1, \ldots, 4$, the downstream level is denoted $y_i$ and the opening of gate $i$ is denoted $u_i$. The nomenclature is such that pool number $i$ ends with gate number $i$.

Each of the actual gate positions $u_{r,i}$, $i = 1, 2, 3$, is manipulated by a command signal $u_i$. However, the PLCs that command gate motors are programmed such that $u_{r,i}$ only moves in response to $u_i$ if $|u_i - u_{r,i}| \geq 0.5 \text{ mm}$. This dead zone nonlinearity has two types of implications. First of all, it limits the controller achievable precision when tracking a reference and it can even induce small amplitude oscillations [4]. Furthermore, when comparing the signals $u_i$ and $u_{r,i}$ in order to detect a fault, this difference must be taken into consideration.

### B. Nonlinearity compensation

Following [2], in order to compensate for a nonlinearity, instead of using as manipulated variables the gate positions $u_{r,i}$, the corresponding water flows $v_i$ crossing the gates are used. These are related by

$$v_i = C_{ds} W u_{r,i} \sqrt{2g(h_{upstr,i} - h_{downstr,i})},$$

where $C_{ds}$ is the discharge coefficient, $W$ is the gate width, $g = 9.8 \text{m/s}²$ is the gravity acceleration, $h_{upstr,i}$ is the water level immediately upstream of the gate and $h_{downstr,i}$ is the water level immediately downstream of the gate. This approach corresponds to representing the canal by a Hammerstein model and to compensating the input nonlinearity using its inverse. The linear controller computes the flow crossing the gates, that is considered to be a virtual command variable $v_i$ and the corresponding gate position is then computed using (1). The discharge coefficient is not estimated separately, but instead is considered to be incorporated in the static gain of the linear plant model.

### C. Canal model

In order to design the controllers, the dynamics of the canal has been approximated by a finite dimension linear state-space model written as

$$x(k + 1) = Ax(k) + Bu(k),$$

$$y(k) = Cx(k)$$

where $k \in \mathbb{N}$ denotes discrete time, $x \in \mathbb{R}^n$ is the full canal state, $y \in \mathbb{R}^p$ is the output made of the downstream pool levels, with $p$ the number of outputs, $v \in \mathbb{R}^p$ is the manipulated variable and $A \in \mathbb{R}^{n \times n}$, $B \in \mathbb{R}^{n \times p}$ and $C \in \mathbb{R}^{p \times n}$ are matrices. In the case of $p = 3$ (three pools), and assuming operation around a constant equilibrium point, these matrices are identified by constraining the model to have the following structure

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

$$C = \begin{bmatrix} C_1 & 0 & 0 \\ 0 & C_2 & 0 \\ 0 & 0 & C_3 \end{bmatrix}.$$ (4)

These matrices have dimensions that match the state $x_i$ associated to each pool such that $x = [x_1' \ x_2' \ x_3']'$. This structure is imposed to reflect the decomposition of the canal model in subsystems, each associated to a different pool. Furthermore, it is assumed that each pool interacts directly only with its immediate (upstream and downstream) neighbors, and only through the input.

### III. CENTRALIZED LQG CONTROL

As shown in figure 1 in normal (no fault) operation, the centralized LQG control considers the control problem as a multivariable one. The controller computes in a centralized way all the manipulated variables, using feedback data from all the process outputs.

For the purpose of designing a centralized LQG regulator that ensures steady-state tracking of a constant reference, augment the plant (2) with an integrator, resulting in the augmented model

$$\ddot{x}(k + 1) = \bar{A}\ddot{x}(k) + \bar{B}v(k),$$ (5)

in which

$$\ddot{x} = \begin{bmatrix} x \\ \dot{x} \end{bmatrix}, \quad \bar{A} = \begin{bmatrix} A & 0 \\ 0 & T_s C \end{bmatrix}, \quad \bar{B} = \begin{bmatrix} B \\ 0 \end{bmatrix},$$

where $I[p \times p]$ stands for the identity matrix and $T_s$ is the sample time. The output of the augmented process is

$$y(k) = \bar{C}\ddot{x}(k),$$ (6)
with \( \bar{C} = [C \ 0] \).

For this augmented process design a LQ regulator as

\[
\begin{equation}
\begin{bmatrix}
    v(k) \\
    x(k) \\
    x_1(k)
\end{bmatrix} = - \begin{bmatrix}
    K_x & K_f \end{bmatrix} \begin{bmatrix}
    x(k) \\
    x_1(k)
\end{bmatrix},
\end{equation}
\]

where

\[
\begin{bmatrix}
    K_x & K_f
\end{bmatrix} = (I + R^{-1} B^\top \bar{S} B)^{-1} R^{-1} B^\top \bar{S} A,
\]

with

\[
\bar{S} = \bar{A}^\top \bar{S} + \bar{B} R^{-1} B^\top \bar{S}^{-1} \bar{A} + \bar{Q}.
\]

Since the augmented system is non-observable, and the state of the integrator is available for direct measurement, only the state of (2) is estimated using a Kalman filter. It is possible to prove that a separation principle holds for this case.

IV. DISTRIBUTED LQG CONTROL

A. Local controller network

For the purpose of designing a network of distributed LQG controllers, decompose the canal in subsystems, each composed of a pool and the downstream pool. As shown in figure 2, associate a local LQG controller to each subsystem. Local controllers negotiate with their neighbors in order to coordinate their control moves. It is assumed that controller \( i \), associated with pool \( i \), has access to the data of pools \( i+1 \) and \( i-1 \) (whenever they both exist) and negotiates only with them.

From the global multivariable model (2, 3), each pool \( i \) is represented by the state model with accessible disturbance \( d_i \)

\[
x_i(k+1) = A_i x_i(k) + B_i v_i(k) + d_i(k)
\]

where

\[
d_1(k) = B_{12} v_2(k),
\]

\[
d_2(k) = B_{21} v_1(k) + B_{23} v_3(k)
\]

and

\[
d_3(k) = B_{32} v_2(k).
\]

The manipulated variable of each local controller \( i \) is computed by

\[
v_i(k) = -K_{x,i} x_i(k) - K_{f,i} x_1(k) + K_{ff,i} d_i(k),
\]

where

\[
K_i = [K_{x,i} \ K_{f,i}]
\]

and \( K_{ff,i} \) are obtained by solving a LQ problem that consists of minimizing the steady state quadratic cost

\[
J_i = \sum_{k=1}^{\infty} x_i^T(k) Q_i x_i(k) + u_i^T(k) R_i u_i(k),
\]

with

\[
Q_i = \begin{bmatrix}
    C_i^T C_i & 0 \\
    0 & 1
\end{bmatrix},
\]

and \( \rho_i > 0 \) a design parameter.

B. Coordination algorithm

When using distributed control, each gate is manipulated by a SISO controller that selects its moves so as to drive the corresponding water level to the reference value. In addition, there is a correction to achieve a coordinated action. The coordination among controllers is performed by the following algorithm:

\[
\text{Coordination algorithm}
\]

At the beginning of each sample time, compute \( v_{i,0}(k) \) by solving a LQG problem associated to model (10) and assuming \( d_i(k) = 0 \).

For \( j = 1 \) up to \( j = N_c \) recursively perform the following cycle

1) For \( i = 1, 2, 3 \), compute \( d_{i,j-1}(k) \) using (11-13) with \( v_i(k) \) replaced by \( v_{i,j-1}(k) \);

2) For \( i = 1, 2, 3 \), compute \( v_{i,j}(k) \) by solving a LQG problem associated to model (10);

Apply to the plant the control given by

\[
v_i(k) = v_{i,j}(k)
\]

V. ACTUATOR FAULT TOLERANT CONTROL

A. Controller reconfiguration

Figure 3 shows a discrete state diagram that explains how controller reconfiguration is performed when an actuator fault occurs in the water channel considered in this paper. For simplicity, only the occurrence of faults in gate 2 are considered. State \( S_1 \) corresponds to the situation in which all gates are working normally with a controller \( C_N \) that matches this situation. When a fault occurs, the system state switches to \( S_2 \), in which gate 2 is faulty (blocked) but the controller used is still the one designed for the no fault situation.

In the presence of a fault, the matrices of the state-space model (3) have the structure

\[
A = \begin{bmatrix}
    A^F_{11} & 0 \\
    0 & A^F_{33}
\end{bmatrix},
\]

\[
B = \begin{bmatrix}
    B^F_{11} & B^F_{12} & B^F_{13} \\
    B^F_{31} & B^F_{32} & B^F_{33}
\end{bmatrix},
\]

\[
C = \begin{bmatrix}
    C^F & 0 \\
    0 & C^F_{33}
\end{bmatrix}.
\]
The superscript $F$ enhances the fact that the matrix blocks are estimated assuming that a fault has occurred and that they are different from the ones in (4).

It is remarked that the matrices in (19) are re-estimated with data obtained under the faulty condition so that the new model (labeled "F") matches it. The dimension of the local states does not even have to be the same as in the model for the normal condition. Furthermore, one has to bear in mind that the state models do not derive from physical laws but, instead, as estimated by fitting them to data, implying thus that there are no constraints related to state dimension other than obtaining a good fit.

Figure 4 shows the controller to apply under a faulty situation for centralized control, whereas figure 5 shows the controller to be used in the same situation for distributed control. In the distributed case, controller reconfiguration implies a reconfiguration of the communication network as well. When the fault is detected, the state switches to $S_3$, in which a controller $C_F$ designed for the faulty situation is connected to the canal. When the fault is recovered (gate 2 returns to normal operation), the state returns to $S_1$. A dwell time condition is imposed to avoid instability that might arise due to fast switching [15]. This means that, once a controller is applied to the plant, it will remain so for at least a minimum time period (called dwell time).

When distributed control is used, the controller designed for normal operation (shown in figure 2), $C_N$, consists of 3 SISO LQG controllers $C_1$, $C_2$ and $C_3$, each regulating a pool and such that each individual controller negotiates the control variable with its neighbors. This means that, in states $S_1$ and $S_2$, $C_1$ negotiates with $C_2$, $C_2$ negotiates with $C_1$ and $C_3$ and $C_3$ negotiates with $C_2$. The controller for the faulty condition (shown in figure 5) is made just of two SISO controllers that control pools 1 and 3 and negotiate with each other.

B. Bumpless transfer

A bumpless transfer algorithm is used in order to ensure the continuity of the manipulated variable command applied to the gates when there are switching among controllers. A way to force the output of the controller $C_F$ to be the same as that of $C_N$ consists in initializing the integrator of the controller $C_F$ such as to compensate the difference between the present gate position and the contribution to control given by state feedback. By solving the equation that yields the manipulated variable as a linear combination of the state with respect to $x_I(k)$, the integrator state is obtained as a function of the actual gate discharge, $v_I(k)$, and the state feedback

$$x^F_I(k) = -K^F_I^{-1}[v_I(k) + K^F_I x^F_I(k)] .$$

If we feedback the real gate position values to the integrator of the fault controller, $C_F$, the integrator state of the controller to be used in the faulty situation can be computed as the following piecewise-defined equation

$$x^F_I(k) = \begin{cases} 
-K^F_I^{-1}[v_I(k) + K^F_I x^F_I(k)] & \text{if } k < k_d, \\
-x^F_I(k-1) + T_e e(k-1) & \text{if } k \geq k_d,
\end{cases}$$

where $k_d$ is the time at which the fault is detected and the superscript $F$ enhances the fact that the integrator belongs to controller $C_F$.

C. Fault detection

For actuator faults, the fault detection algorithm operates as shown in the block diagram of figure 6. For each gate $i$,
For this type of plant, reconfigurable LQG controllers have been applied to achieve fault tolerance. The application of fault tolerant control to a water delivery canal is shown in Figure 6.

The main contributions of the paper are concerned with the application of fault tolerant control to a water delivery canal. For this type of plant, reconfigurable LQG controllers have been applied to achieve fault tolerance.

**VI. EXPERIMENTAL RESULTS**

**A. Fault tolerant centralized control**

Figures 7, 8, and 9 refer to fault tolerant centralized control. At the time instant marked by a red vertical line, a fault occurs that forces gate 2 to become stuck. Shortly after, at the instant marked by the yellow vertical line, this fault is detected, and the controller is reconfigured as explained. From this moment on, there is no warranty on the value of the level $J_2$, but $J_1$ and $J_3$ continue to be controlled. The effect of coordination is apparent in the setpoint decrease of pool 1, close to time 5400 s. In response to the decrease in set-point for pool 1 that occurs at this time (figure 10), controller 1 opens gate 1 to release water (see the label “A” in figure 11). As a consequence of the coordination procedure, the controller connected to gate 3 immediately opens gate 3 (see label “B” in figure 11), anticipating the excess incoming water and almost avoiding deviation of the level from the specified reference (figure 10).

**B. Fault tolerant distributed control**

A similar example for fault tolerant distributed control is reported in figures 10, 11, and 12. The performance obtained is similar. The coordination effect in distributed control is apparent in this figures at $t = 5400$ s. In response to the decrease in set-point for pool 1 that occurs at this time (figure 10), controller 1 opens gate 1 to release water (see the label “A” in figure 11). As a consequence of the coordination procedure, the controller connected to gate 3 immediately opens gate 3 (see label “B” in figure 11), anticipating the excess incoming water and almost avoiding deviation of the level from the specified reference (figure 10).

**VII. CONCLUSIONS**

The main contributions of the paper are concerned with the application of fault tolerant control to a water delivery canal. For this type of plant, reconfigurable LQG controllers have been applied to achieve fault tolerance.

---

Fig. 6. Structure of the fault detection system.

Fig. 7. Centralized LQG controller. Reconfiguration after a fault in gate 2. Pool levels.

Fig. 8. Centralized LQG controller. Reconfiguration after a fault in gate 2. Gate positions.

Fig. 9. Centralized LQG controller. Reconfiguration after a fault in gate 2. Fault detection variables.

---

\[ i = 1, 2, 3, \text{define the error } \bar{u}_i \text{ between the command of the gate position } u_i \text{ and the actual gate position } u_{r,i}, \]
\[ \bar{u}_i(k) = u_i(k) - u_{r,i}(k) \]
\[ \Pi(k) = \gamma \Pi(k-1) + (1 - \gamma) |\bar{u}(k)|. \]
been developed and demonstrated experimentally. A comparison between centralized and distributed control, when considering FTC, shows that the distributed algorithm is a good approximation to the multivariable one. Embedding fault tolerance features improves the performance, as measured by the degree with which the reference level is tracked in each pool. A non-standard feature is the use of the gate nonlinear compensation described in section II-B that improves the plant linearity.

The actuator fault detection scheme used takes advantage of the information available for the gate position that reduces the detection algorithm to a comparison of two signals. The extension from fault detection algorithm and distinguish between the centralized and distributed cases. These problems will be addressed elsewhere.

REFERENCES