GoRoSoBo: an overall control diagram to improve the efficiency of water transport systems in real time

Enrique Bonet, Manuel Gómez, M. T. Yubero and J. Fernández-Francos

ABSTRACT

Agriculture plays an important part in the food chain and water resources for agriculture are essential. A problem is that the water transport systems present low efficiencies in practice. Crop yields must be optimized, and the goal of an operational water manager is to deliver water to irrigation sites accurately and efficiently. In order to fulfill this objective, we propose a centralized overall control diagram to optimize the management of the canal. Our control diagram in real time is mainly composed of two algorithms, CSE and GoRoSoBo. The first one is a powerful tool in canal management, and is able to estimate the real extracted flow in the canal and the hydrodynamic canal state from measured level data at selected points. The second one is an essential tool in the management of the canal, a feedback control algorithm operating in real time. The GoRoSoBo algorithm (Gómez, Rodellar, Soler, Bonet) is able to calculate the optimum gates trajectories for a predictive horizon taking into account the current canal state and the real extracted flow, both obtained by CSE.

Key words | agricultural demands, canal control, flow rate extractions, open channel flow, optimization algorithms

Enrique Bonet (corresponding author)
Spanish National Research Council (CSIC),
Institute of Environmental Assessment and Water Research (IDAEA),
Consejo Superior de Investigaciones Científicas (CSIC),
Instituto de Diagnóstico Ambiental y Estudios del Agua (IDAEA),
C/ Jordi Girona, 18-26,
08034 Barcelona, Spain
E-mail: enrique.bonet@upc.edu

Manuel Gómez
Barcelona School of Civil Engineering,
Flumen Institute,
C/ Jordi Girona, 1,
08034 Barcelona, Spain

M. T. Yubero
Department of Civil and Environmental Engineering,
Barcelona School of Civil Engineering,
C/ Jordi Girona, 1,
08034 Barcelona, Spain

J. Fernández-Francos
Hydraulic Engineering Section, Energy Department,
Polytechnic School of Mieres (University of Oviedo),
C/ Gonzalo Gutiérrez Quirós s/n. 33600 Mieres,
Asturias, Spain

INTRODUCTION

The world population is predicted to grow from 6.9 billion in 2010 to 8.3 billion in 2030 and 9.1 billion in 2050 (UNDESA 2009) and food demand is predicted to increase by 50% in 2030 and by 70% in 2050 (Bruinsma 2009).

The most recent estimate for irrigated agriculture is an increase from 2,743 km^3 in 2008 to 3,858 km^3 in 2050 (FAO 2011a, 2011b). Much of the increase in irrigation water will be in regions already suffering from water scarcity. To study the impact of water scarcity, there are several simulation models available, for instance, MCG (general circulation model) or MCR (regional climate model). Most specifically in Spain, some of these models (AR4 (Rossby Centre regional atmospheric model) and HIRLAM (high resolution limited area model)) predict an increase of 1 °C in temperature and a decrease of 5% in precipitation in 2020, and as a result, a decline in water resources from these areas of 10% according to AEMET (2009) (Spanish National Weather Service). These values could increase significantly according to other studies which calculate a decrease of 20% in precipitation and an increase of 3 °C in temperature in 2050 according to the PNACC (2011).

In Figure 1, we show the consequences of water scarcity in two scenarios for several catchments in Spain. The first scenario represents simulations with an increase of 1 °C in temperature, in one of those scenarios without change in precipitation and in the other with a decrease of 5% in precipitation.

Taking into account these estimations, the main challenge facing the agricultural sector is making 70% more
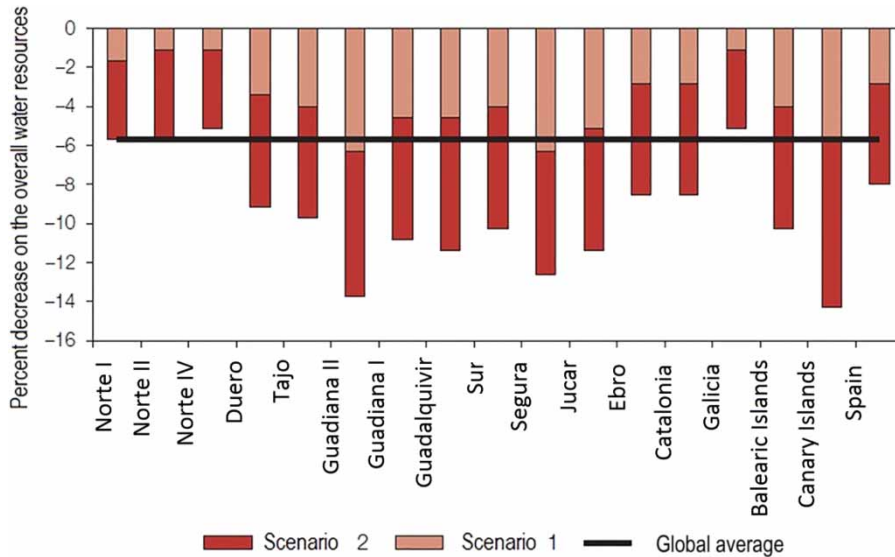
Figure 1 | Evolution of the percent decrease on the overall water resources in Spain for different river basins. Scenarios 1 and 2 represent simulations of a temperature increase of 1 °C, without precipitation change or a decrease of 5%, and the global average line represents the average between Scenarios 1 and 2 (MIMAM 2000).

food available with fewer water resources. More water resources or a more efficient use of the resources will be needed to fit this objective. We cannot waste water resources and therefore we are forced to improve the current irrigation techniques and invest in automating the operation of irrigation canals, for instance, developing new control algorithms.

Many algorithms, in particular open loop algorithms, have difficulties and failures in implementation in real canals (Rogers & Goussard 1998). These difficulties are due to deviations between the predictive/control model and the reality.

The disturbances introduced into the canal lead to important deviations between the model and reality. The disturbances caused by infiltration can be controlled by a good coating; these disturbances are not usually important and they have constant little value. Due to the difficulty of distinguishing between small disturbances and small perturbations caused by conceptual errors of the model or water infiltration losses, they are all considered negligible in canal control. The disturbances introduced by changes in demand deliveries are more dangerous because the canal could overflow or could dry depending on the magnitude of the disturbance. These disturbances are usually unscheduled demand deliveries which are caused by flow rate extractions during the irrigation cycle.

In some cases, the farmers can adjust the supplies obtained by the irrigation community before the irrigation cycle begins; these changes in demand deliveries are known. Thus, these disturbances are known in advance and can be mitigated by an open loop controller (feedforward controller) such as GoRoSo (Bautista *et al.* 2003; Wahlin & Clemmens 2006; Soler *et al.* 2013).

The main problems in a canal are the disturbances caused by climatic variations (rainfalls and associated runoff) or unscheduled demands by farmers that extract more or less flow demanded by them, because these disturbances are more difficult to mitigate by a controller. In such cases, the CSE algorithm (Bonet 2015; Soler *et al.* 2015; Bonet *et al.* 2017) is an excellent tool to approach the real extractions in the canal in real time.

One way to protect the canal from these disturbances could be with ponds built by the irrigation community or the farmers themselves. A reservoir is able to store water according to the crop requirements and regulates the volume of water provided by the canal, but it is not always possible to build large reservoirs to regulate each pool of the canal. Another option to control the disturbances in canals, which operate in steady state, would be to use the wedge storage of every pool, but this only works with very low disturbances. In all other cases, a closed loop controller (feedback controller) is needed to modify the sluice gate

trajectory to return to the desired canal state in real time, which is an easy way for our overall control diagram.

Beyond the disturbances, there are factors that significantly affect the canal, such as changes in Manning's coefficient which the on-line predictive controller needs to know to recalculate the optimal sluice gate trajectory. For this reason, we define a set of algorithms that operate all together in parallel but not all of them necessarily operate in real time. All the information obtained by these algorithms is used to supply the on-line predictive controller (GoRoSoBo) to meet its objective.

In this paper, our overall control diagram has been tested in several numerical examples. The control diagram was tested in a canal with a single pool; the geometry of the canal is based on Bautista's work (Baustista & Clemmens 1997). This kind of canal has been proposed by different authors such as Wylie (1969), Liu *et al.* (1992), Chevereau (1991), and Soler (2003) to test their control algorithms due to the ease of checking the results. In a second example, the overall control diagram was tested with the test cases (Clemmens *et al.* 1998) introduced by the ASCE Task Committee on Canal Automation Algorithms according to specific guidelines established to test canal controllers.

DIAGRAM

As mentioned previously, the main objective is to develop an overall control diagram able to operate the canal in real time, rectifying the gate trajectory in case of disturbances, and re-establishing the desired behavior of the canal by the watermaster. In our opinion, an on-line predictive controller, operating alone and only using as input data the water levels measured at the checkpoints, is not an accurate predictive controller. For this reason, we define a set of algorithms that operate all together in parallel, but not all of them necessarily operating in real time, as the period of time that each one is working depends on the variability of the process in time. Although there are multiple forms to define an overall scheme for canal control, we show our ideal solution of an overall control diagram of a canal in Figure 2. This figure will be commented on in several parts of the paper because the algorithms developed

follow the structure of this diagram. Every process as well as the task developed for each algorithm is introduced as follows.

(1) Crop needs and desired hydrographs for canal outlets: The hydrographs at the lateral diversion points of the main canal are calculated on the basis of the water demands. They are fixed considering the farmers' requirements and other demands accepted by the watermaster. The behavior of the canal supplying these hydrographs determines the 'desired behavior' (Y^*) at several cross sections.

(2) Off-line computation of the reference trajectories: The desired behavior (Y^*) must be transmitted to the 'reference trajectories calculation' algorithm that determines the positions of each gate. This algorithm calculates the optimum behavior (Y_R (reference water level)), which is the one most similar to the desired behavior that is physically possible. We call 'U_R' the optimum gate trajectories calculated to obtain the optimum behavior (Y_R). They must be calculated off-line (e.g., with an anticipated irrigation cycle). There is an extensive bibliography of feedforward control algorithms.

(3) Off-line parameter identification: This in case of variations of empirical parameters such as Manning's coefficient or gate discharge coefficients. The water level measurements obtained during a previous period of the irrigation cycle in steady state can be transmitted to the 'off-line parameter identification' algorithm for estimating these empirical parameters (Figure 2). The values of these parameters can be sent to the 'off-line reference trajectories computation' and the 'on-line current state computation' to be more accurate in their respective processes.

(4) On-line current state computation: This module uses all values stored in the database such as 'u' (gate position) and y_M (measured water level) which have been measured at the checkpoints in the canal during a past time horizon. If the water level measured at checkpoints is different from the desired water level, it is probably due to a disturbance introduced into the system or erroneous roughness coefficients used in the calculation. That is why it is very important to update Manning's coefficients periodically. The algorithm developed in this module is
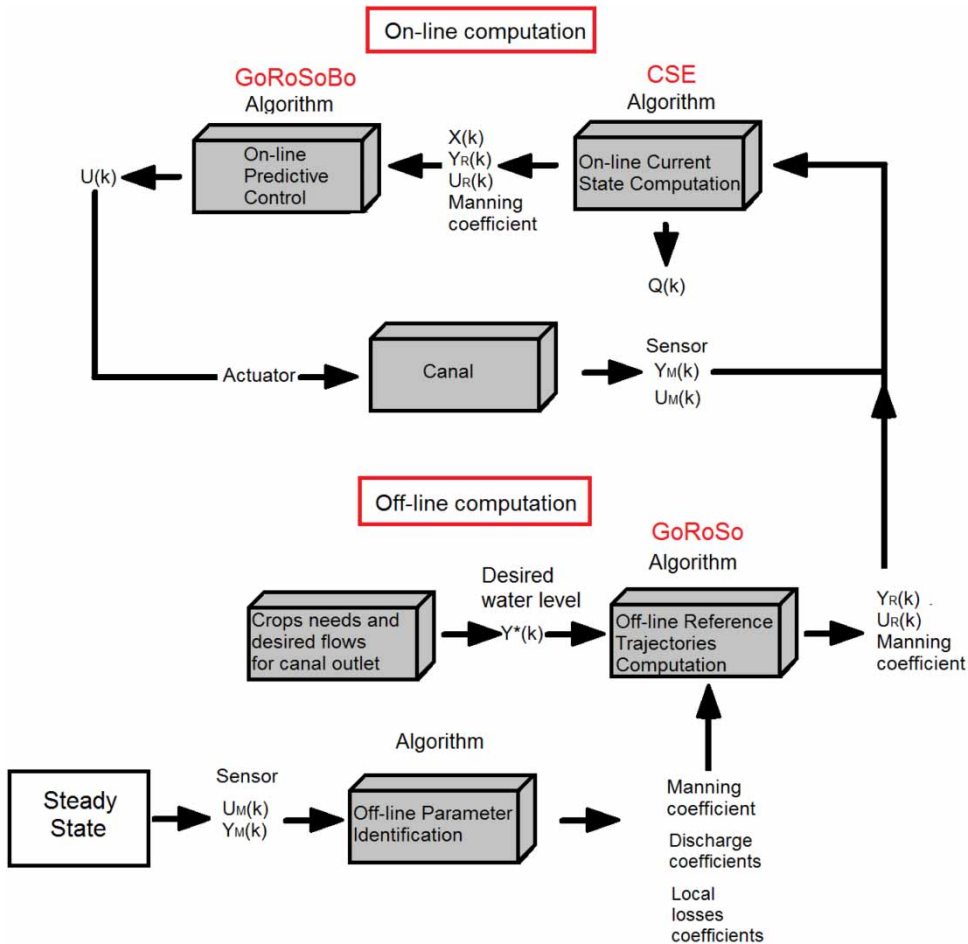
Figure 2 │ Overall control diagram of irrigation canal.

the CSE algorithm (Bonet *et al.* 2017) which is able to estimate the disturbances that have modified the measured water level at checkpoints from the desired water levels. The algorithm can also obtain the current hydrodynamic state of the canal, which is very useful in any case. The hydrodynamic state of a canal is defined as the velocity and water level at each cross section of the canal.

(5) On-line predictive control: The reference trajectories plus the disturbances and the hydrodynamic canal state in real time are transmitted to the next algorithm, called 'on-line predictive control', which must react on-line in case of deviations between the observed data and the desired behavior at the checkpoints during each regulation period (e.g., every 5 min), due to unknown perturbations. The predictive control recalculates new gate positions (U) to return to the reference

behavior. There is an extensive bibliography of feedback control algorithms which recalculate the gate trajectories such as CARA (Marzouki 1989; Clemmens & Wahlin 2004; Wahlin & Clemmens 2006).

In this paper, we focus on the on-line computation task of our overall control diagram, because we are interested in improving the efficiency in water transport in 'real time'.

The CSE solves an inverse problem implemented as a nonlinear optimization problem using the Levenberg–Marquardt method, in which the solution is the flow disturbances obtained from the water level variations measured at several points in the canal, usually next to the canal offtakes.

The GoRoSoBo solves an inverse problem (Equation (1)) implemented as a constrained nonlinear optimization

problem using the Lagrange–Newton method, where the solution is the gate trajectories obtained from the output data partly from the CSE algorithm. In that sense, from the real extracted flow and the current hydrodynamic state, it is easy to obtain the water level disturbances at several points of the canal during a future horizon. Any deviation from the reference is fed back into the on-line predictive control so that this reduces the deviation of the controlled quantity from the reference. There are several canal control algorithms in the literature which use optimization methods to get the optimal gate positions to reach a particular water level target, for example, Sanders & Katopodes (1999).

The development of the CSE algorithm is beyond the scope of this paper and has been shown several times. It can be found in Bonet *et al.* (2017). Instead, the development of GoRoSoBo algorithm is shown below:

$$\Delta Y = [\text{HIM}'(U)]\Delta U$$
$$\Delta U = [\text{HIM}'(U)]^{-1}\Delta Y$$
$$[\text{HIM}'(U)] = \frac{\partial Y}{\partial U} \tag{1}$$
$$[\text{HIM}(U)] = \left(\frac{\partial Y}{\partial U}, \frac{\partial V}{\partial U}\right)$$

where ΔY represents the changes in water level at selected points of the canal, ΔU represents a change in gate position, HIM'(U) is the simplified hydraulic influence matrix that represents the influence of a gate position on the water level at different points of the canal, and HIM(U) is the hydraulic influence matrix that represents the influence of a gate position on the water level and velocity along the canal.

The algorithm establishes a constrained nonlinear optimization problem to solve the last expression using the Lagrange–Newton method, as previously mentioned.

The HIM matrix

The HIM matrix defines the influence of any gate opening over the hydraulic behavior of canal cross sections, usually limited to checkpoint sections. It is based on the full Saint-Venant equations, which describe the free surface flow in canals. In partial derivatives this system is of the hyperbolic,

nonlinear, and second-order type. The two equations are based on the mass and momentum conservation.

Like any hyperbolic system, it can be transformed into its characteristic form. Such transformation of the Saint-Venant equations gives an ordinary system of four equations:

$$\left. \begin{array}{l} \dfrac{dv}{dt} + \dfrac{g}{c(y)}\dfrac{dy}{dt} = g[S_0 - S_f(y, v)] \\[2mm] \dfrac{dv}{dt} - \dfrac{g}{c(y)}\dfrac{dy}{dt} = g[S_0 - S_f(y, v)] \\[2mm] \dfrac{dx^+}{dt} = v + c(y) \\[2mm] \dfrac{dx^-}{dt} = v - c(y) \end{array} \right\} \text{ where } c(y) = \sqrt{\dfrac{gA(y)}{T(y)}} \tag{2}$$

where y is the level of the free surface with reference to the canal bottom, v is the weighted average velocity of all the particles in a canal cross section, t is the time, S_o is the canal bottom slope, $S_f(y, v)$ is the friction slope and c is the celerity of a gravity wave, where $A(y)$ is the area of the wetted surface or a cross section of the flow and $T(y)$ is the top width of the free surface.

$$\left. \begin{array}{l} \dfrac{v_R - v_P}{t_R - t_P} + \left[\theta\dfrac{g}{c_R} + (1-\theta)\dfrac{g}{c_P}\right]\dfrac{y_R - y_P}{t_R - t_P} = gS_0 - [\theta S_{f_R} + (1-\theta)S_{f_P}] \\[3mm] \dfrac{x_R - x_P}{t_R - t_P} = \theta[v_R + c_R] + (1-\theta)[v_P + c_P] \\[3mm] \dfrac{v_R - v_Q}{t_R - t_Q} - \left[\theta\dfrac{g}{c_R} + (1-\theta)\dfrac{g}{c_Q}\right]\dfrac{y_R - y_Q}{t_R - t_Q} = gS_0 - [\theta S_{f_R} + (1-\theta)S_{f_Q}] \\[3mm] \dfrac{x_R - x_Q}{t_R - t_Q} = \theta[v_R - c_R] + (1-\theta)[v_Q - c_Q] \end{array} \right\} \tag{3}$$

The system (Equation (2)) has no known analytical solution, so the use of numerical techniques is necessary. There are many methods that can be used. In order to have the longest possible integration time period with a minimum loss of accuracy, we have adopted a discretization with second-order finite differences, known as 'the method of characteristic curves' (Gómez 1988). If this method is applied to Equation (2) and we take into account the characteristic curves that contain the points P-R and Q-R (Figure 3), respectively, we will obtain the next equations, where $S_{fR} = Sf(y_R, v_R)$, $S_{fP} = Sf(y_P, v_P)$, $S_{fQ} = Sf(y_Q, v_Q)$, and
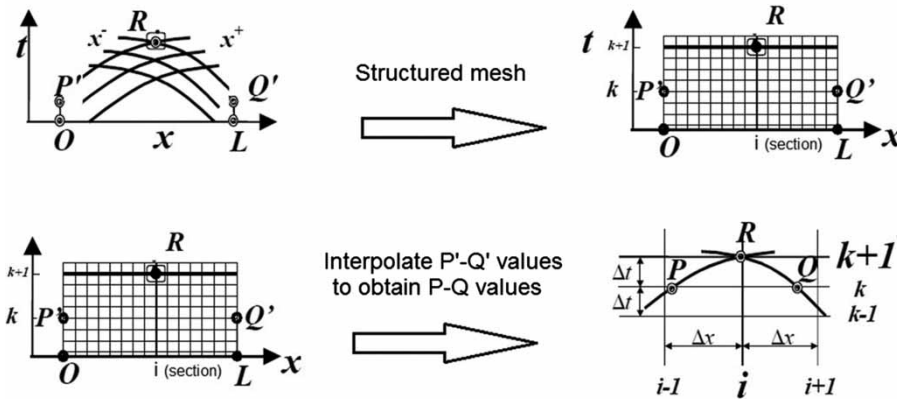
Figure 3 │ The steps for the interpolation onto a structured grid.

$0 \leq \theta \leq 1$ is the coefficient of average time that indicates the type of numerical scheme used. When $\theta = 1$ the numerical scheme is implicit, if $\theta = 0$ is explicit, and when $\theta = 1/2$ the numerical scheme is in central differences or 'method of the characteristic curves'.

If the flow conditions at points P and Q are known, x_P, t_P, y_P and v_P and x_Q, t_Q, y_Q, v_Q are also known, so x_R, t_R, y_R, v_R remain as unknown variables, which can be found by calculating the last four equations (Equation (3)) by using any of the methods solving non-linear equations, such as the Newton–Raphson method.

The way of calculating the influences shown in this section is closely linked to the numerical scheme of characteristic curves. However, usually this scheme is not exactly used because it gives the solution at a point R whose coordinates (x_R, t_R) are unknown a priori. These coordinates are part of the solution and normally it is more important to know the solution of the flow conditions at specific points and at specific time instants. To solve this problem, first interpolate and then solve (Figure 3).

A structured grid like this one (Figure 2) creates a new nomenclature. Indeed, every variable will have a double index, where k refers to time and i to space. Thus, y_{ik} and v_{ik} represent the values for water level and average velocity at the co-ordinates $x_i = i\Delta x$ and $t_k = k\Delta t$ where Δx and Δt are selected by the user.

Obviously, the same system (Equation (3)) is solved, but now with the new unknowns, x_P, y_R, v_R, and x_Q, where the values of y_p and y_Q are dependent on the value of x_P and x_Q

evaluated using an interpolation function of second order too (to be coherent with the numerical procedure used) we have used the Lagrange factors (a way of representing quadratic splines). For a dummy variable z the result is:

$$
\begin{aligned}
z(x) &= s^k(x, z_{i-1}^k, z_i^k, z_{i+1}^k) \\
&= \left(\frac{x - x_i}{\Delta x}\right)\left(\frac{x - x_{i-1}}{2\Delta x}\right) z_{i+1}^k \\
&\quad + \left(\frac{x - x_{i-1}}{\Delta x}\right)\left(\frac{x - x_{i+1}}{-\Delta x}\right) z_i^k \\
&\quad + \left(\frac{x - x_i}{-\Delta x}\right)\left(\frac{x - x_{i+1}}{-2\Delta x}\right) z_{i-1}^k
\end{aligned}
\tag{4}
$$

In this way the variables y_P, v_P, y_Q, and v_Q become functions of x_P and x_Q, as follows:

$$
\begin{aligned}
y_P(x_P) &= s^k(x_P, y_{i-1}^k, y_i^k, y_{i+1}^k) \\
v_P(x_P) &= s^k(x_P, v_{i-1}^k, v_i^k, v_{i+1}^k) \\
y_Q(x_Q) &= s^k(x_Q, y_{i-1}^k, y_i^k, y_{i+1}^k) \\
v_Q(x_Q) &= s^k(x_Q, v_{i-1}^k, v_i^k, v_{i+1}^k)
\end{aligned}
\tag{5}
$$

On the other hand, there are many control structures in canals. The individual study of each one is impossible in this work, so for this reason we will introduce the most usual structures. A common one found is a checkpoint (Figure 4), a target point where the water level is measured with a depth gage, and it includes a sluice gate, a lateral weir outlet, off-take orifice or a pump (shown in Bonet (2015) and Bonet *et al.* (2017)). The interaction of this control structure with
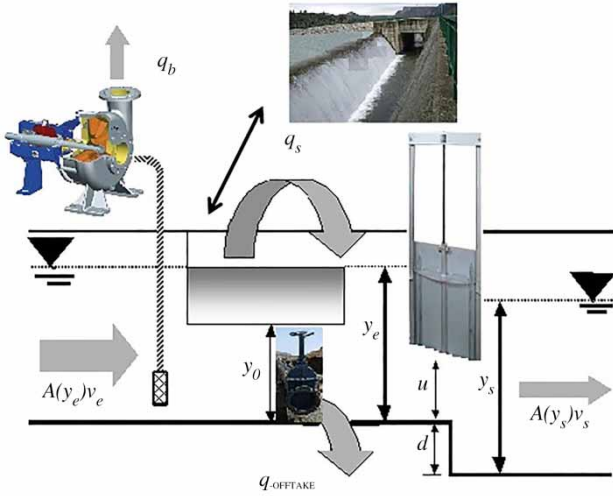
Figure 4 | Diagram of a checkpoint with gate, lateral weir, and pump.

the flow can be described according to the mass and energy conservation equations (Equation (6)).

$$S(y_i)\frac{dy_i}{dt} = A(y_i)v_i - q_b - q_s(y_i) - A(y_o)v_o - q_{\text{offtake}}(y_i)$$
$$* A(y_o)v_o = k_c u \sqrt{y_i - y_o + d}$$
$$q_s(y_i) = C_S a_S (y_i - y_0)^{3/2} \qquad (6)$$
$$q_{\text{offtake}}(y_i) = C_0 A_0 \sqrt{2 g y_i}$$
$$k_c = \sqrt{2 g C_d a_c}$$

where $S(y_i)$ is the horizontal surface of the reception area in the checkpoint. $A(y_i)*v_i$ is the incoming flow to checkpoint, defined in terms of water level and velocity. $A(y_o)*v_o$ is the outflow from the checkpoint which continues along the canal, described in terms of water level and velocity. C_d is the discharge coefficient of the sluice gate and a_c is the sluice gate width. d is the checkpoint drop, and u is the gate opening. q_b is the pumping offtake. $q_s(y_i)$ is the outgoing lateral flow through the weir where C_s is the discharge coefficient, as is the weir width and y_o is the weir height measured from the bottom, called weir equation. $Q_{\text{offtake}}(y_i)$ is the outflow orifice flow where C_O is the discharge coefficient, A_O is the area of the offtake orifice, called orifice offtake equation. *The reverse flow is not taken into consideration for the GoRoSoBo algorithm.

The presence of checkpoints or control structures in the canal leads to the sub-division into canal pools, in a way that there is always a canal pool between two checkpoints, and there is a checkpoint between two pools. If we discretize the control structure in a structured grid, linked with the characteristics of Equation (3) and then change the nomenclature, we should rewrite the control structure Equation (6) arriving at the following system of six equations (Equation (7)).

$$f_1 \equiv x_n - x_P - \frac{1}{2}\Delta t[v_n^{k+1} + c_n^{k+1} + v_P + c_P] = 0$$
$$f_2 \equiv (v_n^{k+1} - v_P) + \frac{g}{2}\frac{c_n^{k+1} + c_P}{c_n^{k+1} c_P}(y_n^{k+1} - y_P) - g\Delta t \left(\frac{S_{fn}^{k+1} S_{f_P}}{2} - S_0\right) = 0$$
$$f_3 \equiv (v_1^{k+1} - v_Q) - \frac{g}{2}\frac{c_1^{k+1} + c_Q}{c_1^{k+1} c_Q}(y_1^{k+1} - y_Q) - g\Delta t \left(\frac{S_{f1}^{k+1} + S_{f_Q}}{2} - S_0\right) = 0$$
$$f_4 \equiv x_1^{k+1} - x_Q - \frac{1}{2}\Delta t \left[v_1^{k+1} - c_1^{k+1} + v_Q - c_Q\right] = 0$$
$$f_5 \equiv A(y_n^{k+1})v_n^{k+1} - q_b - q_s(y_n^{k+1}) - A(y_1^{k+1})v_1^{k+1} - q_{\text{offtake}}(y_n^{k+1}) = 0$$
$$f_6 \equiv A(y_1^{k+1})v_1^{k+1} - k_c u \sqrt{y_n^{k+1} - y_1^{k+1} + d} = 0$$

$$(7)$$

Thus, y_n^{k+1} represents the water level at node n in the section upstream of the control structure at time $k+1$, that is, the incoming water level y_i. In the same way, y_1^{k+1} is defined as the existing water level at the first node of the downstream pool from the checkpoint at the same time $k+1$, and y_o the outgoing water level at the control structure. The same can be said for the velocities v_n^{k+1} and v_1^{k+1} where

$$\Delta t = t^{k+1} - t^P = t^{k+1} - t^Q$$
$$y_P(x_P) = s(x_P, y_{n-2}^k, y_{n-1}^k, y_n^k); y_Q(x_Q) = s(x_Q, y_1^k, y_2^k, y_3^k)$$
$$v_P(x_P) = s(v_P, v_{n-2}^k, v_{n-1}^k, v_n^k); v_Q(x_Q) = s(x_Q, v_1^k, v_2^k, v_3^k)$$
$$c_n^{k+1} = c(y_n^{k+1}); c_1^{k+1} = c(y_1^{k+1})$$
$$S_{fn}^{k+1} = S_f(y_n^{k+1}, v_n^{k+1}); S_{f1}^{k+1} = S_f(y_1^{k+1} v_1^{k+1})$$

On the other hand, x_P, y_n^{k+1}, v_n^{k+1}, y_1^{k+1}, v_1^{k+1}, and x_Q are the unknown variables. In order to continue with the calculation of the influences of a general parameter, in our case, this parameter defines the gate position U. Thus, applying the first derivative of the functions (Equation (7)) with respect to the gate position, we obtain the system of Equation (8).

In Equation (8) for the first time, the gate position U appears explicitly in the description. Despite the fact that the specific form of this function is still unknown, Equation (8) shows that the influence of the parameter U on flow conditions at time $k+1$ is the sum of the indirect influence of the conditions at instant k and the direct influence at instant $k+1$ through the term 'L', which represents the variation in the extraction flow.

$$M \frac{\partial}{\partial U} \begin{pmatrix} x_P \\ y_1^{k+1} \\ v_1^{k+1} \\ y_n^{k+1} \\ v_n^{k+1} \\ x_Q \end{pmatrix} = NS \frac{\partial}{\partial U} \begin{pmatrix} y_{n-2}^k \\ v_{n-2}^k \\ y_{n-1}^k \\ v_{n-1}^k \\ y_n^k \\ v_n^k \\ y_1^k \\ v_1^k \\ y_2^k \\ v_2^k \\ y_3^k \\ v_3^k \end{pmatrix} + L \qquad (8a)$$

where:

$$M = \frac{\partial(f_1, f_2, f_3, f_4, f_5, f_6)}{\partial(x_P, y_n^{k+1}, v_n^{k+1}, y_1^{k+1}, v_1^{k+1}, x_Q)}$$

$$N = -\frac{\partial(f_1, f_2, f_3, f_4, f_5, f_6)}{\partial(x_P, y_P, v_P, y_Q, v_Q, x_Q)}$$

$$L = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & \dfrac{\partial f_6}{\partial U} \end{pmatrix}^T$$

$$S = \frac{\partial(x_P, y_P, v_P, y_Q, v_Q, x_Q)}{\partial(x_P, y_{n-2}^k, v_{n-2}^k, y_{n-1}^k, v_{n-1}^k, y_n^k, v_n^k, y_1^k, v_1^k, y_2^k, v_2^k, y_3^k, v_3^k, x_Q)} \qquad (8b)$$

As a summary, the method of characteristics is applied to the Saint-Venant equations in order to obtain algebraic equations to establish a relation between the influence parameter U and the hydrodynamic canal state; all the influences are lumped together in a global matrix, which is referred to as HIM(U). Based on this system of equations, and using the first derivative ($\partial y/\partial U$, $\partial v/\partial U$) on an analytical process, the changes in flow behavior (water level and velocity) can be established due to a change of gate position at a point at a certain time instant.

The optimization problem

The inverse problem (Equation (1)) is formulated as a constrained optimization problem. It is the classical non-linear problem with constraints and the method used to solve it is the Lagrange–Newton method.

To introduce the optimization problem, we have to evaluate some vectors used in the development. As we explained before, the GoRoSoBo algorithm needs as input data the water level target at some points (checkpoints) for a time horizon established by the watermaster. Now, let us consider a vector (desired water level vector), which contains the water level targets at the checkpoints from the time instant 1 to k_F (Equation (9)) whose dimension is n_y, where $n_y = k_F \times n_c$, k_F is the final instant of the future time horizon and n_c is the number of checkpoints. We define this vector as:

$$Y^* = [y^*(1), y^*(2), \ldots, y^*(k_F - 1), y^*(k_F)]^T \qquad (9)$$

We can check the desired water level vector values in a computational grid in Figure 5 (dots).

In another way, we can obtain the 'state vector' $x(k)$ which is defined as the vector containing the water level and velocity predictions established from the output data of the CSE algorithm at the time instant k of all the discretization points:

$$x(k) = [y_1(k), \ldots, y_i(k), v_i(k), \ldots, v_{n_S}(k)]^T \qquad (10)$$

where $y_i(k)$ and $v_i(k) =$ water depth and mean velocity at point i; and $n_S =$ number of cross sections in which the canal is discretized. In this way, the vector $x(0)$ is the known initial condition.

The state vector at the current time defines the current hydrodynamic state; we show the state vector in a computational grid in Figure 5 (triangles).

We may include all state vectors (Equation (11)) for each k-instant during a past time horizon into a single vector that is called 'prediction vector' (11); the dimension of this vector is $n_x = (2 \times n_s) \times k_F$, where ns is the number of cross sections of the canal:

$$X_1^{k_F} = [x(1)^T, x(2)^T, \ldots, x(k_F - 1)^T, x(k_F)^T]^T \qquad (11)$$
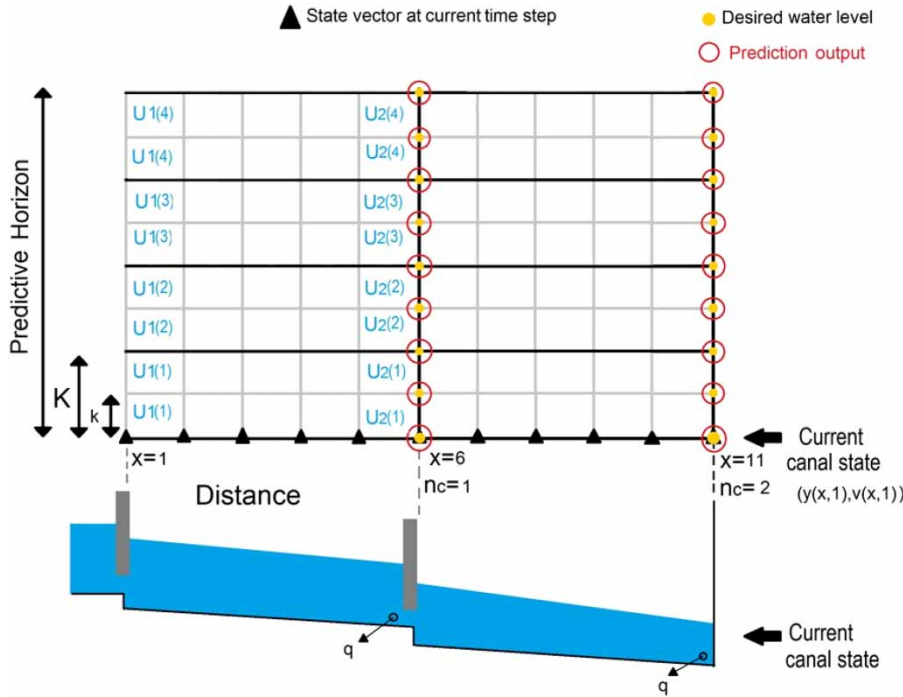
Figure 5 │ Sketch of a numerical grid of a canal with two pools controlled by two checkpoints downstream of each pool. There are pump stations close to each checkpoint. Pump flow trajectories are defined with four operation periods. Also, it shows the x/t-dots where the flow behavior is defined. Note that capital 'K' denotes time interval of control and 'k' with a small letter denotes time instant of simulation.

Because we are only interested in the water level at target points, we define a new vector called 'prediction output vector' that contains the water depth values given at a prescribed number of points (n_c) from the time instant 1 to k_F:

$$y(k) = [y_1(k), \ldots, y_i(k), \ldots, y_{n_C}(k)]^T \qquad (12)$$

The dimension of the prediction output vector is $n_Y = k_F \times n_c$. The vector (Equation (12)) contains all water depth values exactly as the dots (x/t-dots) shown in Figure 5. If you look closely at this figure, the position of the elements of the vector (12) in the grid domain coincides with the elements of the desired water level vector.

As mentioned before, GoRoSoBo calculates the gates trajectories at several points (for instance, pump stations) during a future horizon. In that case, as illustrated in Figure 6, the gates are moving on an operation period K. Then, the gates trajectories can be approached with piece-wise functions. The gates trajectories vector is defined by lumping together all the gate trajectories during the future

horizon, as follows:

$$U = [U_1(1), \ldots, U_{n_g}(1), \ldots, \ldots, U_1(K_F), \ldots, U_{n_g}(K_F)]^T \qquad (13)$$

where the dimension of this vector is $n_U = n_u \times K_F$, nu is the number of gates, and K_F is the final operation period of the future horizon. Thus, the GoRoSoBo algorithm calculates the gate trajectories (ΔU) to return from the predicted water level to the desired water level (ΔY) (1).

If we focus on the optimization problem, the objective is to make the prediction output vector more similar to the desired water level vector by manipulating the gate trajectories vector (Gill *et al.* 1981; Fletcher 1987). In
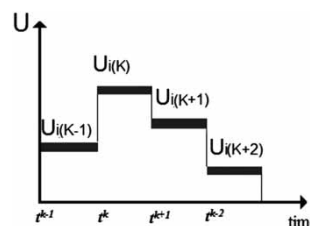

Figure 6 │ Mathematical representation of a pump flow trajectory.

mathematical terms, the objective is to obtain the gate trajectories vector that minimizes the following performance criterion:

$$\text{Minimize } J(U) = \frac{1}{2}[[C]X_{K_i}^{K_F}(U) - Y^*]^T[Q][[C]X_{K_i}^{K_F}(U) - Y^*]$$
$$r_k(U) = 0, \quad i \in I(U)$$
$$r_k(U) \geq 0, \quad i \in NI(U)$$

(14)

where $J(U)$ is the objective function, $X_{ki}^{kf}(U)$ is the prediction vector from the regulation time step K_i to K_F, Y^* is the desired water level vector, Q is a weighting matrix, C is the discrete observer matrix (Malaterre 1994) and $r_k(U)$ is the 'kth' constraint function, $I(U)$ is a set of equalities constraints, and $NI(U)$ is a set of inequalities constraints. U contains the gate trajectories (13).

RESULTS AND DISCUSSION

Numerical example: a canal introducing a single disturbance

In order to explain the process in an easy way, we proposed several scenarios to test the overall control diagram in a canal with just one pool and controlled by only one gate, upstream of the canal (Figure 7). The flow is controlled by this gate downstream from a reservoir providing all the water the canal needs. Water is delivered through gravity
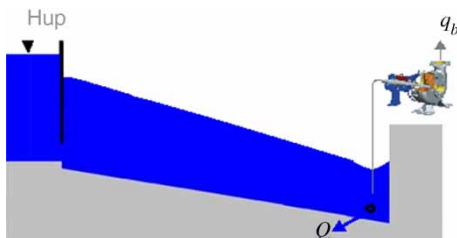

Figure 7 | Canal profile with a single pool.

outlets at the downstream end of the pool, where the checkpoint is located. There is a pumping station at the end of the pool which can introduce disturbances on the system in space and time.

The geometry of the canal proposed is based on Bautista *et al.* (1997). This canal was used by different authors including Wylie (1969), Liu *et al.* (1992), Chevereau (1991), and Soler (2003). The canal geometry adopted in our examples is based on Liu's example as well as the scheduled demand which was introduced in some scenarios.

The canal, with a trapezoidal section, is represented in Figure 7, and the general data are shown in Table 1. The characteristics of the checkpoint, sluice gate, pump station, and orifice offtake are shown in Tables 2 and 3.

In these examples an upstream large reservoir is considered, whose water level $H_{\text{reservoir}}$ is constantly 3 m throughout the test. At the end of the last pool there is a control structure with an orifice offtake and a pump station. The flow through the orifice depends on the level over the orifice and the disturbance is introduced by the pump station. This is the downstream boundary condition. This example starts from an initial steady state (Tables 4 and 5) with a specific demand delivery constant at the end of the pool (5 m³/s through the orifice offtake), and the disturbance is not introduced initially.

The disturbance

In order to test the algorithm, we introduce a disturbance in the canal, which is unknown for the watermaster. Consequently, once the disturbance is introduced there will be variations between the measured water level and the desired water level. In that sense, the water level measurements (Figure 8) deviate from the desired water level after introducing a 'step disturbance', which is introduced 30 minutes after starting the test and lasts until the end of the test, with a value of 2 m³/s. This disturbance is introduced to the computer model as a pump flow change and we obtain

Table 1 | Canal features

Pool number	Pool length (km)	Bottom slope (%)	Side slopes (H:V)	Manning's coefficient (n)	Bottom width (m)	Canal depth (m)
I	2.5	0.1	1.5:1	0.025	5	2.5

Table 2 │ Sluice gate features (canal structures)

Number of control structure or checkpoint	Gate discharge coefficient	Gate width (m)	Gate height (m)	Step (m)
0*	0.61	5.0	2.0	0.0

*Reservoir gate.

Table 3 │ Pump station/orifice offtake features (canal structures)

Number of control structure or checkpoint	Discharge coef./ diameter orifice offtake (m)	Orifice offtake height (m)	Lateral spillway height (m)	Lateral spillway width (m)/discharge coefficient
0*	–	–	–	–
1	2/0.85	0.8	2.0	5/1.99

*Reservoir gate.

Table 4 │ Initial conditions in the canal

Control structure	Initial flow rate (m³/s)	Control structure	Initial water level (m)
Gate 1	5.0	Checkpoint 1	1.6

Table 5 │ Flow delivered by an orifice offtake at the initial time step

Control structure	Flow delivered by an orifice offtake (m³/s)
Gravity outlet 1	5.0

the water level measurements from this mode. Once these water level measurements are introduced in the CSE algorithm, it will propose the pump flow trajectories that describe with best accuracy the variation of water level at the checkpoints during the past time horizon and the hydrodynamic state at the current time step. All these data are sent to GoRoSoBo which sets the predicted water level vector for a future horizon and recalculates the optimum gate trajectories to ensure the target level over the orifices for this future horizon. As we showed in Figure 2, all these actions are repeated every regulation period.

The constraints of the optimization problem are imposed on the gate position, so that the sluice gate opening must not be greater or smaller than U_{max} or U_{min}, respectively, and the gate movements between successive regulation periods (dU_{max}) should be physically acceptable (Table 6).

The gate trajectory is defined by the gate position in time, and the gate position is defined as follows:

$$Gate\ position = \frac{Gate\ opening}{Gate\ height} \tag{15}$$

Results

We represent with a discontinuous line the water level results obtained just using a feedforward algorithm like GoRoSo (Soler 2003; Soler *et al.* 2013) and with a continuous line the results obtained by the overall control diagram at the target point in Figure 8. We can compare the sluice gate trajectories obtained by GoRoSoBo vs. the sluice gate trajectories obtained by GoRoSo in Figure 9.

At the beginning of the test, the flow in the canal is steady and the scheduled delivery is constant for all demand periods, and the sluice gate position remains fixed. After the first 30 minutes (1,800 s), a disturbance is introduced into the system, although the algorithm has no notice until the next regulation period (2,100 s) once the water level is measured at the checkpoint, and the water level has already been reduced by more than 10 cm, from 1.6 cm to 1.50 cm (Figure 8).

Once the water level measurements are sent to CSE, it calculates the extracted flow vector for a past time horizon. CSE establishes the disturbance introduced in the system. This information is sent to GoRoSoBo which modifies the sluice gate trajectory to keep the water level constant at the checkpoint which is not going to increase until three regulation periods later (2,700 s). This is because once the sluice gate generates a wave for increasing the water level at the checkpoint, the wave has to reach the checkpoint (time delay). Once the wave arrives at the checkpoint, the water level increases quickly, recovering the target level of 1.6 m at 3,700 s (Figure 8).

The maximum deviation between the water level measured and desired is around 27.5 cm, from 1.6 cm to 1.325 cm, so the sluice gate movement must be quite significant in order to quickly reduce this deviation (Figure 9). The gate position changes from 0.125 to 0.34 m during a regulation period (1,800–2,100 s).
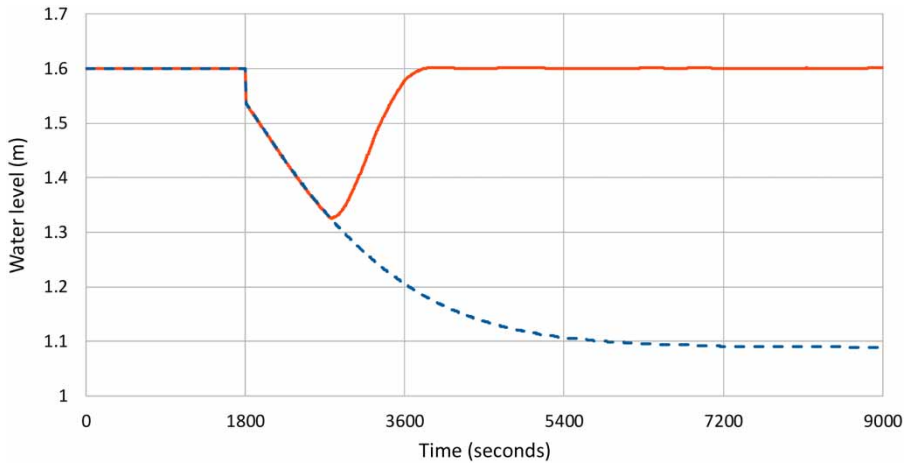
Figure 8 │ Water level at the checkpoint obtained by GoRoSoBo (continuous line)/GoRoSo (discontinuous line).

Table 6 │ Overall and functional constraint values

U_{min} (%)	U_{max} (%)	dU_{max} (%)
0.05	90	2.5

The feedforward algorithm (GoRoSo) does not modify the gate trajectory during the irrigation cycle because the scheduled demand is constant and the algorithm does not know that someone has introduced a disturbance in the canal and the water level decreases at checkpoint. The water level decreases to 1.1 m so the flow extracted by the offtake is close to 3 m³/s (Figure 8). In that sense, the total extracted flow is 5 m³/s, but 3 m³/s are extracted by

the offtake and 2 m³/s are extracted by someone who introduced the disturbance. The difference in gate opening between the feedforward and the feedback algorithms at 9,000 s is due to the disturbance. A difference of 0.06 m in gate opening represents a flow of 2 m³/s in this canal with this flow condition.

Numerical example: a canal with multiple disturbances at the same time: ASCE test cases

In this numerical example, we want to evaluate our overall control diagram in a canal with several pools with multiple flow extractions at the same time. In that case, we introduce
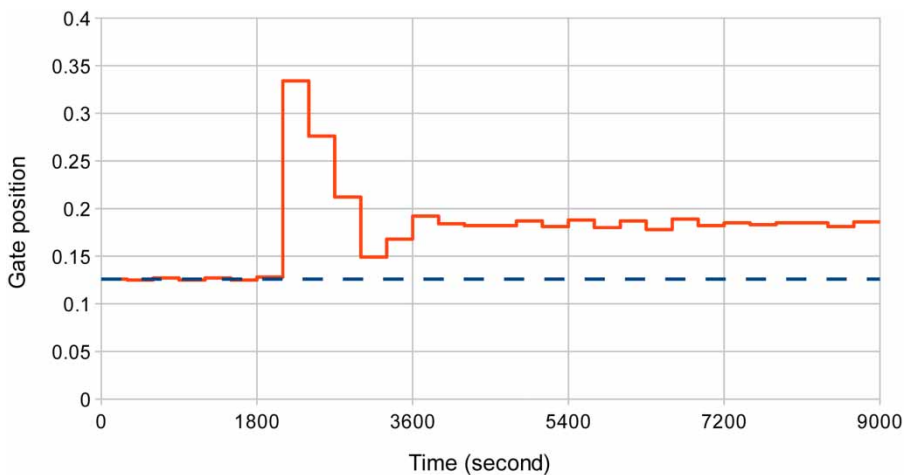

Figure 9 │ Gate trajectory obtained by GoRoSoBo (continuous line)/GoRoSo (discontinuous line).

Table 7 | Features of Maricopa Stanfield canal pools

Pool number	Pool length (km)	Bottom slope	Side slopes (H:V)	Manning's coefficient (n)	Bottom width (m)	Canal depth (m)
I	0.1	2×10^{-3}	1.5:1	0.014	1	1.1
II	1.2	2×10^{-3}	1.5:1	0.014	1	1.1
III	0.4	2×10^{-3}	1.5:1	0.014	1	1.0
IV	0.8	2×10^{-3}	1.5:1	0.014	0.8	1.1
V	2	2×10^{-3}	1.5:1	0.014	0.8	1.1
VI	1.7	2×10^{-3}	1.5:1	0.014	0.8	1.0
VII	1.6	2×10^{-3}	1.5:1	0.014	0.6	1.0
VIII	1.7	2×10^{-3}	1.5:1	0.014	0.6	1.0

the test cases proposed by the ASCE committee to evaluate control algorithms.

Two canals are evaluated by the ASCE Task Committee (Clemmens *et al.* 1998) in several scenarios. Each canal has eight pools separated by undershot sluice gates. All the canal pools have been discretized and numbered in the direction of flow from upstream to downstream. Geometric characteristics of canal 1 (Maricopa Stanfield canal) are shown in Tables 7 and 8 and Figure 10, and for canal 2 (Corning canal) in Tables 9 and 10 and Figure 11. In both canals, there are gravity outlet orifices at the downstream end of each canal pool but only in canal 2 is there direct pumping at the end of the last pool. The ASCE Committee proposes four cases to test feedback controllers in real time, two in the Corning canal and two in the Maricopa Stanfield. We only test one case in each canal with our overall control diagram taking into account the case with more unscheduled

flow changes. The initial conditions for all tests are shown in Table 11 (Maricopa Standfield) and Table 12 (Corning canal). All scenarios are evaluated in Bonet (2015).

The gate discharge coefficient for the gates in Corning canal is equal to the Maricopa Stanfield.

Scenario: Test case 1-2 (Maricopa Stanfield)

The unscheduled deliveries are more significant at target 8 in test case 1-2, where the flow rate changes from 0.9 m³/s to 0.6 m³/s (33%).

Scenario: Test case 2-2 (Corning canal)

The unscheduled water deliveries in test case 2-2 are very relevant, as these water deliveries are a huge amount of flow and these unscheduled water deliveries represent a high percentage with respect to the total flow of the canal. In this case, the unscheduled water deliveries are 81% on

Table 8 | Maricopa Stanfield control structures

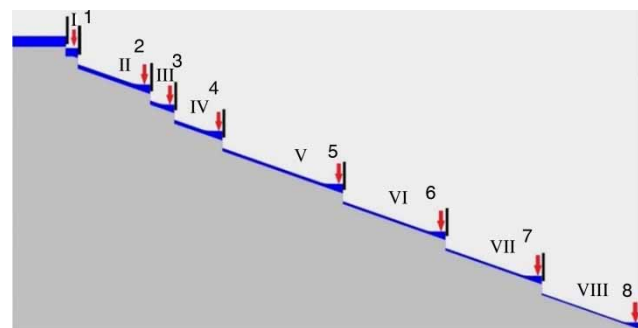| Target points	Gate discharge coefficient	Gate width (m)	Gate height (m)	Step (m)	Length from gate 1 (km)	Orifice offtake height (m)
0	0.61	1.5	1.0	1.0	0	–
1	0.61	1.5	1.1	1.0	0.1	0.45
2	0.61	1.5	1.1	1.0	1.3	0.45
3	0.61	1.5	1.0	1.0	1.7	0.40
4	0.61	1.2	1.1	1.0	2.5	0.45
5	0.61	1.2	1.1	1.0	4.5	0.45
6	0.61	1.2	1.0	1.0	6.2	0.40
7	0.61	1.0	1.0	1.0	7.8	0.40
8	–	–	–	–	9.5	0.40

Figure 10 | Maricopa Stanfield canal profile. The arrows mark the position of checkpoints. The first pool is number I and the first checkpoint is number 1.

Table 9 | Features of Corning canal pools

Pool number	Pool length (km)	Bottom slope	Side slopes (H:V)	Manning's coefficient (n)	Bottom width (m)	Canal depth (m)
I	7	1×10^{-4}	1.5:1	0.02	7	2.5
II	3	1×10^{-4}	1.5:1	0.02	7	2.5
III	3	1×10^{-4}	1.5:1	0.02	7	2.5
IV	4	1×10^{-4}	1.5:1	0.02	6	2.3
V	4	1×10^{-4}	1.5:1	0.02	6	2.3
VI	3	1×10^{-4}	1.5:1	0.02	5	2.3
VII	2	1×10^{-4}	1.5:1	0.02	5	1.9
VIII	2	1×10^{-4}	1.5:1	0.02	5	1.9

the total flow and these water changes are more significant at target 8, where the flow rate changes from $5 \, \text{m}^3/\text{s}$ to $1 \, \text{m}^3/\text{s}$. For this reason, it should be noted that 15 minutes after introducing the unscheduled water deliveries, the water level at target 8 is 1.93 meters, so the lateral spillway must operate. This is the maximum flow rate change at a target in all test cases.

Constraints

The constraints are determined as a certain percentage of the gate height, imposed at the gate positions (Table 13).

Results

We show the results obtained in each test case dividing these into water level at checkpoints and gate trajectories

Table 10 | Corning canal control structures

Target points	Gate width (m)	Gate height (m)	Step (m)	Length from gate 1 (Km)	Orifice offtake height (m)	Lateral spillway height (m)
0	7	2.3	0.2	0	–	3
1	7	2.3	0.2	7	1.05	2.5
2	7	2.3	0.2	10	1.05	2.5
3	7	2.3	0.2	13	1.05	2.5
4	6	2.1	0.2	17	0.95	2.3
5	6	2.1	0.2	21	0.95	2.3
6	5	1.8	0.2	24	0.85	1.9
7	5	1.8	0.2	26	0.85	1.9
8	–	–	–	28	0.85	1.9

which were also divided in a graph with only four checkpoints or four gates, respectively, to analyze the results properly.

Test case 1-2 (Maricopa Stanfield)

In this test 1-2, the canal is in steady state during the first 2 hours. After that, unscheduled water deliveries are introduced into the system at 7,200 s, although the algorithm has no notice until the next regulated period, once the water level is measured at the checkpoints. The gate trajectories calculated by GoRoSoBo are shown in Figures 12 and 13.

It is important to remark that the unscheduled water deliveries are relevant in all targets but especially at target 8, because just in one regulation period, the water level at checkpoint 8 increases from 0.8 m to 1.08 m (Figure 14), more than 25% from the initial condition. A water delivery change of $0.3 \, \text{m}^3/\text{s}$ at checkpoint 8 is introduced, with a total flow rate change at target 8 from $0.6 \, \text{m}^3/\text{s}$ to $0.9 \, \text{m}^3/\text{s}$, that is, a flow rate change of around 50% from the initial one.
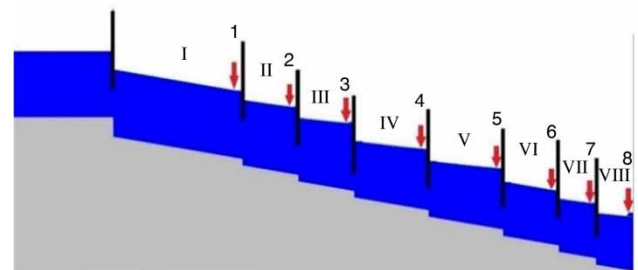

Figure 11 | Corning canal profile. The arrows mark the position of checkpoints. The first pool is number I and the first checkpoint is number 1.

Table 11 │ Initial and unscheduled offtake changes in test case 1-2

Pool number	Offtake initial flow (m³/s)	Check initial flow (m³/s)	Unscheduled offtake changes at 2 hours (m³/s)	Check final flow (m³/s)
Heading	–	2.0	–	2.0
1	0.2	1.8	–	1.8
2	0.0	1.8	0.2	1.6
3	0.4	1.4	−0.2	1.4
4	0.0	1.4	0.2	1.2
5	0.0	1.4	0.2	1.0
6	0.3	1.1	−0.1	0.8
7	0.2	0.9	–	0.6
8	0.9	0.0	−0.3	0.0

Taking into account that all water delivery changes are relevant, such as these which are 0.2 m³/s at targets 1 to 5, 0.1 m³/s at target 6, and 0.3 m³/s at target 8, in a canal that has no storage capacity and a steep slope, this is a hard test for a control algorithm.

Once the GoRoSoBo algorithm knows that the water level is increasing less at checkpoints 2, 4, and 5 at time 7,500 s (Figures 14 and 15), all gates except 7 and 8 have the tendency of closing (Figures 12 and 13).

We can check that the biggest change in gate position at 18,000 s is gate 8 which is logical because the more important flow change is at checkpoint 8.

The sluice gate trajectories have the same movement as the water level at the checkpoints, due to the gate trajectories and water levels fluctuating around the desired solution.

Table 12 │ Initial and unscheduled offtake changes in test case 2-2

Pool number	Offtake initial flow (m³/s)	Check initial flow (m³/s)	Unscheduled offtake changes at 2 hours (m³/s)	Resulting check flow (m³/s)
Heading	–	13.7	–	2.7
1	1.7	12.0	−1.5	2.5
2	1.8	10.2	−1.5	2.2
3	2.7	7.5	−2.5	2.0
4	0.3	7.2	–	1.7
5	0.2	7.0	–	1.5
6	0.8	6.2	−0.5	1.2
7	1.2	5.0	−1.0	1.0
8	0.3 + 2.0*	2.7	−2.0*	0.7

*Changes in downstream pump discharge with no change in offtake flow.

Table 13 │ Overall and functional constraint values

	U_{min} (%)	U_{max} (%)	dU_{max} (%)
Maricopa Stanfield	2	90	5
Corning canal	0.5	90	5

The water level decreases in all targets at 9,900 s due to changes in gate trajectories, and the water level returns to the desired water level in all checkpoints at 21,600 s.

The water levels were almost equal to the desired levels at the end of the test, with a maximum error between them of around 0.3 cm.

Test case 2-2 (Corning canal)

Test case 2-2 is the most difficult test due to the significant unscheduled water deliveries in all targets. The canal state is steady during the first 2 hours, as in the previous test case. There are no unscheduled water deliveries for the first 2 hours, and then unscheduled water deliveries are introduced into the system. The gate trajectories calculated by GoRoSoBo are shown in Figures 16 and 17. GoRoSoBo obtained good results in test case 2-2, as we show in Figures 18 and 19, although it is important to remark that the unscheduled water deliveries are so important, especially at target 8, that once the algorithm has notice of the rising water level at checkpoint 8 at 8,100 s, the water level has already increased 20 cm, so it reaches the top of the cross section. In those circumstances, GoRoSoBo closes sluice gates 1 to 3 and opens sluice gates 4 to 8 (Figures 16 and 17), so in this way the sluice gate trajectories reduce the increased water level at all checkpoints.

Redirecting the water level at checkpoints to the desired water level in a centralized system is not the duty of only one sluice gate. It is possible that some sluice gates have increased the water level error in some checkpoints at a regulation period (for instance, checkpoints 4 and 5) when there are no unscheduled offtake changes in these checkpoints, but the main objective is to redirect the measured water level to the desired values in all checkpoints as soon as possible.

The water level decreases at all checkpoints at 25,200 s (7 hours) and the water level returns to the desired value at
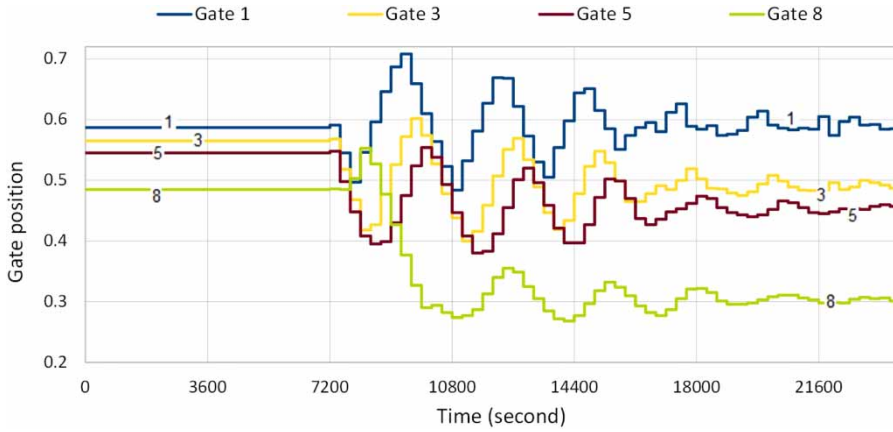
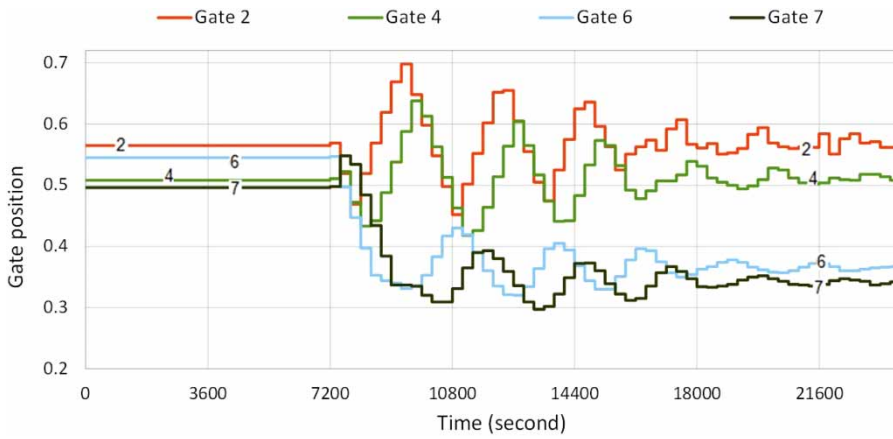**Figure 12** │ Gate trajectories (1, 3, 5, and 8) in test case 1-2 (Maricopa Stanfield).

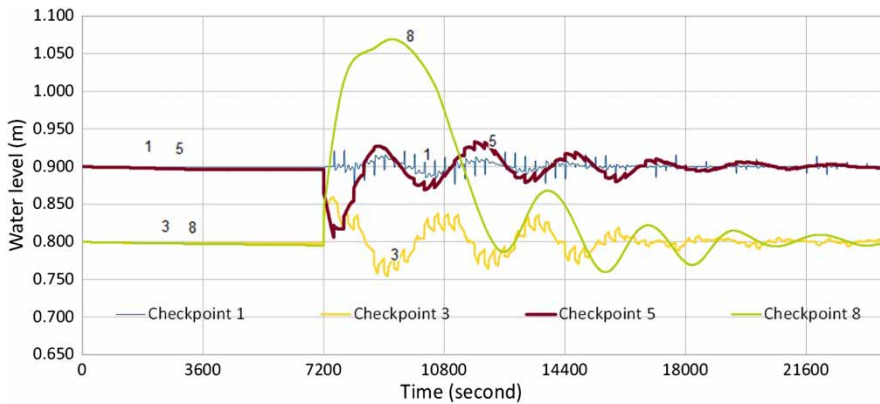**Figure 13** │ Gate trajectories (2, 4, 6, and 7) in test case 1-2 (Maricopa Stanfield).

Figure 14 │ Water level at checkpoints 1, 3, 5, and 8 in test case 1-2 (Maricopa Stanfield).
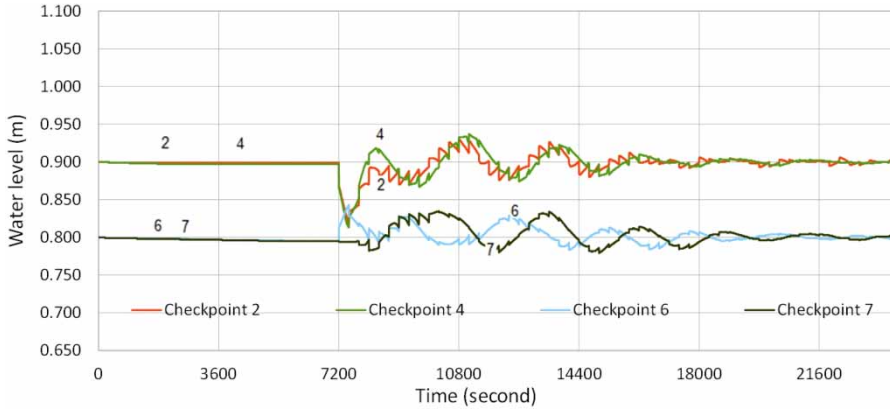
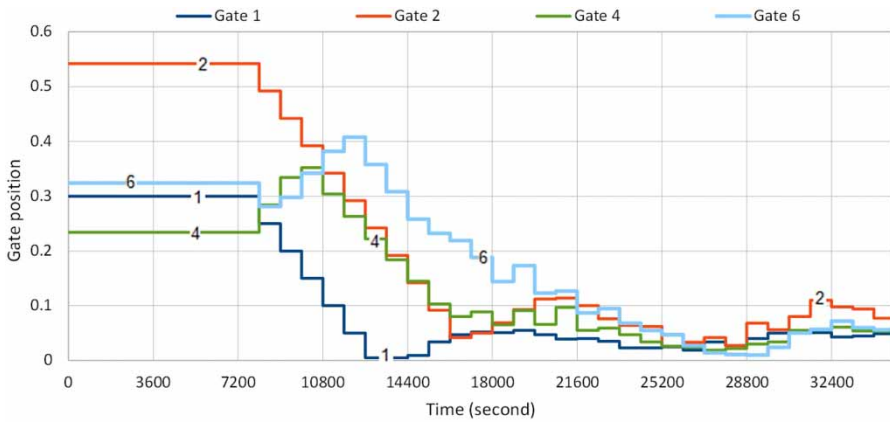**Figure 15** │ Water level at checkpoints 2, 4, 6, and 7 in test case 1-2 (Maricopa Stanfield).

Figure 16 │ Gate trajectories (1, 2, 4, and 6) in test case 2-2 (Corning canal).
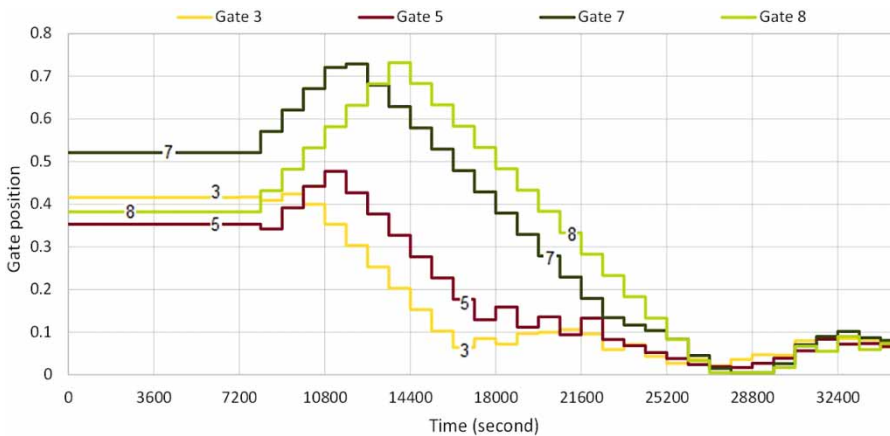
Figure 17 │ Gate trajectories (3, 5, 7, and 8) in test case 2-2 (Corning canal).
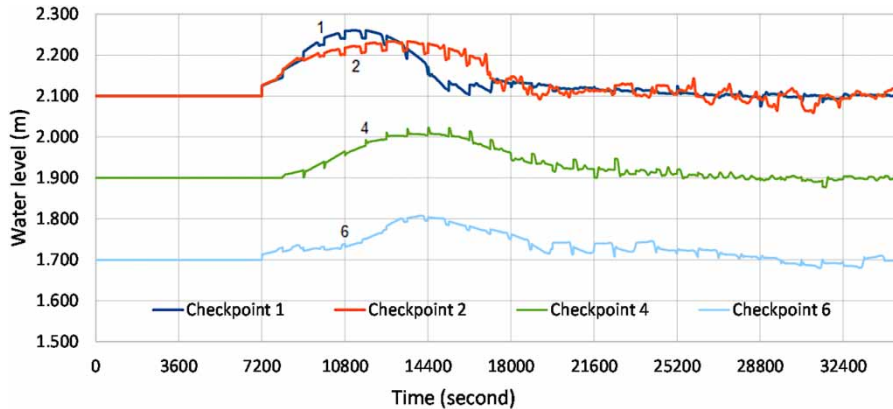
Figure 18 │ Water level at checkpoints 1, 2, 4, and 6 in test case 2-2 (Corning canal).
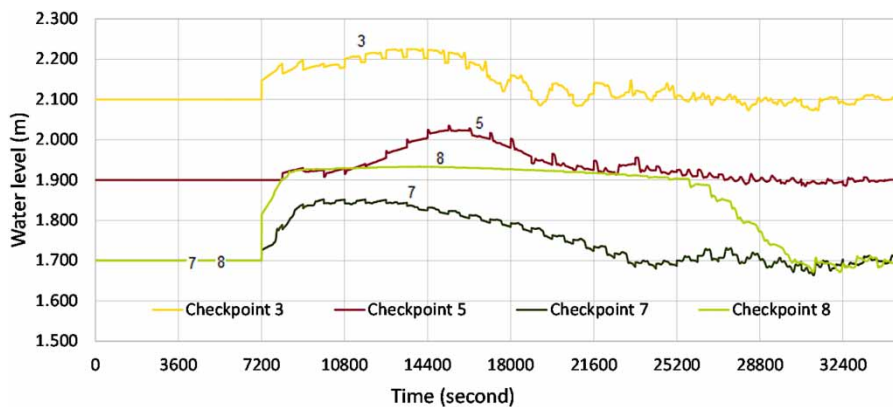

Figure 19 │ Water level at checkpoints 3, 5, 7, and 8 in test case 2-2 (Corning canal).

all the checkpoints at 30,600 s. The unscheduled water delivery is quite important at target 8, introducing a total flow change of 4 m³/s at 7,200 s because the total flow rate at pool 8 is 5 m³/s at the initial time step and the flow rate at pool 8 is 1 m³/s at 7,200 s, so the flow change is close to 80% of the total flow rate of the pool.

The gate movements are almost zero in the last 2 hours of the test because the water levels at checkpoints are equal to the desired values at the end of the test, with a maximum error of around 0.5 cm.

Performance indicators

These indicators were introduced to compare different control algorithms, as it is not easy to evaluate them and judge the controller's ability to deliver water. This is the reason why the ASCE Task Committee devised these test cases.

Although not all these performance indicators evaluate the deviations between the measured and the desired water level at the checkpoints, they take into account other variables such as the changes in flow.

We compare the performance indicators obtained with GoRoSoBo in the test cases with the same indicators obtained with other control algorithms like CLIS (Liu *et al.* 1998) and PILOTE (Malaterre 1995) (see Tables 14 and 15). CLIS is based on an inverse solution method of the Saint-Venant equations, and it is designed for the automation of demand-oriented systems. PILOTE is a LQR closed-loop controller and it is obtained from the steady-state solution of the Riccati equation, a Kalman filter is used to reconstruct the state variables and the

Table 14 │ The performance indicators obtained in test case 1-2 (Maricopa Stanfield)

Test case 1-2 tuned unscheduled

	MAE (%) 12–24 h		*IAE* (%) 12–24 h		*StE* (%) 12–24 h		*IAQ* (m³/s) 12–24 h	
	Max	Average	Max	Average	Max	Average	Max	Average
CLIS	34.5	14.2	5.0	2.0	3.6	1.1	0.2	0.1
PILOTE	43.0	24.9	9.2	5.2	11.2	2.9	2.9	1.4
GoRoSoBo	33.5	10.3	5.0	1.2	2.1	0.7	6.7	3.6

Table 15 │ The performance indicators obtained in test case 2-2 (Corning canal)

Test case 2-2 tuned unscheduled

	MAE (%) 12–24 h		*IAE* (%) 12–24 h		*StE* (%) 12–24 h		*IAQ* (m³/s) 12–24 h	
	Max	Average	Max	Average	Max	Average	Max	Average
CLIS	21.1	14.9	7.6	2.8	0.7	0.4	9.7	5.5
PILOTE	34.2	17.1	10.6	7.1	8.8	4.3	10.4	6.1
GoRoSoBo	13.6	7.8	6.3	2.1	0.8	0.5	15.2	11.7

unknown perturbations from a reduced number of observed variables.

The *MAE* is the maximum deviation of the controlled water level at the checkpoint with regard to the desired water level. From the values obtained by GoRoSoBo with this performance indicator, GoRoSoBo shows the best results in all tests.

The *IAE* is the integrated deviation of controlled water level on the target water depth. From the maximum and average *IAE* values obtained by GoRoSoBo, this algorithm shows the best results in three of four cases.

The *StE* is the deviation of controlled water level at steady state over target water depth, so we only consider the last 2 hours of the irrigation cycle to calculate this performance indicator. GoRoSoBo obtained the best results in three cases and it was the second in two cases.

The *IAQ* gives us an idea of how many changes in flow rate are introduced by the gates to reach the desired flow rate at the end of the test. GoRoSoBo shows the highest values on this performance indicator. When a control algorithm has, as an only objective, to maintain a desired water

level at the checkpoints, the gate trajectories are calculated for that objective and some performance indicators benefit more (i.e., MAE/IAE) than others (i.e., IAQ). In that sense, when significant flow changes are introduced in the system, quick changes on gate trajectories are necessary to recover the desired value at checkpoints as soon as possible. For this reason, increasing the constraints more in gate movements to obtain better values of this performance indicator is not our principal objective. We can conclude that the magnitude of flow changes through the gates is acceptable and the control algorithm maintains the measured water level close to the desired water level and there exists a functional constraint to restrict the gate movement.

CONCLUSIONS

The GoRoSoBo algorithm is able to find the optimum gate trajectory during a predictive horizon taking into account demand deliveries, initial gate trajectories, desired water level vector, disturbances, and the current canal state

which are obtained by CSE. All these data are introduced into GoRoSoBo and the algorithm recalculates the optimum gate trajectory to keep the desired water level constant at checkpoints.

The GoRoSoBo algorithm uses the Lagrange–Newton method to solve a constrained optimization problem. This method is considered the most efficient when the Jacobian matrix and the Hessian matrix, which are used in the computation of the gate trajectories, have been compiled.

The introduction of constraints was absolutely necessary to ensure stability in our optimized problem, due to inherent instability in the unconstrained problem, which is caused by the condition number of the Hessian and the HIM matrixes. Not all elements of the HIM have similar values; there are gates that have a significant influence in certain checkpoints and little influence in others. This disparity of influence between elements of the matrix inevitably leads to a band matrix, of course, badly conditioned. This was a reason to use the Marquardt coefficient which improves the Hessian matrix condition.

The watermaster can be more or less strict about the constraints in the optimization problem, because he must consider the main priority of the canal. For instance, if the main priority is to maintain the water level constant at any price, the constraints on the gates would be lax, so the constraints will not have an important role in computing the gate trajectories by GoRoSoBo. If the main priority is to keep the water level constant, with reduced gate movements, the constraints would be more restrictive, so they will have an important role in the optimization problem.

The *IAQ* value is much higher in GoRoSoBo than in other controllers proposed. This index is strongly linked to the main priority of the canal. In the case where the priority is to keep the water level constant at the checkpoints, the gate movements are significant and the flow rate variations through the sluice gate also, so the *IAQ* value will also be significant.

One of the main problems of our overall control diagram is the computation time. In cases where the predictive horizon is large, the regulation interval is small, the canal length and the number of checkpoints are significant, the calculation time of the algorithm is too large for operating in real time and much more so if it is necessary to update the hydraulic influence matrix in every regulation time step. A solution with the time conflict would be to parallelize the algorithm with OpenMP to reduce the CPU time, as current workstations have several processors with dozens of cores which would increase the speed-up in solving the problem. Another possibility to speed-up the calculating process would be if GoRoSoBo calculated the gate trajectories using the HIM in a previous time step, while at the same time, another algorithm calculated the HIM at the current time step to be used by GoRoSoBo in the next time step.

The results obtained by GoRoSoBo using our overall control diagram have been very satisfactory, as we have shown in the case of a simple canal and the test cases by the ASCE.

REFERENCES

AEMET (Agencia Estatal de Meteorología) 2009 *Generation of Regional Scenarios of Global Warming in Spain (Generación de escenarios regionalizados de cambio climático para España)*. M. Brunet, M. J. Casado and & M. de Castro, Ministerio de Agricultura, Pesca y Alimentación, Spain.

Bautista, E., Clemmens, A. J. & Strelkoff, T. 1997 Comparison of numerical procedures for gate stroking. *Journal of Irrigation and Drainage Engineering* **113** (2), 129–136.

Bautista, A. M., Strelkoff, T. S. & Clemmens, A. J. 2003 General characteristics of solutions to the open-channel flow, feedforward control problem. *Journal of Irrigation and Drainage Engineering* **129** (2), 129–137.

Bonet, E. 2015 *Experimental Design and Verification of a Centralize Controller for Irrigation Canals*. PhD thesis, UPC, Technical University of Catalonia, Spain.

Bonet, E., Gómez, M., Soler, J. & Yubero, M. T. 2017 CSE algorithm: 'canal survey estimation' to evaluate the flow rate extractions and hydraulic state in irrigation canals. *Journal of Hydroinformatics* **19** (1), 62–80. DOI: 10.2166/hydro.2016.014.

Bruinsma, J. 2009 The Resource Outlook to 2050: By How Much do Land, Water and Crop Yields Need to Increase by 2050? Prepared for the FAO Expert Meeting on 'How to Feed the World in 2050', 24–26 June 2009, Rome, Italy.

Chevereau, G. 1991 *Contribution to the Study of Regulation in Hydraulic Systems Free Surface*. PhD thesis, Institut National Polytechnic de Grenoble, France.

Clemmens, A. J. & Wahlin, B. T. 2004 Simple optimal downstream feedback canal controllers: ASCE test case results. *Journal of Irrigation and Drainage Engineering* **130** (1), 35–46.

Clemmens, A. J., Kacerek, T. F., Grawitz, B. & Schuurmans, W. 1998 Test cases for canals control algorithms. *Journal of Irrigation and Drainage Engineering* **124** (1), 23–30.

FAO (Food and Agriculture Organization of the United Nations) 2011a *The State of the World's Land and Water Resources: Managing Systems at Risk.* Earthscan, London, UK.

FAO (Food and Agriculture Organization of the United Nations) 2011b *AQUASTAT online database.* FAO, Rome, Italy. http://www.fao.org/nr/water/aquastat/data/query/index.html.

Fletcher, R. 1987 *Practical Methods of Optimization,* 2nd edn. John Wiley & Sons, Chichester, UK.

Gill, P. E., Murray, W. & Wright, M. H. 1981 *Practical Optimization.* Academic Press, London, UK.

Gómez, M. 1988 *The Contribution of Case Study of Transient Flow in Sewer Systems.* PhD thesis, UPC, Bacrcelona, Spain.

Liu, F., Feyen, J. & Berlamont, J. 1992 Computation method for regulating unsteady flow in open channels. *Journal of Irrigation and Drainage Engineering* 118 (10), 674–689.

Liu, F., Feyen, J., Malaterre, P. O., Baume, J. P. & Kosuth, P. 1998 Development and evaluation of canal automation algorithm CLIS. *Journal of Irrigation and Drainage Engineering* 124 (1), 40–46.

Malaterre, P. O. 1994 *Modelisation, Analysis and LQR Optimal Control of an Irrigation Canal.* PhD thesis, LAAS-CNRS-ENGREF-Cemagref, Etude EEE n14, France.

Malaterre, P. O. 1995 PILOTE: Optimal control of irrigation canals. In: *First International Conference on Water Resources Engineering, Irrigation and Drainage, San Antonio, Texas, USA,* 14–18 August 1995.

Marzouki, T. Z. 1989 The hydraulics of the river Larcis (dams, river, irrigation). Hydraulic modeling, analysis and improvement of management, Cemagref, ENGREF, CARA, 63 pp.

Ministerio de Medio Ambiente 2000 *White Book of Water in Spain.* MIMAM, Madrid, Spain.

PNACC 2011 http://www.magrama.gob.es/es/cambio-climatico/temas/impactos-vulnerabilidad-yadaptacion/2_informe_seguimiento_pnacc_tcm7-197096.pdf (accessed 10 October 2013).

Rogers, D. C. & Goussard, J. 1998 Canal control algorithms currently in use. *Journal of Irrigation and Drainage Engineering* 124 (1), 11–15.

Sanders, B. F. & Katopodes, N. D. 1999 Control of canal flow by adjoint sensitivity method. *Journal of Irrigation and Drainage Engineering* 125 (5), 287–297.

Soler, J. 2003 *Study About Controllers in Irrigation Canals Using Nonlinear Numerical Methods.* PhD thesis, Polytechnic University of Catalonia, Spain.

Soler, J., Gómez, M. & Rodellar, J. 2013 Goroso: feedforward control algorithm for irrigation canals based on sequential quadratic programming. *Journal of Irrigation and Drainage Engineering* 139 (1), 41–54. 10.1061/(ASCE)IR.1943-4774.0000507.

Soler, J., Gómez, M. & Bonet, E. 2015 Canal monitoring algorithm. *Journal of Irrigation and Drainage Engineering* 142 (3). 10.1061/(ASCE)IR.1943-4774.0000982, 04015058.

UNDESA (United Nations Department of Economic and Social Affairs) 2009 World Population Prospects: The 2008 Revision, Highlights. Working Paper No. ESA/P/WP.210. United Nations, New York.

Wahlin, B. T. & Clemmens, A. J. 2006 Automatic downstream water-level feedback control of branching canal networks: simulation results. *Journal of Irrigation and Drainage Engineering* 132 (3), 208–219.

Wylie, E. B. 1969 Control of transient free-surface flow. *Journal of the Hydraulics Division* 95 (1), 347–362.