Delft University of Technology

Delft Center for Systems and Control

Technical report 13-030

Event-driven hierarchical control of irrigation canals*

A. Sadowska, B. De Schutter, and P.-J. van Overloop

If you want to cite this report, please use the following reference instead:
A. Sadowska, B. De Schutter, and P.-J. van Overloop, "Event-driven hierarchical control of irrigation canals," *Proceedings of the USCID Seventh International Conference on Irrigation and Drainage*, Phoenix, Arizona, pp. 457–471, Apr. 2013.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

EVENT-DRIVEN HIERARCHICAL CONTROL OF IRRIGATION CANALS

Anna Sadowska[1]
Bart De Schutter[2]
Peter-Jules van Overloop[3]

ABSTRACT

We present a novel, simple and cost-effective strategy for control of irrigation canals to aid water deliveries to the users through the canal. The method enhances water deliveries through the canal by incorporating, alongside local PI controllers maintaining water levels in each canal pool at some predefined setpoints, a higher-layer centralized controller. The purpose of that centralized controller is to coordinate the local controllers by modifying the setpoints in individual pools. This speeds up the delivery process so that water is available to users faster than when only local controllers are used. Because the higher-layer centralized controller is invoked only when deliveries are requested and in normal operating conditions the canal is maintained merely by the local upstream PI controllers, the method is computationally efficient and resilient to temporary communication failures. We use Time Instant Optimization Model Predictive Control as the main control framework to design the higher-layer centralized controller and present a simulation study to illustrate the method proposed in this paper.

INTRODUCTION

Irrigation canals are open channels that transport water through the land from a source (a river or a reservoir) to users (farmers) for the purpose of irrigation of crops (Chow, 1959). They consist of a number of pools that are interconnected with one another in a cascade by hydraulic structures (e.g. gates) that control water flow between neighboring pools. Considering the vast importance of irrigation canals in agriculture, it is a crucial task to be able to control the water flow through a canal efficiently and effectively, and ideally with minimal resources involved.

There are numerous methods proposed in the literature to control water flow in irrigation canals (Malaterre and Baume, 1998; Rutz et al., 1998; Malaterre, 2007; Weyer, 2008). Some rely on manual controlling of the gates by a human operator. These methods, however, may fail to provide a sufficiently good performance due to many variables describing the state of the canal that the operator may need to account for, which ultimately may prove intractable. An alternative to manual operation of

[1] Delft Center for Systems and Control, Delft University of Technology, email: A.D.Sadowska@tudelft.nl
[2] Delft Center for Systems and Control, Delft University of Technology, email: B.DeSchutter@tudelft.nl
[3] Water Resources Management, Delft University of Technology, email: P.J.A.T.M.vanOverloop@tudelft.nl.

gates in a canal is automatic control where the gates between individual pools are controlled automatically employing various types of feedforward or feedback controllers (Van Overloop, 2006; Schuurmans, 1997). In general, the functioning of feedforward controllers depends entirely on the internal model of the system that a controller uses. Then, knowing external inputs affecting the system, a feedforward controller calculates the appropriate control inputs by verifying the effect of the disturbances on the model. In contrast, feedback controllers act in response to actual measured signals (e.g. water levels) and according to pre-specified rules, they set up the control inputs to counteract any disturbances given the observed effect of the disturbances on the system.

Controllers can be classified as centralized when there is a central governing entity that takes into account the state of the whole canal and calculates control inputs for all canal pools according to the information available (Malaterre, 2007; Sepúlveda Toepfer, 2007). The main advantage of centralized controllers is the fact that they have access to global information regarding the canal and thus see the broad picture of the situation. However, for large-scale problems centralized algorithms may require significant computational effort to produce results in a reasonable time due to many variables that they need to consider. On contrary, decentralized controllers use only local information. More specifically, to determine a suitable position of a gate in a pool, a controller takes into consideration the state of that particular canal pool and, possibly, the state of pools immediately upstream and downstream. Another kind of control algorithm using only local information are distributed controllers. They use inter-gate communication to find a control action that is not only satisfactory for themselves but also for the neighboring pools. Therefore, distributed controllers aim to obtain a solution that is of comparable quality to the solution that would be given by a centralized controller (Šiljak, 1991; Negenborn et al., 2009b). While in many dynamical systems it is beneficial to use distributed controllers, in the case of canal control it can be argued that in fact it is more advantageous to employ a centralized controller over a distributed controller due to the stationary nature of a canal and a large amount of communication between neighboring gates that may be needed for the gates to reach values of control actions suitable for all of them.

In practice, despite the benefits that sophisticated controllers can provide, very simple controllers are often used owing to their low cost and robust functioning. A very popular solution is the application of an upstream PI controller to control gate position locally in every canal pool (Litrico et al., 2003; Van Overloop et al., 2005; Litrico and Fromion, 2006; Litrico et al., 2007). PI controllers are feedback controllers that react to deviations in water levels with respect to some given setpoint (Åström and Hägglund, 1995). Since they do not depend on any particular model of the plant, they are widely used and in fact, when tuned properly, are able to provide a rather satisfactory performance.

In this paper we present a strategy to improve the performance of local PI controllers applied at individual gates for upstream control in order to facilitate water deliveries to users through irrigation canals. To that end we propose a hierarchical control structure, see Figure 1. Our proposed scheme is such that in normal operating conditions, only local PI controllers take care of the water flow in the canal. However, when any of the users announces a sudden delivery request, a higher hierarchy

centralized controller steps in to coordinate the individual canal pools and thus enhance the delivery process. This centralized controller (hereafter called the Coordinator) uses Model Predictive Control
tools to compute its control action and communicates it to the local sites only when changes are needed. Therefore, the alterations to the settings of the local sites are infrequent. In such a way, the local controllers remain fully in charge of the canal when there are no deliveries requested and the centralized controller is used to help coordinating the deliveries only. By allowing the local PI controllers to take care of the canal and changing the settings of the local sites infrequently to aid a delivery, we make sure that even if the communication lines fail for some time, the control remains acceptable due to the PI controllers, which can control the canal independently of the Coordinator.

As said earlier, the Coordinator only acts in response to a new delivery request. Thus, it is event-triggered as opposed to time-triggered. To that end, we propose to use the Time Instant Optimization (Van Ekeren et al., 2011), which allows to define how many times the setpoint may be changed and essentially optimize the switching time instants.

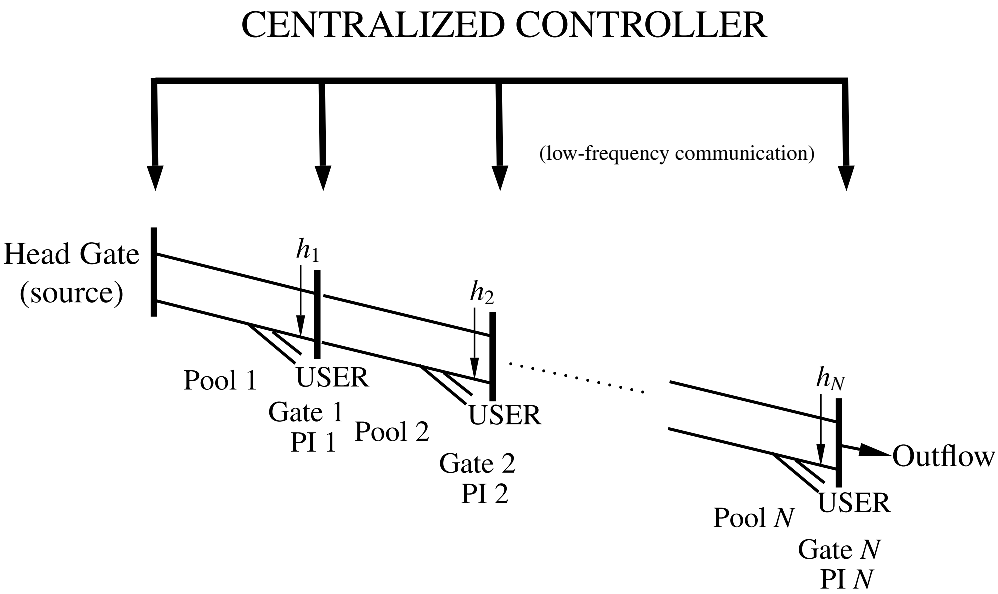

Figure 1. A schematic of the hierarchical controller proposed in the paper.

The rationale behind the application of the higher-layer centralized controller is as follows. When a delivery is requested and only local PI controllers are in use, it may take a considerable amount of time before water is available to users due to the time that is needed for water to travel from the source to the user. In particular, when only PI controllers operate and water is requested by one of the users, the required volume is released from the head gate. Then, as that extra amount of water reaches pool i, the local PI controller reacts to an increased water level and pushes it towards the subsequent pool $i+1$. This situation is repeated until water reaches the user. Therefore, depending on the distance between the head gate and the user, the delay may be significant. While the Coordinator cannot change the internal property of the canal of how long it takes for water to travel between the head gate and any single pool, the way the Coordinator works is to coordinate the local sites and by changing the setpoints in the pools, the Coordinator makes the water available to the user quicker than it would be without the Coordinator. Importantly, one of the main advantages of the way the Coordinator works is that it performs its job with minimal possible disruption to further parts of the canal. For example, if some base flow is needed for downstream users, the Coordinator aims at preserving that flow as closely as possible.

The idea of altering setpoints was previously proposed for power networks in (Negenborn et al., 2009a) where a supervisory control problem to prevent voltage collapse in the network was considered. Prominently, however, the supervisory controller provides new setpoints after each run of the optimizer, which is undesirable in our system because of the restriction of how much and how often the Coordinator can interfere with the local sites. For water systems, the concept of changing setpoints and hierarchical control structure was studied in (Zafra-Cabeza et al., 2011) for the purpose of risk management. In particular, various risk factors were defined: operational, financial, political and others. Considering these risk factors, the functioning of the higher hierarchy controller is to adapt the setpoints when needed to minimize the risk exposure. This control layer as well as the lower layer control are both realized employing the MPC strategy. Moreover, the lower layer controllers are distributed and thus, arguably, a significant volume of communication between individual controllers might be needed for the controllers to reach consensus on what control actions to apply. In addition, the scheme proposed in (Zafra-Cabeza et al., 2011) allows to change the setpoints freely in terms of frequency, which does not comply with our system's requirements.

The outline of this paper is as follows. In the next section we give some preliminaries regarding the topic: we present the dynamic model of the canal as well as present the main concepts of Model Predictive Control and Time Instant Optimization. Then, the hierarchical controller is introduced and, in the following section, its functioning is illustrated in a simulation study. After that we give our concluding remarks.

PRELIMINARIES

Model of an irrigation canal

In this section we present a dynamical model of a canal. We assume that the canal consists of N pools. Due to gravitational forces, water in a canal flows from upstream pools to downstream pools. The flow of water in a canal can be modeled using nonlinear partial differential equations, the so-called Saint Venant's equations (Chow, 1959; Van Overloop, 2006; Malaterre and Baume, 1998):

$$\begin{aligned} &\frac{\partial Q}{\partial x}+\frac{\partial A}{\partial t}=q_{\mathrm{lat}}, \\ &\frac{\partial Q}{\partial t}+\frac{\partial}{\partial x}\left(\frac{Q}{A}\right)^2+gA\frac{\partial h}{\partial x}+\frac{gQ|Q|}{C^2RA}=0. \end{aligned} \tag{1}$$

In the formula, Q denotes the flow, x is the longitudinal distance, A is the cross section area, t is time, q_{lat} is the unitary lateral inflow or outflow, g is the gravitational acceleration, h is the water height, R is the hydraulic radius and C is the Chézy constant, see (Chow, 1959; Van Overloop, 2006; Malaterre and Baume, 1998) for details. Unfortunately, the Saint Venant's equations are not suitable to be used for the purpose of real-time canal control due to their high complexity. However, by discretizing and linearizing the model, we obtain a simplified model that proves to be more efficient for control. For pool $i=1,\ldots,N$ the model reads

$$h_i(k+1)=A_ih_i(k)+B_{\mathrm{u}i}u_i(k)+B_{\mathrm{d}i}d_i(k), \tag{2}$$

where k is the discrete time step counter, h_i denotes water level at the end of pool i, u_i is the control input denoting the outflow from the canal, d_i is a disturbance inflow or outflow, and A_i, $B_{\mathrm{u}i}$ and $B_{\mathrm{d}i}$ are suitable matrices. In particular, assuming an upstream control, we end up with a model of the form

$$h_i(k+1)=h_i(k)+\frac{T_{\mathrm{m}}}{c_i}(u_{i-1}(k-k_{\mathrm{d}i})-u_i(k)+d_i(k)+g_i(k)), \tag{3}$$

in which T_{m} denotes the sampling period (equal for all pools), c_i is the surface area, and $k_{\mathrm{d}i}$ is a time delay (in control steps) representing the time required for an inflow from upstream gate $i-1$ to influence the water level in pool i. Clearly, for $i=1$, the inflow is the flow from the head gate. Moreover, $d_i(k)$ denotes a water offtake from the canal due to a request made by the user and $g_i(k)$ is a known disturbance in the pool i due to for instance rainfall.

As mentioned earlier in the paper, local PI controllers are employed throughout the canal to control the water level immediately upstream of each gate. Taking that into consideration, each canal pool $i\in\{1,\ldots,N\}$ is described with the following discrete-time model

$$\begin{aligned} h_i(k+1) &= h_i(k)+\frac{T_{\mathrm{m}}}{c_i}(u_{i-1}(k-k_{\mathrm{d}i})-u_i(k)+d_i(k)+g_i(k)), \\ u_i(k) &= u_i(k-1)+K_{\mathrm{P}i}(e_i(k)-e_i(k-1))+K_{\mathrm{I}i}e_i(k), \qquad \text{(PI controller)} \\ u_0(k) &= Q_{\mathrm{S}}(k), \\ e_i(k) &= h_i(k)-h_i^{\mathrm{ref}}(k), \end{aligned} \tag{4}$$

where e_i denotes the deviation between the water level in Pool i and a given setpoint for that pool, h_i^{ref}, and Q_{S} denotes the inflow from the head gate. Note that in (4) the local PI controllers are already incorporated. In that sense, (4) represents the closed-loop dynamics of the pools $i=1,\ldots,N$ in terms of the local controllers u_i. However, these dynamics are subject to control inputs from the Coordinator as shown in due course.

Model Predictive Control

In this section we briefly recall the concept of Model Predictive Control (Maciejowski, 2002; Camacho and Bordons, 1999), which is used in this paper to develop the controller. Model Predictive Control, also known as Receding Horizon Control, is a very powerful tool due to, amongst others, its ability to take care of state and control input constraints and to deal with multivariable systems. It is a type of optimal controller that at each time step uses current measurements and the internal model of the plant to obtain state predictions $x(k+1|k),\dots,x(k+N_\mathrm{p}|k)$ for the following N_p steps. These predictions are then used to evaluate a given cost function $J(x(k+1|k),\dots,x(k+N_\mathrm{p}|k),u(k|k),\dots,u(k+N_\mathrm{p}-1|k))$. The objective of MPC is to find a suitable sequence of control actions over the whole prediction horizon $u(k|k),\dots,u(k+N_\mathrm{p}-1|k)$ minimizing the cost function. Here $x(k+j|k)$ denotes the state prediction for time $k+j$ obtained at time k and $u(k+j|k)$ denotes the optimal control found by the optimizer at time k to be applied at time $k+j$. Once the sequence of optimal controls over the prediction horizon is found, the first control action $u(k|k)$ from the sequence is applied to the plant and the process is repeated at next time step $k+1$ looking again N_p steps into the future and using new information as it comes along.

Time instant optimization

Time instant optimization is an approach to MPC that was first introduced in (De Schutter and De Moor, 1998) for optimal traffic control. For water systems the idea of optimizing time instants was primarily proposed for discontinuous on/off hydraulic structures in (Van Ekeren et al., 2011). Note that in classical MPC for on/off structures it needs to be decided at each step whether the structure should be switched on or off at every time from the current moment up to the time N_p steps ahead. Therefore, the problem results in N_p binary control variables and as a result the problem may turn out to be impracticable to solve numerically. An alternative is to decide how many on/off switches are allowed for the next N_p steps and optimize when the switching time instants are to occur. Hence, the optimization problem redefined using the rationale of TIO-MPC reduces the number of control variables thus making the problem more viable computationally.

HIERARCHICAL CONTROLLER DESIGN

In this section we introduce the hierarchical centralized controller to coordinate the local PI controllers and thus enhance the water delivery process. It is assumed that a single delivery request is described by its volume per second and time instants when the delivery should start and finish. For example, a request can be made for $0.1\mathrm{m}^3/\mathrm{s}$ to start in 1 hour and last for 30min. For the time being, to present our concept in a simplified way, we also assume that no overlapping of the requests of individual users is allowed.

The Coordinator coordinates the water deliveries to the users by controlling the water flow through the head gate as well as by manipulating the reference levels in individual canal pools at appropriate times when it is needed. In other words, the Coordinator provides the local controllers with a block-shaped setpoint profile: it finds a modified value of the setpoint and time instants when this modified value should be switched on and back off to return to the normal operating value of the setpoint. Examples of possible setpoint profiles are given in Figure 2. Importantly, because of the nature of the profile found by the Coordinator, the Coordinator only needs to communicate twice to each local site: once to provide the changed value of the setpoint at the time of change and later when the setpoint should return to the normal level. This is an essential feature because it implies that there is no need for frequent interference with the operation of local sites.

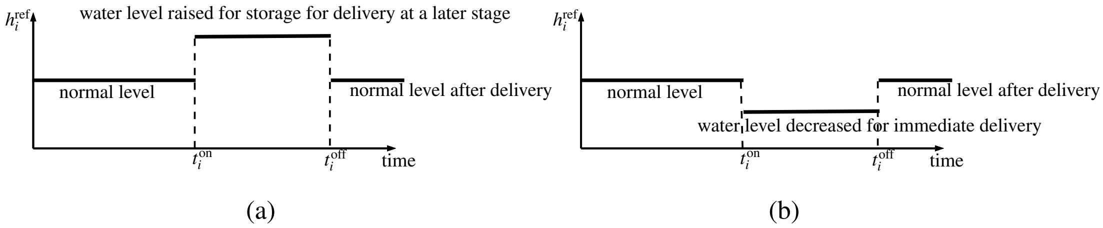

Figure 2. Possible setpoint profiles provided by the Coordinator.

Following this introduction, the control inputs to be found by the Coordinator are

$$\mathcal{U}_s = \begin{pmatrix} \widetilde{Q}_{\mathrm{S,\,demand},s} \\ H_s^{\mathrm{ref,\,delivery}} \\ T_s^{\mathrm{on}} \\ T_s^{\mathrm{off}} \end{pmatrix}. \tag{5}$$

Here, $s \in \mathbb{N}$ is the delivery counter, which is incremented every time the Coordinator is activated. Moreover, $\widetilde{Q}_{\mathrm{S,\,demand},s}$ denotes a profile of the extra flow from the head gate to the first pool for the whole prediction horizon $N_{\mathrm{p}} \in \mathbb{N}$ needed for delivery s, i.e.

$$\widetilde{Q}_{\mathrm{S,\,demand},s} = (Q_{\mathrm{S,\,demand},s}(0), \dots, Q_{\mathrm{S,\,demand},s}(N_{\mathrm{p}}-1))^T. \tag{6}$$

From $Q_{\mathrm{S,\,demand},s}(j)$, $j = 0, \dots, N_{\mathrm{p}}-1$, we can determine the overall flow from the head gate to be used in (4) as

$$Q_{\mathrm{S}}(k_{\mathrm{active},s} + jA_{\mathrm{c}} + i) = Q_{\mathrm{S,\,base}} + Q_{\mathrm{S,\,demand},s}(j), \tag{7}$$

where $Q_{\mathrm{S,\,base}}$ denotes the base flow in the canal and $i = 0, \dots, A_{\mathrm{c}}-1$. Here, we use $A_{\mathrm{c}} = T_{\mathrm{c}}/T_{\mathrm{m}} \in \mathbb{N}$, in which T_{c} is the length of the control cycle of the Coordinator, which is a multiple of the sampling time of the model T_{m}. Moreover, $k_{\mathrm{active},s} \in \mathbb{N}$ denotes the Coordinator's activation time step for the s^{th} delivery defined as

$$k_{\mathrm{active},s} = \left\lceil \frac{t_{\mathrm{active},s}}{T_{\mathrm{m}}} \right\rceil, \tag{8}$$

where $\lceil x \rceil$ denotes the ceiling function. By the above definition, $t_{\text{active},s} \leq k_{\text{active},s} T_{\text{m}}$, where $t_{\text{active},s}$ is the activation time of the Coordinator.

Further control inputs in (5) are

$$\begin{aligned} H_s^{\text{ref, delivery}} &= (h_{1,s}^{\text{ref, delivery}}, \ldots, h_{N,s}^{\text{ref, delivery}})^T, \\ T_s^{\text{on}} &= (t_{1,s}^{\text{on}}, \ldots, t_{N,s}^{\text{on}})^T, \\ T_s^{\text{off}} &= (t_{1,s}^{\text{off}}, \ldots, t_{N,s}^{\text{off}})^T, \end{aligned} \tag{9}$$

where $h_{i,s}^{\text{ref, delivery}} \in \mathbb{R}$, and where in the spirit of TIO-MPC, $t_{i,s}^{\text{on}} \in \mathbb{R}$ and $t_{i,s}^{\text{off}} \in \mathbb{R}$ are the switching time instants such that

$$h_i^{\text{ref}}(k) = \begin{cases} h_i^{\text{ref, normal}} & \text{if } k \leq k_{i,s}^{\text{on}} \text{ or } k \geq k_{i,s}^{\text{off}}, \\ h_{i,s}^{\text{ref, delivery}} & \text{otherwise,} \end{cases} \tag{10}$$

in which $k_{i,s}^{\text{on}}$ and $k_{i,s}^{\text{off}}$ are discrete-time equivalents of the continuous variables $t_{i,s}^{\text{on}}$ and $t_{i,s}^{\text{off}}$ given certain model sampling time T_{m}:

$$k_{i,s}^{\text{on}} = \left[\frac{t_{i,s}^{\text{on}}}{T_{\text{m}}} \right] \qquad \text{and} \qquad k_{i,s}^{\text{off}} = \left[\frac{t_{i,s}^{\text{off}}}{T_{\text{m}}} \right], \tag{11}$$

where $[x]$ denotes the value of x rounded to the nearest integer. Moreover, $h_i^{\text{ref, normal}}$ is the normal operating level of the setpoint in canal pool i.

In view of the above, the cost function that the Coordinator minimizes once triggered is

$$J_s \tag{12}$$

$$= \alpha \sum_{j=1}^{A_{\text{c}} N_{\text{p}}} \left(u_N \left(k_{\text{active},s} + j - 1 \right) \right) - Q_{\text{S, base}}) \tag{13}$$

$$+ \sum_{i=1}^{N} \sum_{j=1}^{A_{\text{c}} N_{\text{p}}} \left[\gamma_1 \left(\max(h_i \left(k_{\text{active},s} + j \right) - h_i^{\text{max,des}}, 0) \right)^2 \right. \tag{14}$$

$$\left. + \gamma_2 \left(\min(h_i \left(k_{\text{active},s} + j \right) + h_i^{\text{min,des}}, 0) \right)^2 \right] \tag{15}$$

$$+ \sum_{i=1}^{N} \sum_{j=1}^{A_{\text{c}} N_{\text{p}}} \beta \left(h_i \left(k_{\text{active},s} + j \right) - h_i^{\text{ref}} \left(k_{\text{active},s} + j \right) \right)^2 \tag{16}$$

$$+ \sum_{i=1}^{N} \mu \left(h_i^{\text{ref,normal}} - h_i^{\text{ref}} \left(k_{\text{active},s} + N_{\text{p}} A_{\text{c}} - 1 \right) \right)^2, \tag{17}$$

in which α, γ_1, γ_2, and β are positive weighting coefficients, and u_N denotes flow through gate N.

In the cost function J_s, the term (13) penalizes any deviations in flow through the last gate of the canal with respect to the value of the base flow $Q_{\text{S, base}}$ as it is required that the flow through the last gate to further parts of the canal behind the N pools should always be as close as possible to some given $Q_{\mathrm{S,base}}$. The terms (14) and (15) penalizes control actions resulting in the water levels violating the upper and lower bounds. Note that the values $h_i^{\mathrm{max,des}}$ and $h_i^{\mathrm{min,des}}$ denote desired operating upper and lower bounds; actual physical bounds stemming from canal geometry are less strict and are imposed as hard constraints (see (18) below). Furthermore, the term (16) penalizes any deviations of the water levels from their desired levels and the term (17) poses a penalty on the final value of the reference levels so that after the delivery has been finished, the reference levels return to normal.

The hard constraints are as follows

$$h_i^{\min} \le h_i(\ell) \le h_i^{\max}, \qquad (\ell = k_{\mathrm{active},s}+1,\dots,k_{\mathrm{active},s}+N_{\mathrm{p}}A_{\mathrm{c}}), \tag{18}$$

$$h_i^{\min} \le h_{i,s}^{\mathrm{ref,delivery}} \le h_i^{\max}, \tag{19}$$

$$t_{i,s}^{\mathrm{off}} \ge t_{i,s}^{\mathrm{on}} + T_{\mathrm{m}}, \tag{20}$$

$$t_{i,s}^{\mathrm{on}} \ge k_{\mathrm{active},s} T_{\mathrm{m}} \tag{21}$$

$$Q_{\mathrm{S,demand},s}(n) \ge 0, \qquad (n = 0,\dots,N_{\mathrm{p}}-1), \tag{22}$$

$$0 \le Q_{\mathrm{S}}(m) \le Q_{\mathrm{capacity}}, \qquad (m = k_{\mathrm{active},s},\dots,k_{\mathrm{active},s}+N_{\mathrm{p}}A_{\mathrm{c}}-1), \tag{23}$$

for all $i \in \{1,\dots,N\}$. The meaning of constraints (18) and (19) is that the water levels as well as the modified setpoints chosen by the Coordinator need to remain within safety bounds to avoid the risk of flooding or drying out the canal. Constraints (20) and (21) limit the possible choices of the switching time instants in that the first switch must not occur before the time $k_{\mathrm{active},s}T_{\mathrm{m}}$ when the Coordinator is activated for the s^{th} delivery and the second switch must occur strictly after the first one. In addition, the last two constraints, (22) and (23), give restrictions on possible values of $Q_{\mathrm{S,demand},s}$. In particular, this extra flow needs to be nonnegative and together with the base flow, the overall flow must not exceed the maximum capacity of the head gate. Furthermore, Equation (10) can also be treated as a hard constraint defining that in between the switches the setpoint may not change.

Since the Coordinator only works in response to requested offtakes, its triggering condition can be described as follows. The Coordinator optimizes (12) subject to constraints (10) and (18)–(23) when it learns about a new delivery. In particular, we assume that the Coordinator remains inactive until the trigger switches it on (when a delivery s is requested). Then the Coordinator finds suitable values $\mathcal{U}_s$, i.e. a profile $\widetilde{Q}_{\mathrm{S,demand},s}$, the switching time instants T_s^{on} and T_s^{off} and the modified setpoints $H_s^{\mathrm{ref,delivery}}$. Assuming a long enough prediction horizon N_{p}, no overlapping requests, and a strictly deterministic case, the Coordinator performs the optimization only once per delivery. Therefore, after finding the suitable control action $\mathcal{U}_s$ for delivery s, the Coordinator is switched off until a new delivery $s+1$ comes along. Note that we also require that the time between any two activations of the Coordinator is at least $T_{\min}$ so the changes to the local settings are not too often. The time $T_{\min}$ represents the minimal reactivation time of the Coordinator and is a multiple of the sampling time T_{m}. It may be viewed as a design parameter that can be chosen according to the requirements of a particular system. It is also required that before the Coordinator can

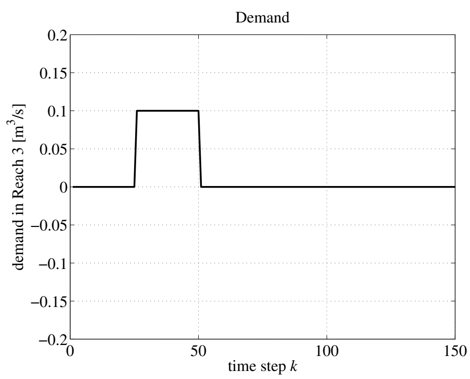

Figure 3. Demand profile in Pool 3.

be reactivated for delivery $s+1$, all setpoints changed for delivery s need to return back to their normal levels.

The functioning of the system governed by the Coordinator for $t \in (0, T_\mathrm{f})$ can be illustrated by the following algorithm:

(1) s=0,
(2) k=1,
(3) if a new delivery is requested, go to (4), otherwise go to (6),
(4) if k $\geq k_{\mathrm{active},s-1} + T_{\min}/T_\mathrm{m}$ and for all i, k $\geq k_{i,s-1}^{\mathrm{off}}$,
set s=s+1 and go to (5); otherwise go to (6),
(5) solve the MPC problem,
(6) k=k+1,
(7) wait until t=kT_m,
(8) if t<T_f go to (3), otherwise stop.

SIMULATION STUDY

This section illustrates the method introduced in the paper by simulations. For the sake of clarity of presentation, we use a canal consisting of 5 pools. The sampling period of the model is $T_\mathrm{m} = 1\,\mathrm{min}$ and the sampling time of the Coordinator is $T_\mathrm{c} = 5\,\mathrm{min}$. For all pools we use $K_{\mathrm{P}i} = 3.6$ and $K_{\mathrm{I}i} = 0.2$, chosen by fine tuning to be the proportional and integral gains of the upstream PI controllers. Moreover, the weighting coefficients used to evaluate the cost function (12) are $\alpha = 10$, $\beta = 5$, $\gamma_1 = \gamma_2 = 1$, and $\mu = 3$.

The surface areas of the pools are (in square meters): 397, 653, 503, 1530, and 1614. Furthermore, the delays in all pools before an inflow from a pool immediately upstream affects water levels at the end of the pool are: 7, 10, 3, 1, and 9 steps, respectively. The prediction horizon in our simulations is $N_\mathrm{p} = 120$ simulation steps, which is equivalent to 24 control cycles of the Coordinator. This number is chosen to enable the Coordinator to verify how its actions would affect the whole canal given internal delays in each canal pool.

We start the simulation from steady state in which water height in all pools is equal to $h_i(0) = -0.6\,\mathrm{m}$ and the flow is $u_i(0) = 1.5\mathrm{m}^3/\mathrm{s}$. The initial flow from the head gate is $1.5\,\mathrm{m}^3/\mathrm{s}$ and that base flow should be maintained throughout the simulation. Note that water levels are given as negative numbers since the coordinate frame used in the simulations is assumed to be located at the ground level. Thus, water levels in the canal with respect to that coordinate frame are negative numbers.

The simple demand chosen to illustrate how the hierarchical centralized controller works is such that from $k = 25$ until $k = 50$ there is an outflow from Pool 3 of $0.1\mathrm{m}^3/\mathrm{s}$, see Figure 3. Note that in the classical way when only local PI controllers operate, an offtake in Pool 3 would require an announcement at least 20 steps before the actual offtake can be done. Recall that with only local PI controllers, water is delivered to the user by adding the required amount of water from the head gate to the first pool and waiting for local controllers to transport it to Pool 3. However, in the hierarchical control settings the Coordinator only finds out about the delivery requested 5 steps ahead (at $k = 20$) and yet, as shown below, it is still able to realize it. In the future, we will also formally consider the extreme case when no notice period is required and offtakes can be announced and immediately realized, e.g. in the case of emergency etc.

Simulation results are given in Figures 4–6. In particular, in Figure 4 we present the profile of flow from the head gate as found by the Coordinator. In Figure 5 we show how the setpoints are changed by the Coordinator for all pools. We see that the setpoints of Pools 1 and 2 are lowered before the offtake starts soon after the Coordinator is activated (i.e. just after $k = 20$). That means that water can be released from these pools and made available for the delivery in Pool 3. This is an important observation because since it takes 20 steps to deliver water from the head gate to Pool 3, it would be impossible to merely use water from the head gate for the delivery since there is not enough time to transport that water to Pool 3. However, by the changed setpoints in Pool 1 and Pool 2, water can be delivered timely without disturbing the rest of the canal. We also see that the setpoint in Pool 3 is increased a little, which allows to store water released from Pool 1 in Pool 2 in Pool 3 for a delivery in that pool a few moments later. Moreover, notice that the setpoints also

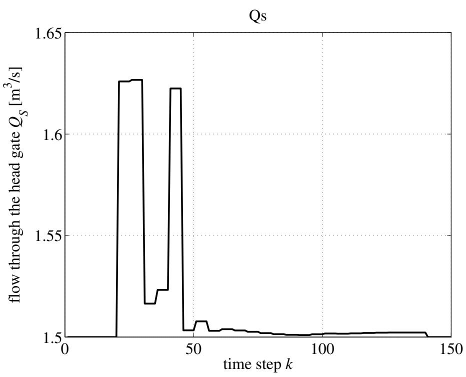

Figure 4. Inflow from the head gate.

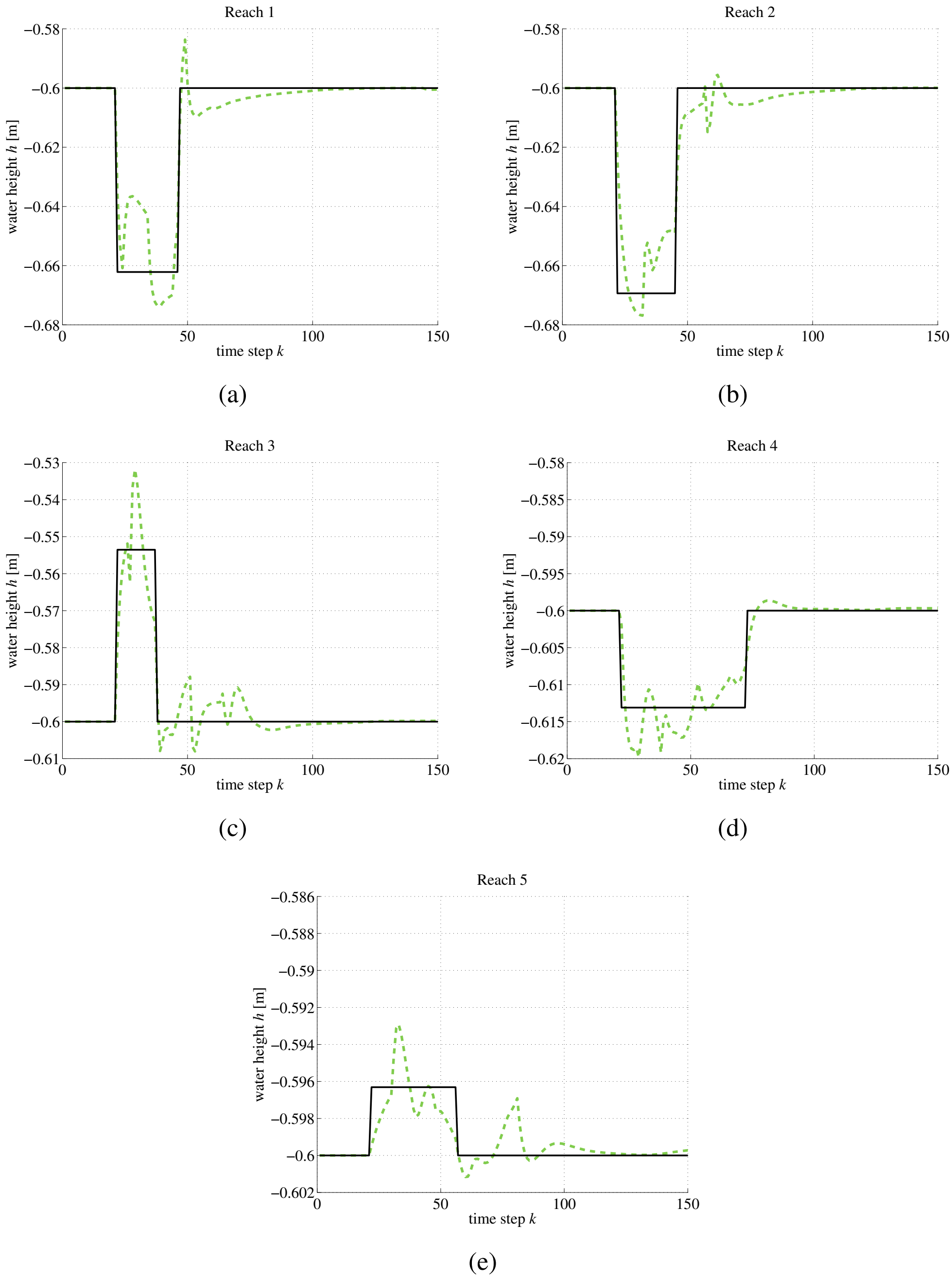

Figure 5. Water levels (dashed line) and setpoint (solid line) for each pool.

change in Pools 4 and 5, yet when looking at the scale, we immediately see that this change is minor. While there are no deliveries in these pools, the changes can be explained by the objective of the Coordinator to maintain the flow through the fifth gate of the canal as close as possible to the given base flow. Hence, by also modifying the setpoints in Pools 4 and 5, the Coordinator has more means to meet this objective. Indeed, Figure 6 shows that the deviation in flow through the fifth gate is minor, thus demonstrating that the delivery is accomplished with minimal disruption to the

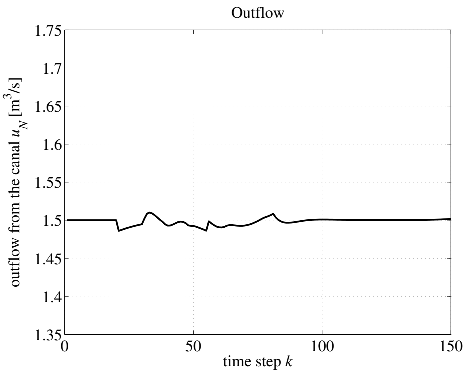

Figure 6. Outflow from the fifth pool.

remaining part of the canal and confirming good performance yielded by the Coordinator.

CONCLUSIONS

We have showed a new hierarchical control method to tackle water deliveries to farmers through an irrigation canal. The method is simple because it mainly relies on the application of local PI controllers at each gate to control water levels upstream of the gate. However, to boost the performance of the local PI controllers and allow faster deliveries, we have proposed a higher-layer centralized controller – the Coordinator – whose job is to coordinate the local controllers and hence enable shorter times before an offtake can be made after it has been announced. To that end, Time Instant Optimization was used within the framework of Model Predictive Control. The findings of the paper are illustrated by simulation results demonstrating the effectiveness of the method.

Because the model of the canal used is linear, in our work we assume a proportional relation between water level and volume in the pools. As, in reality, volume is a nonlinear function of the water level, there may be corrections necessary to the imposed inflow at the head gate. This can be done by using Volume Compensation as in (Bautista and Clemmens, 2005) or by employing a nonlinear internal model in MPC in future applications. Our future work will also include tests on a more accurate model of an actual irrigation canal and its local controllers, possibly accounting for nonlinearities, measurement noise, and unmodeled dynamics. Moreover, we will also extend the method to allow for overlapping deliveries and analyze the performance of sudden schedule changes (without any lead time) to make the method more universal and applicable in the field.

REFERENCES

K. J. Åström and T. Hägglund. *PID Controllers: Theory, Design, and Tuning*. Instrument Society of America, Research Triangle Park, NC, 1995.

E. Bautista and A. Clemmens. Volume compensation method for routing irrigation canal demand changes. *Journal of Irrigation and Drainage Engineering*, 131(6): 494–503, 2005.

E. Camacho and C. Bordons. *Model predictive control*. Springer, Berlin Heidelberg, 1999.

V. T. Chow. *Open-Channel Hydraulics*. McGraw-Hill Civil Engineering. McGraw-Hill, London, 1959.

B. De Schutter and B. De Moor. Optimal traffic light control for a single intersection. *European Journal of Control*, 4(3):260–276, 1998.

X. Litrico and V. Fromion. Tuning of robust distant downstream PI controllers for an irrigation canal pool. I: Theory. *Journal of Irrigation and Drainage Engineering*, 132(4):359–368, 2006.

X. Litrico, V. Fromion, J.-P. Baume, and M. Rijo. Modelling and pi controller design for an irrigation canal. In *Proceedings of the 2003 European Control Conference, Cambridge, UK*, 2003.

X. Litrico, P.-O. Malaterre, J.-P. Baume, P. Vion, and J. Ribot-Bruno. Automatic tuning of PI controllers for an irrigation canal pool. *Journal of Irrigation and Drainage Engineering-ASCE*, 133:27–37, 2007.

J. M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, Essex, England, 2002.

P.-O. Malaterre. Control of irrigation canals: why and how? In *International Workshop on Numerical Modelling of Hydrodynamics for Water Resources*, pages 271–293, 2007.

P.-O. Malaterre and J.-P. Baume. Modeling and regulation of irrigation canals: existing applications and ongoing researches. In *IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 3850–3855, 1998.

R. R. Negenborn, S. Leirens, B. De Schutter, and J. Hellendoorn. Supervisory nonlinear MPC for emergency voltage control using pattern search. *Control Engineering Practice*, 17(7):841–848, 2009a.

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

V. Rutz, C. Ruiz, and L. Ramirez. Predictive control in irrigation canal operation. In *Proceedings of the 2011 IEEE International Conference on Systems, Man, and Cybernetics*, volume 4, pages 3897–3901, 1998.

J. Schuurmans. *Control of Water Levels in Open-Channels*. PhD thesis, Delft University of Technology, The Netherlands, 1997.

C. Sepúlveda Toepfer. *Instrumentation, Model Identification and Control of an Experimental Irrigation Canal*. PhD thesis, Universitat Politécnica de Catalunya, Spain, 2007.

H. van Ekeren, R. R. Negenborn, P. J. van Overloop, and B. De Schutter. Hybrid model predictive control using time-instant optimization for the Rhine-Meuse delta. In *Proceedings of the 2011 IEEE International Conference on Networking, Sensing and Control*, pages 216–221, 2011.

P.-J. van Overloop. *Model Predictive Control on Open Water Systems*. PhD thesis, Delft University of Technology, The Netherlands, 2006.

P.-J. van Overloop, J. Schuurmans, R. Brouwer, and C. Burt. Multiple-model optimization of proportional integral controllers on canals. *Journal of Irrigation and Drainage Engineering*, 131(2):190–196, 2005.

D. D. Šiljak. *Decentralized Control of Complex Systems*. Academin Press, Boston, Massachusetts, 1991.

E. Weyer. Control of irrigation channels. *IEEE Transactions on Control Systems Technology*, 16(4):664–675, 2008.

A. Zafra-Cabeza, J. M. Maestre, M. A. Ridao, E. F. Camacho, and L. Sanchez. Hierarchical distributed model predictive control for risk mitigation: An irrigation canal case study. In *Proceedings of the 2011 American Control Conference*, pages 3172–3177, 2011.