Technical report

Coordinated model predictive reach control for irrigation canals

R.R. Negenborn, P.J. van Overloop, B. De Schutter

If you want to cite this report, please use the following reference instead:

R.R. Negenborn, P.J. van Overloop, B. De Schutter. Coordinated model predictive reach control for irrigation canals. In *Proceedings of the European Control Conference 2009* (ECC'09), Budapest, Hungary, pp. 1420-1425, August 2009.

Delft University of Technology, Delft, The Netherlands

Coordinated Distributed Model Predictive Reach Control of Irrigation Canals

Rudy R. Negenborn, Peter-Jules van Overloop, and Bart De Schutter

***Abstract*— Irrigation canals are large-scale systems, covering vast geographical areas, and consisting of many interconnected canal reaches that interact with control structures such as pumps and gates. The control of such irrigation canals is usually done in a manual way, in which a human operator travels along the irrigation canal to adjust the settings of the gates and pumps in order to obtain a desired water level. In this paper we discuss how distributed model predictive control (MPC) can be applied to determine autonomously what the settings of these control structures should be. In particular, we propose the application of a distributed MPC scheme for control of the West-M irrigation canal in Arizona. We present a linearized model representing the dynamics of the canal, we propose a distributed MPC scheme that uses this model as a prediction model, and we illustrate the performance of the scheme in simulation studies on a nonlinear simulation model of the canal.**

***Index Terms*— Distributed control, model predictive control, large-scale systems, irrigation canals.**

I. INTRODUCTION

Irrigation canals are used for transporting water from source nodes, such as lakes, large rivers, etc., to sink nodes, such as small rivers and pipes that transport water to agricultural fields of farmers. Irrigation canals consist of many connected canal reaches, the inflow or outflow of which can be controlled by adjusting structures such as overshot or undershot gates, activating pumps, filling or draining water reservoirs, and controlled flooding of water meadows or of emergency water storage areas [1].

In the near future the importance of efficient and reliable irrigation management systems for delivering water to users will keep on increasing, among others due to the effects of global warming (more heavy rain during the spring season, but possibly also drier summers).

Due to the large scale of irrigation networks, control of such networks in general cannot be done in a centralized way, in which from a single location measurements from the whole system are collected and actions for the whole system are determined. Instead, control is typically decentralized over several local control bodies, each controlling a particular part of the network [2]. Currently, coordination between such decentralized local control bodies is either non-existing, or takes place at a very slow time scale, i.e., years, in the form of agreements laid out in contracts.

In order to improve the operation of irrigation systems the controllers of different parts of the irrigation network should cooperate and coordinate their local water management actions on a daily, hourly, or even minute basis, such that predictions or forecasts of expected water consumption, future rain fall, future droughts, future arrival of increased water flow via rivers, etc. can be taken into account using various weather and hydrological sensors, and prediction models. Model predictive control (MPC) is a control strategy that enables such a control framework.

In [3] an MPC scheme is proposed that is used by a single controller to determine in a centralized way the set-points for local flow controllers in an irrigation canal. In [4] we made a first attempt to implement a distributed MPC scheme to take over this task. In that paper, a highly simplified model of the so-called West-M irrigation canal is studied. The assumption is made that local PI controllers are present to control the control gates and that constraints on the minimum and maximum water levels and on the minimum and maximum gate positions do not have to be taken into account. In [4], MPC controllers are then designed for each individual control structure. In addition, simulation studies are carried out only on a linearized model of the system.

In this paper, we make a next step for obtaining a distributed MPC controller that can be used in practice. We consider control of a validated, nonlinear model of the West-M canal using a distributed MPC scheme. Hereby, it is not assumed that local PI controllers are present. Instead, the changes in the positions of gates are determined directly. In addition, operational constraints on the water levels and gate positions are taken into account. Moreover, we design MPC controllers for controlling parts of an irrigation canal consisting of several, instead of single, canal reaches and control structures. Furthermore, we perform simulation studies on the nonlinear, instead of a linear, system.

This paper is organized as follows. In Section II, we briefly outline the distributed MPC scheme that we employ. In Section III, we discuss a linearized model of the dynamics of the West-M irrigation canal, and set up the distributed MPC control scheme for control of this system. In Section IV, we illustrate the potential of the proposed approach through simulation studies on a nonlinear simulation model of the canal. Section V concludes the paper and contains directions for future research.

R.R. Negenborn and B. De Schutter are with the Delft Center for Systems and Control of Delft University of Technology, Mekelweg 2, 2628 CD Delft, The Netherlands, e-mail: r.r.negenborn@tudelft.nl, b.deschutter@dcsc.tudelft.nl. B. De Schutter is also with the Marine and Transport Technology department of Delft University of Technology. Peter-Jules van Overloop is with the department of Water Management of Delft University of Technology, Stevinweg 1, 2628 CN Delft, The Netherlands, e-mail: p.j.a.t.m.vanoverloop@tudelft.nl.

II. DISTRIBUTED MODEL PREDICTIVE CONTROL

In distributed MPC the control of a system is divided over several controllers. An individual controller on the one hand obtains measurements from and determines actions for its part of the network, and on the other hand communicates with other controllers in order to obtain coordination and to improve the overall network performance. To actually determine which actions to take, each controller uses MPC.

In [5] we have proposed a distributed MPC scheme for control of general transportation networks. Irrigation canals are a particular type of transportation networks, and therefore this scheme is also suitable for distributed control of irrigation canals. Below, we briefly outline the scheme and the assumptions made on the system under control.

A. Dynamics

Consider a network divided into n subnetworks. It is assumed that the dynamics of subnetwork $i \in \{1, \ldots, n\}$ are given by a deterministic linear discrete-time time-invariant model (possibly obtained after symbolic or numerical linearization of a nonlinear model in combination with discretization):

$$\begin{aligned} \mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_{1,i}\mathbf{u}_i(k) \\ + \mathbf{B}_{2,i}\mathbf{d}_i(k) + \mathbf{B}_{3,i}\mathbf{v}_i(k) \end{aligned} \quad (1)$$

$$\begin{aligned} \mathbf{y}_i(k) = \mathbf{C}_i \mathbf{x}_i(k) + \mathbf{D}_{1,i}\mathbf{u}_i(k) \\ + \mathbf{D}_{2,i}\mathbf{d}_i(k) + \mathbf{D}_{3,i}\mathbf{v}_i(k), \end{aligned} \quad (2)$$

where at control step k, for subnetwork i, $\mathbf{x}_i(k) \in \mathbb{R}^{n_{\mathbf{x}_i}}$ are the local states, $\mathbf{u}_i(k) \in \mathbb{R}^{n_{\mathbf{u}_i}}$ are the local inputs, $\mathbf{d}_i(k) \in \mathbb{R}^{n_{\mathbf{d}_i}}$ are the local known or measureable exogenous inputs, $\mathbf{y}_i(k) \in \mathbb{R}^{n_{\mathbf{y}_i}}$ are the local outputs, $\mathbf{v}_i(k) \in \mathbb{R}^{n_{\mathbf{v}_i}}$ are the remaining variables influencing the local dynamical states and outputs, and $\mathbf{A}_i \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{x}_i}}$, $\mathbf{B}_{1,i} \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{u}_i}}$, $\mathbf{B}_{2,i} \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{d}_i}}$, $\mathbf{B}_{3,i} \in \mathbb{R}^{n_{\mathbf{x}_i} \times n_{\mathbf{v}_i}}$, $\mathbf{C}_i \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{x}_i}}$, $\mathbf{D}_{1,i} \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{u}_i}}$, $\mathbf{D}_{2,i} \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{d}_i}}$, $\mathbf{D}_{3,i} \in \mathbb{R}^{n_{\mathbf{y}_i} \times n_{\mathbf{v}_i}}$ determine how the different variables influence the local states and outputs of subnetwork i. The $\mathbf{v}_i(k)$ variables represent the influence of other subnetworks on subnetwork i, and are therefore equal to some of the variables of models representing dynamics of neighboring subnetworks. So-called interconnecting input variables $\mathbf{w}_{\text{in},ji}(k) \in \mathbb{R}^{n_{\mathbf{w}_{\text{in},ji}}}$ are the variables of subnetwork i that are influenced by subnetwork j, i.e., a selection of $\mathbf{v}_i(k)$. So-called interconnecting output variables $\mathbf{w}_{\text{out},ji}(k) \in \mathbb{R}^{n_{\mathbf{w}_{\text{out},ji}}}$ are the variables of subnetwork i that influence a neighboring subnetwork j, i.e., a selection of $\mathbf{x}_i(k)$, $\mathbf{u}_i(k)$, and $\mathbf{y}_i(k)$. Fig. 1 illustrates the relations between the variables of the models of two subnetworks.

Let subsystem i be connected to m_i *neighboring* subsystems. Let the set of indices of the m_i subsystems connected to subsystem i be denoted by the neighbors set $\mathcal{N}_i = \{j_{i,1}, \ldots, j_{i,m_i}\}$. Define the interconnecting inputs and outputs for the control problem of controller i at control step k as:

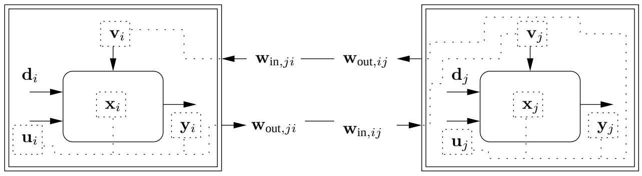

Fig. 1. Illustration of the relation between the models and variables of two subnetworks i and j.

$$\mathbf{w}_{\text{in},i}(k) = \mathbf{v}_i(k) \quad (3)$$

$$\mathbf{w}_{\text{out},i}(k) = \mathbf{K}_i \begin{bmatrix} \mathbf{x}_i^{\mathrm{T}}(k) & \mathbf{u}_i^{\mathrm{T}}(k) & \mathbf{y}_i^{\mathrm{T}}(k) \end{bmatrix}^{\mathrm{T}}, \quad (4)$$

where $\mathbf{K}_i$ is an interconnecting output selection matrix that contains zeros everywhere, except for a single 1 per row corresponding to a local variable that corresponds to an interconnecting output variable. The variables $\mathbf{w}_{\text{in},i}(k)$, $\mathbf{w}_{\text{out},i}(k)$ are partitioned such that:

$$\mathbf{w}_{\text{in},i}(k) = \left[\mathbf{w}_{\text{in},j_{i,1}i}^{\mathrm{T}}(k), \ldots, \mathbf{w}_{\text{in},j_{i,m_i}i}^{\mathrm{T}}(k)\right]^{\mathrm{T}} \quad (5)$$

$$\mathbf{w}_{\text{out},i}(k) = \left[\mathbf{w}_{\text{out},j_{i,1}i}^{\mathrm{T}}(k), \ldots, \mathbf{w}_{\text{out},j_{i,m_i}i}^{\mathrm{T}}(k)\right]^{\mathrm{T}}. \quad (6)$$

The interconnecting inputs to the control problem of controller i with respect to controller j must be equal to the interconnecting outputs from the control problem of controller j with respect to controller i, since the variables of both control problems model the same quantity. For controller i this thus gives rise to the following *interconnecting constraints*:

$$\mathbf{w}_{\text{in},ji}(k) = \mathbf{w}_{\text{out},ij}(k) \quad (7)$$

$$\mathbf{w}_{\text{out},ji}(k) = \mathbf{w}_{\text{in},ij}(k), \quad (8)$$

for all $j \in \mathcal{N}_i$.

B. Assumptions

It is assumed that each of the subnetworks $i \in \{1, \ldots, n\}$ is controlled by a control controller i that:

- has a prediction model of the form (1)–(2) of the dynamics of subnetwork i;
- can measure or estimate the state $\mathbf{x}_i(k)$ of its subnetwork;
- can estimate exogenous inputs $\mathbf{d}_i(k+l)$ of its subnetwork over a certain horizon of length N, for $l = \{0, \ldots, N-1\}$;
- can communicate with neighboring controllers.

C. Control objectives

It is assumed that the controllers are cooperative, meaning that the individual controllers strive for the best overall network performance. In addition, it is assumed that the objectives of the controllers can be represented by convex functions $J_{\text{local},i}$, for $i \in \{1, \ldots, n\}$, which are typically linear or quadratic. Such functions are commonly encountered, in particular for systems that can be represented by (1)–(2).

D. Distributed MPC scheme

The distributed MPC scheme that we employ comprises at control step k the following steps:

1) For $i = 1, \ldots, n$, controller i makes a measurement of the current state of the subnetwork $\mathbf{x}_i(k)$ and estimates the expected exogenous inputs $\mathbf{d}_i(k+l)$, for $l = 0, \ldots, N-1$.
2) The controllers cooperatively solve their control problems in the following serial iterative way[1]:
   a) Set the iteration counter s to 1 and initialize the Lagrange multipliers $\tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji}(k)$, $\tilde{\boldsymbol{\lambda}}^{(s)}_{\text{out},ij}(k)$ arbitrarily.
   b) For $i = 1, \ldots, n$, one controller i after another determines $\tilde{\mathbf{x}}^{(s)}_i(k+1)$, $\tilde{\mathbf{u}}^{(s)}_i(k)$, $\tilde{\mathbf{w}}^{(s)}_{\text{in},ji}(k)$, $\tilde{\mathbf{w}}^{(s)}_{\text{out},ji}(k)$ as solutions of the following optimization problem:

$$\min \; J_{\text{local},i}\left(\tilde{\mathbf{x}}_i(k+1), \tilde{\mathbf{u}}_i(k), \tilde{\mathbf{y}}_i(k)\right) + \sum_{j \in \mathcal{N}_i} J^{(s)}_{\text{inter},i}\left(\tilde{\mathbf{w}}_{\text{in},ji}(k), \tilde{\mathbf{w}}_{\text{out},ji}(k)\right), \quad (9)$$

   subject to the local dynamics (1)–(2) and (3)–(4) of subsystem i over the horizon, the current state $\mathbf{x}_i(k)$, and the known exogenous inputs $\tilde{\mathbf{d}}_i(k)$. The additional performance criterion $J_{\text{inter},i}$ in (9) at iteration s is defined as

$$J^{(s)}_{\text{inter},i}\left(\tilde{\mathbf{w}}_{\text{in},ji}(k), \tilde{\mathbf{w}}_{\text{out},ji}(k)\right) = \begin{bmatrix} \tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji}(k) \\ -\tilde{\boldsymbol{\lambda}}^{(s)}_{\text{out},ij}(k) \end{bmatrix}^{\mathrm{T}} \begin{bmatrix} \tilde{\mathbf{w}}_{\text{in},ji}(k) \\ \tilde{\mathbf{w}}_{\text{out},ji}(k) \end{bmatrix} + \frac{\gamma_{\text{c}}}{2} \left\| \begin{bmatrix} \tilde{\mathbf{w}}_{\text{in,prev},ij}(k) - \tilde{\mathbf{w}}_{\text{out},ji}(k) \\ \tilde{\mathbf{w}}_{\text{out,prev},ij}(k) - \tilde{\mathbf{w}}_{\text{in},ji}(k) \end{bmatrix} \right\|_2^2,$$

   where $\|\mathbf{a}\|_2$ is the 2-norm of vector $\mathbf{a}$. Furthermore, $\tilde{\mathbf{w}}_{\text{in,prev},ij}(k) = \tilde{\mathbf{w}}^{(s)}_{\text{in},ij}(k)$ and $\tilde{\mathbf{w}}_{\text{out,prev},ij}(k) = \tilde{\mathbf{w}}^{(s)}_{\text{out},ij}(k)$ is the information computed at the current iteration s for each controller $j \in \mathcal{N}_i$ that has solved its problem *before* controller i in the *current* iteration s. In addition, $\tilde{\mathbf{w}}_{\text{in,prev},ij}(k) = \tilde{\mathbf{w}}^{(s-1)}_{\text{in},ij}(k)$ and $\tilde{\mathbf{w}}_{\text{out,prev},ij}(k) = \tilde{\mathbf{w}}^{(s-1)}_{\text{out},ij}(k)$ is the information computed at the *previous* iteration $s-1$ for the other controllers. The constant γ_{c} is a positive scalar that penalizes the deviation from the interconnecting variable iterates that were computed by the controllers before controller i in the current iteration and by the other controllers during the last iteration. The results $\tilde{\mathbf{w}}^{(s)}_{\text{in},ji}(k)$ and $\tilde{\mathbf{w}}^{(s)}_{\text{out},ji}(k)$ of the optimization are sent to controller j.
   c) Update the Lagrange multipliers,

$$\tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in},ji}(k) = \tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji}(k) + \gamma_{\text{c}}\left(\tilde{\mathbf{w}}^{(s)}_{\text{in},ji}(k) - \tilde{\mathbf{w}}^{(s)}_{\text{out},ij}(k)\right). \quad (10)$$

   Send $\tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in},ji}(k)$ to controller j and receive the multipliers from controller j to be used as $\tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{out},ij}(k)$.
   d) Move on to the next iteration $s+1$ and repeat steps 2b–2c. The iterations stop when the following stopping condition is satisfied:

$$\left\| \begin{bmatrix} \tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in,err},j_{1,1}1}(k) \\ \vdots \\ \tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in,err},j_{n,m_n}n}(k) \end{bmatrix} \right\|_\infty \leq \gamma_\epsilon, \quad (11)$$

   with $\tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in,err},ji}(k) = \tilde{\boldsymbol{\lambda}}^{(s+1)}_{\text{in},ji}(k) - \tilde{\boldsymbol{\lambda}}^{(s)}_{\text{in},ji}(k)$, and where γ_ϵ is a small positive scalar and $\|\cdot\|_\infty$ denotes the infinity norm.
3) The controllers implement the actions until the beginning of the next control step.

Under the assumptions that we have made on the objective functions and prediction models the solution of this scheme converges to the solution that a centralized MPC controller would have obtained for a sufficiently small γ_ϵ, see [5].

In the next section we discuss how the presented approach can be used for controlling irrigation canals.

III. Control of an Irrigation Canal

Let an irrigation canal be controlled by n controllers. Each controllers controls the water levels and control structures in several connected canal reaches. Let there be m canal reaches. Let the set of canal reaches that controller $i \in \{1, \ldots, n\}$ controls be denoted by $\mathcal{R}_i$. Below we describe for a particular controller the dynamics of the canal reaches it considers, the operational constraints it has to take into account, and the formulation of its control goals.

A. Subnetwork dynamics

The subnetwork of controller i consists of several interconnected canal reaches that are usually separated by control structures, such as undershot gates. Next, we model these components.

1) A single canal reach: The dynamics of canal reaches can be described in detail using a system of hyperbolic partial differential equations called the Saint Venant equations [6]. Although a model obtained using such a detailed representation is desirable for simulation, for control this high level of detail is usually not necessary and in addition undesired for computational reasons. Therefore, instead of representing the reach dynamics with the Saint Venant equations, we employ the integrator delay model [7], [1], similarly as in [3]. This model has shown to adequately capture relevant dynamics [7], and it reduces computations required for simulation of the dynamics (and consequently model-based optimization) significantly.

The integrator delay model is a linear discrete-time model, which models how the water level in the canal changes over time. Let time be discretized into control steps $k \in \mathbb{N}_0$ (where $\mathbb{N}_0$ are the positive natural numbers) and let the continuous time between two control steps k and $k+1$

[1] The tilde notation is used to represent variables over the prediction horizon. E.g., $\tilde{\mathbf{u}}_i(k) = [\mathbf{u}_i(k)^{\mathrm{T}}, \ldots, \mathbf{u}_i(k+N-1)^{\mathrm{T}}]^{\mathrm{T}}$.

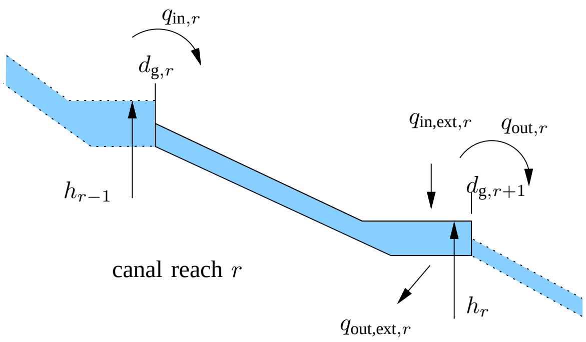

Fig. 2. Illustration of canal reach r and its associated variables.

correspond to $T_{\text{c}} \in \mathbb{R}^+$ (s) (where $\mathbb{R}^+$ are the positive real numbers). Each canal reach is considered to have an inflow from an upstream canal reach as illustrated in Figure 2. Let this inflow into reach r be given by $q_{\text{in},r}(k) \in \mathbb{R}^+$ (m^3/s). A canal reach has an outflow to a downstream canal reach. Let $q_{\text{out},r}(k) \in \mathbb{R}^+$ (m^3/s) denote this outflow. In addition to this inflow and outflow due to upstream and downstream canal reaches there can be additional local inflow (e.g., due to rainfall) and outflow (e.g., due to outflow caused by farmers). Let such inflow be represented by $q_{\text{ext,in},r}(k) \in \mathbb{R}^+$ (m^3/s) and such outflow by $q_{\text{ext,out},r}(k) \in \mathbb{R}^+$ (m^3/s). The inflow $q_{\text{ext,in},r}(k)$ and outflow $q_{\text{ext,out},r}(k)$ are assumed to be known or predicted accurately in advance.

Depending on how the inflows and outflows change over time, the levels of the water in reaches will change. Instead of considering the levels of the water at each location in the reaches, the integrator delay model only considers the level $h_r(k) \in \mathbb{R}^+$ (m) of the water at the downstream end of a reach r, since this is usually the place where offtakes are located. In addition to the amount of inflow and outflow, also the surface of the reach influences how much the level of the water will change. Let $e_r(k) \in \mathbb{R}$ (m) denote the deviation of the level of the water in canal reach r from a given reference water level for that canal reach, i.e., $e_r(k) = h_r(k) - h_{\text{ref},r}$, and let the surface of reach r be $c_r \in \mathbb{R}^+$ (m^2). It takes some time for a change in the inflow of reach r to result in a change of the water level at the downstream end of the reach. Let this delay be $k_{\text{d},r} \in \mathbb{N}_0$ control steps for reach r.

Using the variables defined above, the model describing how the level of the water in a single canal reach changes from one control step k to the next control step $k+1$ is given by:

$$e_r(k+1) = e_r(k) + \frac{T_{\text{c}}}{c_r} q_{\text{in},r}(k - k_{\text{d},r}) - \frac{T_{\text{c}}}{c_r} q_{\text{out},r}(k) + \frac{T_{\text{c}}}{c_r} q_{\text{ext,in},r}(k) - \frac{T_{\text{c}}}{c_r} q_{\text{ext,out},r}(k),$$

or,

$$e_r(k+1) = e_r(k) + \Delta e_r(k) + \frac{T_{\text{c}}}{c_r} \Delta q_{\text{in},r}(k - k_{\text{d},r}) - \frac{T_{\text{c}}}{c_r} \Delta q_{\text{out},r}(k) + \frac{T_{\text{c}}}{c_r} \Delta q_{\text{ext,in},r}(k) - \frac{T_{\text{c}}}{c_r} \Delta q_{\text{ext,out},r}(k), \quad (12)$$

where $\Delta e_r(k)$, $\Delta q_{\text{in},r}(k)$, $\Delta q_{\text{out},r}(k)$, $\Delta q_{\text{ext,in},r}(k)$, and $\Delta q_{\text{ext,out},r}(k)$ represent changes in the values of the respective variables from $k-1$ to k.

2) Undershot gates: By adjusting the gate position of undershot gates flows can be altered. Sometimes a local flow controller is present that accepts flow set-points and after that autonomously adjusts the gate position in order to meet the set-points. However, such a local flow controller is not always present and we therefore explicitly include the gate position of undershot gates in the model. In order to do this, the discharge formula of an undershot gate is linearized. Under free-flow conditions, the discharge for such a gate at the upstream end of reach r depends on the water level h_{r-1} at the downstream end of the upstream reach $r-1$ and on the opening of the gate $d_{\text{g},r}$ of reach r. The linearized discharge can be written down as [3]:

$$q_{\text{in},r}(k) = q_{\text{in},r}(k-1) + C_{e,r}(k)\Delta h_{r-1}(k) + C_{u,r}(k)\Delta d_{\text{g},r}(k),$$

with

$$C_{e,r}(k) = \frac{g c_{\text{w},r} W_{\text{s},r} \mu_r d_{\text{g},r}(k)}{\sqrt{2g(h_{r-1}(k) - (z_{\text{s},r} + \mu_r d_{\text{g},r}(k)))}}$$

$$C_{u,r}(k) = c_{\text{w},r} W_{\text{s},r} \mu_r \sqrt{2g(h_{r-1}(k) - (z_{\text{s},r} + \mu_r d_{\text{g},r}(k)))} - \frac{g c_{\text{w},r} W_{\text{s},r} \mu_r^2 d_{\text{g},r}}{\sqrt{2g(h_{r-1}(k) - (z_{\text{s},r} + \mu_r d_{\text{g},r}(k)))}},$$

where for reach r, $c_{\text{w},r}$ is a calibration coefficient, $W_{\text{s},r}$ is the width of the gate (m), μ_r is the contraction coefficient, $h_{r-1}(k)$ is the downstream level of the upstream canal reach $r-1$ (m), g the gravitational acceleration (m/s^2), $z_{\text{s},r}$ the crest level of the gate (m), and $d_{\text{g},r}(k)$ the gate opening (m). Hence, the following relation for the change in the discharge can be obtained:

$$\Delta q_{\text{in},r}(k) = C_{e,r}(k)\Delta e_{r-1}(k) + C_{u,r}(k)\Delta d_{\text{g},r}(k). \quad (13)$$

A similar relation is obtained for the downstream discharge as follows:

$$\Delta q_{\text{out},r}(k) = C_{e,r+1}(k)\Delta e_r(k) + C_{u,r+1}(k)\Delta d_{\text{g},r+1}(k). \quad (14)$$

3) Dependencies on neighboring reaches: The canal reaches controlled by a single controller are connected to one another. By (13) and (14) we observe that in order to evaluate the model of canal reach r, the values of the variables $\Delta e_{r-1}(k)$ and $h_{r-1}(k)$ of the upstream canal reach $r-1$ and of the variable $\Delta d_{\text{g},r+1}(k)$ of the downstream canal reach $r+1$ have to be known.

B. Operational constraints

Several operational constraints have to be satisfied with respect to the operation of canal reach r:

- There is a maximum value for the change in the gate position, both upwards and downwards, i.e.,

$$\Delta d_{\text{g},r}(k) \geq \Delta d_{\text{g},r,\min} \quad (15)$$

$$\Delta d_{\text{g},r}(k) \leq \Delta d_{\text{g},r,\max}, \quad (16)$$

where $\Delta d_{\text{g},r,\min} \leq 0$, $\Delta d_{\text{g},r,\max} \geq 0$.

- The gate position should always be positive and the gate should not be lifted out of the water. Therefore, a minimum and a maximum on the absolute gate position are present, i.e.,

$$d_{\mathrm{g},r}(k) \geq 0 \quad (17)$$

$$d_{\mathrm{g},r}(k) \leq \frac{2}{3}(h_{r-1}(k) - z_{\mathrm{s},r}), \quad (18)$$

where $\frac{2}{3}(h_{r-1}(k) - z_{\mathrm{s},r})$ is the maximum water level above the crest.

C. Control objectives

The changes in the gate position determined by controller i should be chosen in such a way that

1) the deviations of water levels from provided set-points e_r are minimized in all canal reaches;
2) the changes in the deviations of the water levels Δe_r from one control step to the next are minimized in all canal reaches to encourage smooth water level changes;
3) the changes in the gate positions $\Delta d_{\mathrm{g},r}$ are minimized in all canal reaches to reduce wear of equipment.

The objective function $J_{\mathrm{local},i}$ for controller i is therefore written as:

$$\begin{aligned} J_{\mathrm{local},i} = & \sum_{l=0}^{N-1} \sum_{r \in \mathcal{R}_i} q_e \left(e_r(k+1+l)\right)^2 \\ & + \sum_{l=0}^{N-1} \sum_{r \in \mathcal{R}_i} q_{\Delta e} \left(\Delta e_r(k+1+l)\right)^2 \\ & + \sum_{l=0}^{N-1} \sum_{r \in \mathcal{R}_i} q_{\Delta d_{\mathrm{g}}} \left(\Delta d_{\mathrm{g},r}(k+l)\right)^2, \end{aligned}$$

where q_e, $q_{\Delta e}$, and $q_{\Delta d_{\mathrm{g}}}$ are penalty coefficients. These penalty coefficients are chosen as follows:

$$q_e = \frac{1}{(e_{\mathrm{MAVE}})^2}, \; q_{\Delta e} = \frac{1}{(\Delta e_{\mathrm{MAVE}})^2}, \; q_{\Delta d_{\mathrm{g}}} = \frac{1}{(\Delta d_{\mathrm{g,MAVE}})^2},$$

where e_{MAVE}, Δe_{MAVE}, and $\Delta d_{\mathrm{g,MAVE}}$ are the maximum allowed value estimates (MAVE) of e, Δe, and Δd_{g}, respectively. These estimates indicate how much a variable is allowed to vary. By defining the objective function in this way the various objective terms in the objective function are normalized.

D. Summarizing

The equations representing the system are linear, and the objective functions are quadratic. It is now straightforward to cast the resulting prediction model, constraints, and objective function in the form suitable for application of the distributed MPC scheme of Section II. In the next section we employ this scheme based on linearized models to control a nonlinear representation of an irrigation canal.

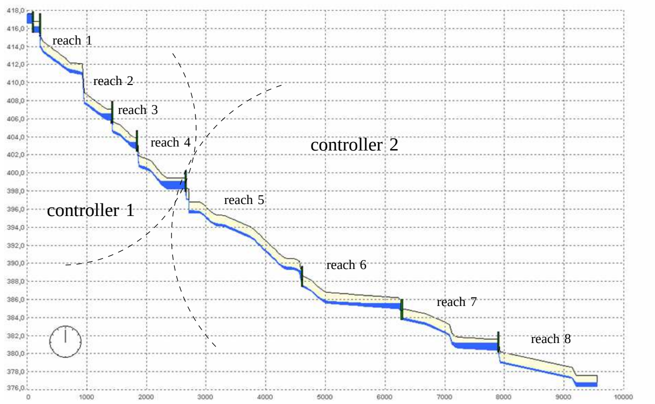

Fig. 3. Longitudinal view of the West-M irrigation canal and its division into two subnetworks.

IV. Case Study

In this section we describe a simulation result to illustrate the performance of the MPC scheme discussed in this paper. The irrigation canal that we consider is based on the West-M canal (as illustrated in Figure 3), which is an irrigation canal close to Phoenix, in the south of Arizona. This canal has been used by the ASCE Task Committee on Canal Automation Algorithms to define Test Canal 1 for testing automatic control schemes [8]. The canal is used to provide water to farmers. The length of the canal is almost 10 km and the maximum capacity of the head gate is $2.8\,\mathrm{m}^3/\mathrm{s}$ [3]. The canal consists of 8 canal reaches. At each of the reaches of the canal water can be taken out at offtakes for irrigation purposes. Between each of the reaches control structures are present in the form of undershot gates to change the water flow locally. Between canal reaches 5 and 6 a local PI controller is present, and therefore canal reaches 5 and 6 are considered as one reach. We refer to [8] for details on the dynamics of the canal.

For the benchmark system under study, MPC schemes have been proposed based on a single controller determining in a centralized way the set-points for the local flow controllers. MPC has been proposed for controlling the first 2 canal reaches of the benchmark system in [9], for controlling the first 3 canal reaches of the system in [10], and for controlling all canal reaches in [11], [3].

Here, we consider distributed control of the canal using two controllers that each control their own part of the network. For controller 1, the set of controlled reaches is $\mathcal{R}_1 = \{1,2,3,4\}$. For controller 2, the set of controlled reaches is $\mathcal{R}_2 = \{5,7,8\}$.

We consider a nonlinear simulation model of the canal, implemented in Sobek [12]. For solving the optimization problems at each control step we use the ILOG CPLEX v10.0 quadratic programming solver through the Tomlab v5.7 interface in Matlab v7.3.

A. Scenario

The time T_{c} between two consecutive control steps is 120 s. A prediction horizon length of $N = 30$ is chosen

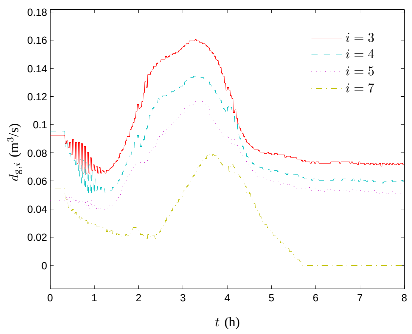

Fig. 4. Evolution of actions over the full simulation.

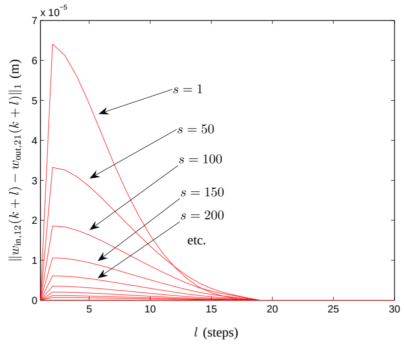

Fig. 5. Evolution at $t = 2.23$ of the absolute error (1-norm) between the interconnecting output of controller 1 and the interconnceting inputs of controller 2, i.e., the values that both controllers would like to give to the interconnecting variables $\Delta e_4(k+l)$, for $l = 1, \ldots, N$ over the iterations.

to take into account the total delay present in the irrigation canal. The controllers use as parameters $\gamma_c = 1000$, $\gamma_\epsilon = 1.10^{-3}$. As parameters for the objective functions the controllers use the following values: $e_{\text{MAVE}} = 0.15$, $\Delta e_{\text{MAVE}} = 0.005$, $\Delta u_{\text{MAVE}} = 0.0075$.

We perform a simulation of 240 time steps, corresponding to 8 hours. We consider an increase in the offtake of canal reach 3 at time step 60, corresponding to continuous $t = 2$ h, and a decrease in the offtake of the same canal reach at time step 120, corresponding to continuous time $t = 4$ h.

We show over a full simulation which actions the controllers choose, and illustrate for a particular time step how controllers obtain agreement on interconnecting variables.

B. Results

Figure 4 shows the gate settings that the two controllers determine to take. We can clearly observe how the controllers anticipate the additional offtake in reach 3 between $t = 2$ and $t = 4$ by already before $t = 2$ increasing the inflow in the reaches. Similarly, we observe that already before $t = 4$ the controllers again decrease their inflows, anticipating the offtake decrease in canal reach 3 at $t = 4$.

Figure 5 illustrates how at a particular time ($t = 2.23$) the controllers obtain agreement on the values of the interconnecting variables $\Delta e_4(k + l)$, for $l = 1, \ldots, N$. As the number of iterations increases (s becomes larger), the absolute error between the interconnecting inputs and interconnecting outputs with respect to $\Delta e_4(k+l)$ decreases, ultimately resulting in agreement.

In this experiment and in experiments with alternative scenarios (in each of which the gates where free flowing), we have observed that the performance of the distributed MPC approach over the full simulation is within 10% of the performance that obtainable by a centralized MPC approach.

V. Conclusions and Future Research

In this paper we have considered model predictive control (MPC) for distributed control of irrigation canals. We have discussed the use of an iteration-based, distributed MPC scheme for the control of irrigation canals. With this scheme performance comparable to the performance of a centralized MPC scheme can be achieved in a distributed way. On a benchmark irrigation canal we have illustrated the potential of the approach. In this case study, two controllers using linear prediction models have successfully determined which actions to take for controlling a nonlinear hydro-dynamic representation of the West-M irrigation canal in Arizona.

Future work consists of further assessing the performance of the proposed scheme, in particular when larger irrigation canals are controlled and the number of controllers increases. Moreover, when the gates become submerged, the dependencies between canal reaches will change. Future work will address this change.

Acknowledgments

Research supported by the BSIK project “Next Generation Infrastructures (NGI)”, the Delft Research Center Next Generation Infrastructures, the European STREP project “Hierarchical and distributed model predictive control (HD-MPC)”, and the project “Multi-Agent Control of Large-Scale Hybrid Systems” (DWV.6188) of the Dutch Technology Foundation STW.

References

[1] M. Cantoni, E. Weyer, Y. Li, S. K. Ooi, I. Mareels, and M. Ryan, “Control of large scale irrigation networks,” *Proceedings of the IEEE*, vol. 95, no. 1, pp. 75–91, Jan. 2007.

[2] D. D. Šiljak, *Decentralized Control of Complex Systems*. Boston, Massachusetts: Academic Press, 1991.

[3] P. J. van Overloop, “Model predictive control on open water systems,” Ph.D. dissertation, Delft University of Technology, Delft, The Netherlands, June 2006.

[4] R. R. Negenborn, P. J. van Overloop, T. Keviczky, and B. De Schutter, “Distributed model predictive control for irrigation canals,” *Networks and Heterogeneous Media*, vol. 4, no. 2, pp. 359–380, June 2009.

[5] R. R. Negenborn, B. De Schutter, and J. Hellendoorn, “Multi-agent model predictive control for transportation networks: Serial versus parallel schemes,” *Engineering Applications of Artificial Intelligence*, vol. 21, no. 3, pp. 353–366, Apr. 2008.

[6] V. T. Chow, *Open-Channel Hydraulics*. New York, New York: McGraw-Hill Book, 1959.

[7] J. Schuurmans and M. Ellerbeck, “Linear approximation model of the WM canal for controller design,” in *Proceedings of the First International Conference on Water Resources Engineering*, San Antonio, Texas, June 1995, pp. 353–357.

[8] A. J. Clemmens, T. F. Kacerek, B. Grawitz, and W. Schuurmans, “Test cases for canal control algorithms,” *Journal of Irrigation and Drainage Engineering*, vol. 124, no. 1, pp. 23–30, Jan. 1998.

[9] P. O. Malaterre and J. Rodellar, “Multivariable predictive control of irrigation canals. design and evaluation on a 2-pool model,” in *Proceedings of the International Workshop on Regulation of Irrigation Canals: State of the Art of Research and Applications*, Marrakech, Morocco, Apr. 1997, pp. 239–248.

[10] V. M. Ruiz and J. Ramirez, “Predictive control in irrigation canal operation,” in *Proceedings of the 1998 IEEE International Conference on Systems, Man, and Cybernetics*, San Diego, California, Oct. 1998, pp. 3897–3901.

[11] B. T. Wahlin, “Performance of model predictive control on ASCE test canal 1,” *Journal of Irrigation and Drainage Engineering*, vol. 130, no. 3, pp. 227–238, May 2004.

[12] Delft Hydraulics, “Sobek,” URL: http://delftsoftware.wldelft.nl/, 2008.