Flood Control with Model Predictive Control for River Systems with Water Reservoirs

Maarten Breckpot[1]; Oscar Mauricio Agudelo[2]; and Bart De Moor[3]

Abstract: Many control strategies can be found in the literature for controlling river systems. One of these methods is model predictive control (MPC), and it has already shown its efficiency for set-point control of reaches and irrigation channels. This paper shows that MPC can also be used for flood control of river systems. The proposed controllers use the buffer capacity of water reservoirs in an optimal way when there is a risk of flooding, and they recover the used buffer capacity as fast as possible. The performance of the controllers is tested on a river system consisting of multiple channels, gates, and a water reservoir. One controller is used in combination with a Kalman filter, which estimates all the states of the river system on the basis of a very limited number of measured water levels. It was observed that the influence of this estimator on the control performance was minimal. **DOI: 10.1061/(ASCE)IR.1943-4774.0000577.**

CE Database subject headings: Floods; Kalman filters; Open channel flow; Optimization; River systems; Reservoirs.

Author keywords: Control systems; Feedback control; Floods; Kalman filters; Open-channel flow; Optimization; River systems.

Introduction

Many different types of control strategies can be found in the literature for controlling channels, such as proportional-integral, heuristic, predictive, and optimal controllers (Malaterre et al. 1998; Burt et al. 1998; Litrico et al. 2006; van Overloop et al. 2005). The control strategy used in this work is model predictive control (MPC) (Rossiter 2003; Mayne et al. 2000). Model predictive control originates from the process industry and is used in various applications from chemicals and food processing to automotive and aerospace applications (Qin and Badgwell 2003). Because MPC formulates the control problem as an optimization problem, it can minimize the deviations of water levels from their targets, and it can be used for set-point control. It can also be used for flood control because flood limits can be incorporated inside the optimization problem. To achieve a high control performance, the controller requires an accurate model of the river system to be controlled.

The standard way to model the dynamics of a channel is done through the Saint-Venant equations. Based on these equations, together with the dynamics of hydraulic structures and junctions, a mathematical model can be derived for river systems. Because of the complexity of these models, controllers do not work directly with these equations but use approximate models. Models based on identification techniques or simple integrator-delay models form a good approximation and have been used in combination with MPC for set-point control (Schuurmans 1997; van Overloop 2006; Wahlin and Clemmens 2006; van Overloop et al. 2010; Puig et al. 2009; Negenborn et al. 2009; van Overloop et al. 2008). However, because these models describe the water levels of every channel at only one point, they are not the right choice for flood control. Therefore, a model approximating the Saint-Venant equations along the entire river is used in this study. A similar model has already been used in combination with MPC (Xu et al. 2011), but this work looked at only one channel and did not focus on flood control. Other works can be found on flood control, but they work with models describing the dynamics at a very limited number of points (Romera et al. 2011), do not take the nonlinearities of gates and hydraulic structures into account (Thai 2005), and none of them have water reservoirs, which should be used optimally.

Previous works have used a simplified conceptual model for the controller (Barjas Blanco et al. 2010; Breckpot et al. 2010). Recently, an approximate model of the Saint-Venant equations was used with a very fine spatial discretization for controlling a single channel (Breckpot et al. 2012a) and a two-channel system with one gate (Breckpot et al. 2012b). In this work, this method is extended to a larger river system where also junctions and a water reservoir are present. Also, a Kalman filter (Kalman 1960) is included in the control scheme because in practice only a limited number of water levels are measured, and MPC requires knowing the entire state of the system.

Contribution

This paper shows how MPC and a Kalman filter can be used for flood control of river systems containing water reservoirs. The authors are not aware of any other publication in which these techniques are applied to optimally use the buffer capacity of reservoirs for flood control on the basis of models approximating the river dynamics along the entire river. Because the proposed controllers work directly with approximations of the Saint-Venant equations describing the water-level dynamics along every channel, they can effectively be used for flood control.

[1]Ph.D. Fellow, Research Foundation—Flanders (FWO), Dept. of Electrical Engineering-ESAT (SCD-SISTA), KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium (corresponding author). E-mail: maarten.breckpot@esat.kuleuven.be

[2]Postdoctoral Researcher, Dept. of Electrical Engineering-ESAT (SCD-SISTA), KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium. E-mail: mauricio.agudelo@esat.kuleuven.be

[3]Full Professor, Dept. of Electrical Engineering-ESAT (SCD-SISTA)/iMinds Future Health Dept. KU Leuven, Kasteelpark Arenberg 10, B-3001 Leuven, Belgium. E-mail: bart.demoor@esat.kuleuven.be

Note. This manuscript was submitted on June 20, 2012; approved on January 14, 2013; published online on January 16, 2013. Discussion period open until December 1, 2013; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Irrigation and Drainage Engineering***, Vol. 139, No. 7, July 1, 2013. © ASCE, ISSN 0733-9437/2013/7-532-541/$25.00.

The main contribution of this paper is the formulation of the optimization problem. Inspired by a paper of Malaterre and Baume (1999), the authors did not work with the gate positions as control variables but with the gate discharges. This means that the gate equations are pulled out of the optimization problem. Because the gate dynamics are the main nonlinearities of river systems, the resulting optimization problem is less complex to solve. After finding the optimal gate discharges, a conversion is made based on the gate equations. The remaining model equations present in the optimization problem are based on the Saint-Venant equations. The paper shows that, for the purpose of controller design, these equations can be approximated accurately with one linear state-space model without losing control performance. The big advantage of this is that the total computation time needed by the controller is limited.

Another contribution lies in the formulation of the optimization problem such that the buffer capacity is used in an optimal way and the modification of its parameters to recover the used buffer capacity quickly.

Paper Outline

The paper starts by discussing the river system and the corresponding control problem, followed by showing how the dynamics of a river system can be modelled. Then it is explained how MPC can be used for flood control as well as the use of the Kalman filter for estimating the states. This is followed by a discussion of the simulation results when the proposed controllers are used for the test system. The paper ends with some concluding remarks.

Notation

Matrices are denoted with bold capital letters, e.g., $\mathbf{X}$, whereas bold lowercase letters are used for vectors, e.g., $\mathbf{x}$. Scalars or entries of vectors are not bold. The ith component of $\mathbf{x}$ is x_i. $\|\mathbf{x}\|_{\mathbf{W}}^2$ denotes $\mathbf{x}^{\mathbf{T}}\mathbf{W}\mathbf{x}$. Superscript (i) indicates the channel or the gate the variable or parameter belongs to, and 1_n represents a vector of ones of dimension n.

Problem Formulation

The goal of this study is to find a controller that can be used for set-point control of rivers while preventing them from flooding. For this the controller needs to use the available buffer capacity of the present water reservoirs in an optimal way during periods of heavy rainfall. The controller should also empty the reservoirs as fast as possible before focusing on set-point control again. Each channel has a safety limit and a flood limit. When the water levels risk violating only their safety limits, the controller is allowed to use the buffer capacity of the water reservoirs up to its own safety limit. If the water levels risk violating also the flood limits, then all the available buffer capacity of the water reservoirs can be used. The controller should also take into account that only a very limited number of water levels are measured in practice. The controller can use the future disturbances to find the best control actions.

Fig. 1 gives a schematic overview of the river system used to test the control performance. Channels 1–4 form the main part of the river, whereas Channels 5 and 7 connect the water reservoir (Channel 6) with the river. The squares represent hydraulic gates that can be used to control the discharges locally. $Q_{\text{in}}(t)$ is the discharge entering the river, whereas $Q_{\text{out}}(t)$ corresponds with the discharge leaving the river. When there is no flooding risk, the controller should keep the water levels of the first and the fourth channels as close as possible to their set points. The discharge $Q_{\text{in}}(t)$

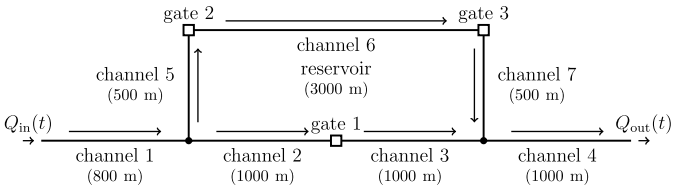

Fig. 1. River system to be prevented from flooding; arrows indicate general flow direction from upstream to downstream

entering the river system is a disturbance signal. The controller can control the water levels with $Q_{\text{out}}(t)$ and the three gates. Every control variable has an associated upper and lower limits, and the gates have a maximal rate of change constraint.

Modeling

Channel Modeling

The standard approach in the literature to model open channel flow is done by using the Saint-Venant equations (Chaudry 2008; Cunge et al. 1980; Sturm 2001; Litrico and Fromion 2009). The equations modeling the dynamics of a single channel without later inflow are the following two partial differential equations (PDEs):

$$\frac{\partial A(z,t)}{\partial h(z,t)}\frac{\partial h(z,t)}{\partial t}+\frac{\partial Q(z,t)}{\partial z}=0 \tag{1}$$

$$\frac{\partial Q(z,t)}{\partial t}+\frac{\partial}{\partial z}\frac{Q(z,t)^2}{A(z,t)}+gA(z,t)\left(\frac{\partial h(z,t)}{\partial z}+S_f(z,t)-S_0\right)=0 \tag{2}$$

where t = time variable; z = space variable; $Q(z,t)$ = water discharge (m^3/s); $h(z,t)$ = water depth (m); $A(z,t)$ = cross-sectional flow area (m^2); g = gravitational acceleration (m/s^2); S_0 = bed slope; and $S_f(z,t)$ = friction slope. Eq. (1) describes the conservation of mass, and Eq. (2) the conservation of momentum. The resistance law $S_f(z,t)$ is modeled in this study with the Manning relation (Chow 1959):

$$S_f(z,t)=\frac{n_{\text{mann}}^2 Q(z,t)|Q(z,t)|}{A(z,t)^2 R(z,t)^{1/3}} \tag{3}$$

where n_{mann} = Manning coefficient ($s/m^{1/3}$); $R(z,t)=A(z,t)/P(z,t)$ = hydraulic radius (m); and $P(z,t)$ = wetted perimeter of the cross section (m). The channels discussed in this study have a trapezoidal shape with side slope S. The parameters of a trapezoidal channel can be seen in Fig. 2, where L is the length of the channel (m), and B is the bottom width (m).

Together with these PDEs, upstream and downstream boundary conditions are needed for each channel. The most simple one is where the upstream or downstream discharge is given, e.g., it

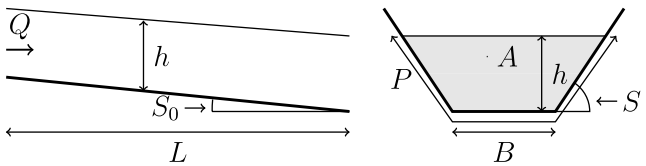

Fig. 2. Parameters of trapezoidal channel

J. Irrig. Drain Eng. 2013.139:532-541.

can be controlled directly with a pump or it might be a disturbance signal. Other possibilities are gates and junctions.

Gate Dynamics

The gates used in this study are underflow-vertical sluices. In general, the gate dynamics can be modeled as follows:

$$Q_{\text{gate}}(t) = \tilde{f}[c(t), h_{\text{up}}(t), h_{\text{down}}(t)] \tag{4}$$

where $Q_{\text{gate}}(t)$ = discharge controlled by the gate; $c(t)$ = gate opening (m); $h_{\text{up}}(t)$ = water level on the upstream part of the gate; $h_{\text{down}}(t)$ = water level on the downstream part; and $\tilde{f}:\mathbb{R}^3 \rightarrow \mathbb{R}$ is a nonlinear function. The function $\tilde{f}$ used in this study is based on Lin et al. (2002) and Sepúlveda et al. (2009) and has the following form:

$$Q_{\text{gate}}(t) = C_D(t)c(t)w\sqrt{2gh_{\text{up}}(t)} \tag{5}$$

where $C_D(t)$ = discharge coefficient; and w = width of the gate (m). The equation used for $C_D(t)$ depends on the flow condition:

$$\text{Free flow: } C_D(t) = \frac{C_C}{\sqrt{1+\alpha(t)}} \tag{6}$$

$$\text{Submerged flow: } C_D(t) = C_C \frac{\left[\beta(t) - \sqrt{\beta(t)^2 - \left(\frac{1}{\alpha(t)^2} - 1\right)^2\left(1 - \frac{1}{\gamma(t)^2}\right)}\right]^{1/2}}{\frac{1}{\alpha(t)} - \alpha(t)} \tag{7}$$

where C_C = contraction coefficient; $\beta(t) = [1/\alpha(t) - 1]^2 + 2[\gamma(t) - 1]$; $\alpha(t) = C_C c(t)/h_{\text{up}}(t)$; and $\gamma(t) = h_{\text{up}}(t)/h_{\text{down}}(t)$. C_C can vary from 0.598 to 0.74, but for most applications, 0.611 is usually assumed (Henderson 1966; Liggett and Cunge 1975). The gate is in free flow if $h_{\text{down}}(t)$ is below the limit $h_{\text{down,max}}(t)$:

$$h_{\text{down,max}}(t) = \frac{C_C c(t)}{2}\left(\sqrt{1 + \frac{16}{\alpha(t)[1+\alpha(t)]}}\right) \tag{8}$$

Otherwise, the gate is in a submerged flow condition. The roles of $h_{\text{up}}(t)$ and $h_{\text{down}}(t)$ interchange when $h_{\text{down}}(t)$ becomes larger than $h_{\text{up}}(t)$, and the discharge becomes negative.

Given these equations, the boundary conditions for two channels i and j connected to each other with a gate m can be formulated as follows:

$$Q^{(i)}(L^{(i)}, t) = Q^{(j)}(0, t) \tag{9}$$

$$\text{if } h^{(i)}(L^{(i)}, t) \geq h^{(j)}(0, t)$$

$$Q^{(i)}(L^{(i)}, t) = \tilde{f}[c^{(m)}(t), h^{(i)}(L^{(i)}, t), h^{(j)}(0, t)] \tag{10}$$

else

$$Q^{(i)}(L^{(i)}, t) = -\tilde{f}[c^{(m)}(t), h^{(j)}(0, t), h^{(i)}(L^{(i)}, t)] \tag{11}$$

This simply means that the gate discharge is equal to the discharge leaving the first channel and to that entering the second channel.

Junctions

Junctions are places along the river where multiple channels coincide. At these points all the water levels of the channels should be equal, and the sum of the discharges at the end of the upstream channels should be equal to the sum of the discharges at the beginning of the downstream channels. Applying this to the river system in Fig. 3 results in the following equations:

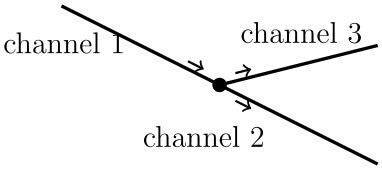

Fig. 3. Example of river system consisting of three channels coinciding in junction; arrows indicate general flow direction

$$h^{(1)}(L^{(1)}, t) = h^{(2)}(0, t) \tag{12}$$

$$h^{(1)}(L^{(1)}, t) = h^{(3)}(0, t) \tag{13}$$

$$Q^{(1)}(L^{(1)}, t) = Q^{(2)}(0, t) + Q^{(3)}(0, t) \tag{14}$$

Test Example

The river system to be controlled is shown in Fig. 1. Based on the equations and boundary conditions defined in the previous subsections, the mathematical model corresponding to this system consists of the following set of equations:

- The Saint-Venant equations [Eqs. (1) and (2)] to model the dynamics of every channel individually;
- The gate equations [Eqs. (9)–(11)] as boundary conditions between Channels 2 and 3, 5 and 6, and 6 and 7;
- Equations similar to Eqs. (12)–(14) for all channels ending in or starting from a junction: Channels 1, 2, and 5 and Channels 3, 4, and 7; and
- The following two last boundary conditions:

$$Q^{(1)}(0, t) = Q_{\text{in}}(t) \tag{15}$$

$$Q^{(4)}(L^{(4)}, t) = Q_{\text{out}}(t) \tag{16}$$

Table 1 contains the values of the parameters of the channels and the gates of the test example.

Discretization and Numerical Implementation

Because there is no analytical solution for the Saint-Venant equations, the infinite dimensional variables will be approximated on a finite grid (Strelkoff and Falvey 1993). The partial derivatives are approximated with finite differences, whereas the θ-method, i.e., $f(t_j + \theta\Delta t) = \theta f(t_j + \Delta t) + (1-\theta)f(t_j)$ with $\theta \in [0, 1]$ and Δt the integration step, is used for the time integration. The spatial

Table 1. Parameters of Different Channels and Gates of Test Example

	Channels				Gates
Parameter	1	2–4	5 and 7	6	1–3
L (m)	800	1,000	500	3,000	
S_0	0.0002	0.0002	0.0001	0.0001	
$n_{\text{mann}} \cdot (\text{s/m}^{1/3})$	0.02	0.02	0.02	0.02	
B (m)	6	6	6	6	
S	1	1	1	1	
w (m)					6

J. Irrig. Drain Eng. 2013.139:532-541.

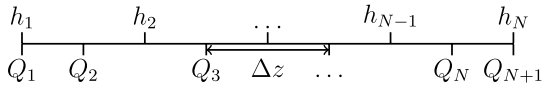

Fig. 4. Staggered grid structure used to discretize Saint-Venant equations

grid used in this paper is a staggered grid (see Fig. 4), with N water levels and $N+1$ discharges for a given channel. The derivatives in Eq. (1) are approximated by the following [$Q(z_i, t_j) = Q_i^j$; and $Q_i^{j+\theta} = \theta Q_i^{j+1} + (1-\theta)Q_i^j$]:

$$\frac{\partial h_i^j}{\partial t} \simeq \frac{h_i^{j+1} - h_i^j}{\Delta t} \tag{17}$$

$$\frac{\partial Q_i^j}{\partial z} \simeq \frac{Q_i^{j+\theta} - Q_{i-1}^{j+\theta}}{\Delta z} \tag{18}$$

$$\frac{\partial A_i^j}{\partial h} \simeq \left(\frac{\partial A}{\partial h}\right)_i^{j+\theta} \tag{19}$$

A similar approach is used for the terms $\partial Q(z,t)/\partial t$, $A(z,t)$, $\partial h(z,t)/\partial z$, and $S_f(z,t)$ in Eq. (2). The advection term $\partial[Q^2(z,t)/A(z,t)]/\partial z$ is approximated with an upwinding approach:

$$\frac{\partial}{\partial z}\left(\frac{Q^2}{A}\right)_i^j \simeq \begin{cases} \frac{1}{\Delta z}\left[\left(\frac{Q^2}{A}\right)_{i+1}^{j+\theta} - \left(\frac{Q^2}{A}\right)_i^{j+\theta}\right] & Q_i^j < 0 \\ \frac{1}{\Delta z}\left[\left(\frac{Q^2}{A}\right)_i^{j+\theta} - \left(\frac{Q^2}{A}\right)_{i-1}^{j+\theta}\right] & Q_i^j \geq 0 \end{cases} \tag{20}$$

In this way, the two PDEs describing the dynamics of a single channel are transformed into a system of nonlinear equations:

$$\mathbf{f}[\mathbf{h}(t_{j+1}), \mathbf{h}(t_j), \mathbf{q}(t_{j+1}), \mathbf{q}(t_j)] = \mathbf{0} \tag{21}$$

where $\mathbf{h}(t_j) = [h_1(t_j), \ldots, h_N(t_j)]^T$; $\mathbf{q}(t_j) = [Q_1(t_j), \ldots, Q_{N+1}(t_j)]^T$; and $\mathbf{f}$:$\mathbb{R}^{4N+2} \rightarrow \mathbb{R}^{2N+1}$. The system of nonlinear equations [Eq. (21)] for each channel, together with all the boundary conditions, needs to be solved for $\mathbf{h}(t_{j+1})$ and $\mathbf{q}(t_{j+1})$ of every channel given the values of these variables at time step t_j. This can be done with Newton method. A discussion about the choice of Δt and θ can be found in (Clemmens et al. 2005). In this paper, θ is set to 0.6.

Model Predictive Control

To find a controller that is able to prevent rivers from flooding, it is important that the controller considers the flood levels when looking for the best control actions. The controller should also take possible actuator limits and rainfall predictions into account. Model predictive control is an optimization-based control strategy that can perfectly deal with these requirements. It uses a process model to predict the future process outputs within a specified prediction horizon. It determines the next inputs for the process by solving an optimization problem over this horizon, taking input and output constraints (e.g., flood limits), future disturbances (rainfall predictions), and the process model into account. Only the first element of the complete optimal sequence is applied to the process, the new current state of the system is measured or estimated, and the entire procedure is repeated.

Nonlinear MPC

Approximate Model

Model predictive control solves at every time step an optimization problem in which the model equations are considered as equality constraints. However, the model equations discussed in the previous section are too complex to be used directly inside the optimization problem. The main complexity comes from the nonlinear gate equations. Inspired by Malaterre and Baume (1999), these gate equations can be excluded from the optimization problem when we work with the discharge at the gates as control variables instead of the gate positions. Once the optimal gate discharges are found, a conversion step is needed to find the corresponding gate positions. Under this control scheme, the boundary conditions for the river system are given by the following instead of Eqs. (9)–(11):

$$Q^{(i)}(L^{(i)}, t) = Q_{\text{gate}}^{(m)}(t) \tag{22}$$

$$Q^{(j)}(0, t) = Q_{\text{gate}}^{(m)}(t) \tag{23}$$

The resulting model equations will be approximated with linear time-varying state-space models by linearizing the discretized Saint-Venant equations [Eq. (21)] around the linearization point $\mathbf{x}_{\text{lin}}(k)$, $\mathbf{u}_{\text{lin}}(k)$, and $\mathbf{d}_{\text{lin}}(k)$:

$$\begin{aligned} \mathbf{x}(k+1) - \mathbf{x}_{\text{lin}}(k+1) = {} & \mathbf{A}(k)[\mathbf{x}(k) - \boldsymbol{x}_{\text{lin}}(k)] \\ & + \mathbf{B}(k)[\mathbf{u}(k) - \mathbf{u}_{\text{lin}}(k)] \\ & + \mathbf{D}(k)[\mathbf{d}(k) - \mathbf{d}_{\text{lin}}(k)] \end{aligned} \tag{24}$$

where $\mathbf{x}(k) = [\mathbf{h}^{(1)T}(t_k), \ldots, \mathbf{h}^{(n_c)T}(t_k), \mathbf{q}^{(1)T}(t_k), \ldots, \boldsymbol{q}^{(n_c)T}(t_k)]^T$; n_c = number of channels; $\boldsymbol{u}(k) = [Q_{\text{gate}}^{(1)}(t_k), Q_{\text{gate}}^{(2)}(t_k), Q_{\text{gate}}^{(3)}(t_k), Q^{(4)}(L^{(4)}, t_k)]^T$; $\boldsymbol{d}(k) = Q^{(1)}(0, t_k)$; and $\mathbf{A}(k)$, $\mathbf{B}(k)$, and $\mathbf{D}(k)$ = state-space matrices at time t_k. Eq. (24) can be rewritten as

$$\mathbf{x}(k+1) = \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{D}(k)\mathbf{d}(k) + \boldsymbol{\beta}(k) \tag{25}$$

where

$$\boldsymbol{\beta}(k) = \mathbf{x}_{\text{lin}}(k+1) - \mathbf{A}(k)\mathbf{x}_{\text{lin}}(k) - \mathbf{B}(k)\mathbf{u}_{\text{lin}}(k) - \mathbf{D}(k)\mathbf{d}_{\text{lin}}(k) \tag{26}$$

The linearization points $\mathbf{x}_{\text{lin}}(k)$, $\mathbf{u}_{\text{lin}}(k)$, and $\mathbf{d}_{\text{lin}}(k)$ should be as close as possible to the values found by the optimizer based on these linear models. Therefore, $\mathbf{d}_{\text{lin}}(k)$ is set equal to the predicted disturbance signal, and $\mathbf{u}_{\text{lin}}(k)$ will be set equal to the optimal control actions found by the optimizer in the previous time step. $\mathbf{x}_{\text{lin}}(k)$ is found by using the nonlinear model equations to predict what the states are corresponding to $\mathbf{u}_{\text{lin}}(k)$ and $\mathbf{d}_{\text{lin}}(k)$. After this prediction step, the linear models are derived and can be used in the optimization problem.

Optimization Problem

The optimization problem that needs to be solved at every time step is a quadratic program (QP) of the following form:

$$\begin{aligned} \min_{u,x,\boldsymbol{\xi},\boldsymbol{\zeta}} & \sum_{k=1}^{N_P} \|\mathbf{x}(k) - \mathbf{r}\|_W^2 + \sum_{k=0}^{N_P-1} \|\mathbf{u}(k) - \mathbf{u}(k-1)\|_{\boldsymbol{R}}^2 \\ & + \|\boldsymbol{\xi}\|_S^2 + \mathbf{s}^T\boldsymbol{\xi} + \|\boldsymbol{\zeta}\|_V^2 + \mathbf{v}^T\boldsymbol{\zeta} \end{aligned} \tag{27}$$

$$\text{s.t. } \mathbf{x}(0) = \hat{\mathbf{x}}_0 \tag{28}$$

$$\mathbf{x}(k+1) = \mathbf{A}(k)\mathbf{x}(k) + \mathbf{B}(k)\mathbf{u}(k) + \mathbf{D}(k)\mathbf{d}(k) + \boldsymbol{\beta}(k) \tag{29}$$

J. Irrig. Drain Eng. 2013.139:532-541.

$$\underline{\mathbf{u}}(k) \le \mathbf{u}(k) \le \bar{\mathbf{u}}(k), \qquad \forall \ \ i = 1, \ldots, n_c: \tag{30}$$

$$\mathbf{h}^{(i)}(k) \le \mathbf{h}^{(i)}_{\max,1} + \eta(k)\xi_i \tag{31}$$

$$\mathbf{h}^{(i)}(k) \le \mathbf{h}^{(i)}_{\max,2} + \eta(k)\zeta_i \tag{32}$$

$$\boldsymbol{\xi} \ge 0 \tag{33}$$

$$\boldsymbol{\zeta} \ge 0 \tag{34}$$

where N_P = prediction horizon; $\mathbf{W} \in \mathbb{R}^{n_x \times n_x}$, $\mathbf{R} \in \mathbb{R}^{n_u \times n_u}$, $\mathbf{S} \in \mathbb{R}^{n_c \times n_c}$, and $\mathbf{V} \in \mathbb{R}^{n_c \times n_c}$ = four positive semidefinite diagonal weighting matrices; n_x = total number of water levels and discharges; n_u = number of inputs; $\mathbf{s} \in \mathbb{R}^{n_c}$ and $\mathbf{v} \in \mathbb{R}^{n_c}$ = two weighting vectors; $\mathbf{r}$ = set points for the states; $\hat{\mathbf{x}}_0$ = current state estimate of the process; $\mathbf{h}_{\max,1}$ = safety levels; $\mathbf{h}_{\max,2}$ = flood levels; $\underline{\mathbf{u}}$ and $\bar{\mathbf{u}}$ = operational limits on the inputs; $\boldsymbol{\xi}, \boldsymbol{\zeta} \in \mathbb{R}^{n_c}$ = two vectors of slack variables (one slack variable for each channel); and $\eta(k) = 1/r_c^{k-1}$ = time-dependent weight ($r_c > 1$). The weighting matrices $\mathbf{W}$, $\mathbf{R}$, $\mathbf{S}$, and $\mathbf{V}$ define the relative importance of the difference between the states and their set points, the changes of the control actions, and the two vectors of slack variables $\boldsymbol{\xi}$ and $\boldsymbol{\zeta}$. It can be shown that this QP has only one (global) solution (Nocedal and Wright 2000). In this study, quadprog of *MATLAB* (2011a) is used to solve the QPs. This controller will be called NMPC (nonlinear MPC).

The meaning of the safety limits and the flood limits was explained in the "Problem Formulation" section. The flood limits and safety limits for the water levels are implemented as soft constraints [Eqs. (31) and (32)]. The maximal violation of these limits for each channel along the prediction horizon are penalized in the objective function by the terms $\|\boldsymbol{\xi}\|_S^2 + \mathbf{s}^T\boldsymbol{\xi}$ and $\|\boldsymbol{\zeta}\|_V^2 + \mathbf{v}^T\boldsymbol{\zeta}$. A sufficiently large $\mathbf{s}$ and $\mathbf{v}$ will ensure that the constraints will only be violated when no feasible solution exists for the hard constrained optimization problem (Hovd and Braatz 2001). This means that the constraints are enforced as exact soft constraints. If the constraints cannot be prevented from being violated, the controller will minimize these violations and hence reduce the flood risk. The quadratic terms $\|\boldsymbol{\xi}\|_S^2$ and $\|\boldsymbol{\zeta}\|_V^2$ are imposed to have a well-posed QP and are extra tuning parameters (Scokaert and Rawlings 1999). A time-dependent weight $\eta(k)$ is used to penalize future constraint violations in the prediction horizon increasingly to avoid long-lasting constraint violations (Hovd and Braatz 2001).

Because the buffer capacity of the reservoir between the safety limit and the flood limit may not be used to keep the water levels of all the other channels below their safety limits, the diagonal element of $\mathbf{S}$ and the element of $\mathbf{s}$ corresponding to the reservoir are set higher than the elements corresponding with the other channels. The diagonal elements of $\mathbf{V}$ and the elements of $\mathbf{v}$ are set higher than all the elements of $\mathbf{S}$ such that the controller will use the remaining buffer capacity for flood prevention of all channels. If there is no flood risk, the controller needs to focus on setpoint control of the water levels of Channels 1 and 4. This is achieved by choosing the elements in the matrix $\mathbf{W}$ corresponding to these water levels larger (but sufficiently smaller than the weights of the slack variables) than the entries associated to the water levels of the other channels. $\mathbf{R}$ influences the control effort of the different input variables.

$\underline{\mathbf{u}}(k)$ and $\bar{\mathbf{u}}(k)$ are the operational limits on the inputs at time step k. These limits are time variable for the control actions corresponding to the three gate discharges. For the controllable discharge at the end of the river, they are constant and assumed to be given. These limits are time variable because they depend on the rate of change constraint $\boldsymbol{\Delta}_{\max}$ of the gates and on the upstream and downstream water levels at each gate at every time step. These limits can be calculated with an Algorithm 1. In this algorithm n_g is equal to the total number of gates, $\mathbf{u}_{\text{pred}}$ is the sequence of control actions found by the optimizer in the previous time step, $c_{\min,m}$ is the lower limit on gate m, and $c_{\max,m}$ is the upper limit on gate m. The algorithm uses a function l to calculate the gate position corresponding to a desired gate discharge and upstream and downstream water levels. There is one important aspect of this function. There can be situations in which the position of a gate can remain unchanged from iteration j to $j+1$ to achieve a given gate discharge. However, for decreasing water levels, the gate opening can become larger than these water levels. The calculated $\underline{u}_m(j+2)$ and $\bar{u}_m(j+2)$ will have the same value if the $\Delta_{\max,m}$ is not big enough to lower the gate between the maximal and minimal water levels. This can happen for all future values of j and at the end the lower limit is equal to the upper limit for that control variable at every time step within the prediction window. This indicates that the value for the gate discharge is fixed and the controller will stop using this gate, losing in this way one degree of freedom. This uncontrollability can be avoided by limiting the maximal value returned by this function l with the maximal water level at the gate. This will make sure that $\underline{u}_m$ will be different from $\bar{u}_m$ at every time step, which will ensure the controllability of the gate. Internally, the function l solves Eqs. (5)–(7) iteratively with the bisection method.

Because the linear model used in the optimization problem is only an approximation of the nonlinear model and because of the dependence of the limits $\underline{\mathbf{u}}$ and $\bar{\mathbf{u}}$ on the states of the system, the solution of the QP is not guaranteed to be the solution of the nonlinear optimization problem. Therefore, the prediction step, the linearization step, the algorithm, and the optimization step will be iterated over. First, the optimal control actions found in the previous time step are used with the nonlinear model to predict what the future states of the model will be within the prediction window. On the basis of this sequence of future states, a sequence of linear state space models is derived, which is used to find $\underline{\mathbf{u}}$ and $\bar{\mathbf{u}}$ and in the optimization problem. This procedure is repeated with the new control actions until convergence or a maximum number of iterations is reached. Then, the controller looks for the gate positions corresponding to the desired gate discharges and applies them to the river system. This conversion is done with the function l discussed previously.

Algorithm 1. Calculation of the time-varying limits

Algorithm to find the time-varying limits $\underline{\mathbf{u}}(k) = [\underline{u}_1(k), \ldots, \underline{u}_{n_g}(k)]^T$ and $\bar{\mathbf{u}}(k) = [\bar{u}_1(k), \ldots, \bar{u}_{n_g}(k)]^T$. The function s selects the upstream and downstream water levels corresponding to gate m, whereas the function l calculates the gate position corresponding to the desired discharge for gate m.

% at time step t_k
$\tilde{\mathbf{x}} = \mathbf{x}(t_k)$
$c_{\text{prev}} = [c^{(1)}(t_{k-1}), \ldots, c^{(n_g)}(t_{k-1})]^T$
for $j = 1, \ldots, N_P$ do
for $m = 1, \ldots, n_g$ do
$[h_{\text{up}}, h_{\text{down}}] = s(\hat{\mathbf{x}}, m)$
if $h_{\text{up}} \ge h_{\text{down}}$ then
$\underline{u}_m(j) = f[\max(c_{\text{prev},m} - \Delta_{\max,m}, c_{\min,m}), h_{\text{up}}, h_{\text{down}}]$
$\bar{u}_m(j) = f[\min(c_{\text{prev},m} + \Delta_{\max,m}, c_{\max,m}), h_{\text{up}}, h_{\text{down}}]$
else

J. Irrig. Drain Eng. 2013.139:532-541.

$\bar{u}_m(j) = -\tilde{f}[\max(c_{\text{prev},m} - \Delta_{\max,m}, c_{\min,m}), h_{\text{down}}, h_{\text{up}}]$
$\underline{u}_m(j) = -\tilde{f}[\min(c_{\text{prev},m} + \Delta_{\max,m}, c_{\max,m}), h_{\text{down}}, h_{\text{up}}]$
end if
$c_{\text{prev},m} = l[u_{\text{pred},m}(j), h_{\text{up}}, h_{\text{down}}]$
end for
$\boldsymbol{d} = Q_{\text{in}}(t_{k+j-1})$ $\tilde{\mathbf{x}} = \mathbf{A}(j)\tilde{\mathbf{x}} + \mathbf{B}(j)\mathbf{u}_{\text{pred}}(k) + \mathbf{D}(j)\mathbf{d} + \boldsymbol{\beta}(j)$
end for

Buffer Capacity Recovery

After a period of heavy rainfall, the controller should recover the buffer capacity to handle future rainfall without flooding. This can be achieved by working with two different weighting matrices $\mathbf{W}$ and reference signals $\mathbf{r}$. During normal operation (set-point control), the diagonal elements of the matrix $\mathbf{W}$ corresponding to the most important water levels will have the highest values compared with those corresponding to the other water levels, and r is set equal to the desired set points. However, when some of the buffer capacity is used for flood prevention, the set points of the water levels of the channels downstream of the reservoir (Channels 3, 4, and 7) are set lower than the set point of the reservoir, and the diagonal elements of $\mathbf{W}$ corresponding to these water levels are increased. By decreasing these water levels, Gate 3 can be used to remove the excess water in the reservoir. This change in $\mathbf{W}$ and $\mathbf{r}$ is performed when one of the water levels in the reservoir is 20 cm above its set point. This is checked every time a new state estimate of the process is computed. When the buffer capacity is recovered, $\mathbf{W}$ and $\mathbf{r}$ are set equal to their initial values (set-point control). This is done when all the water levels of the reservoir are at the most 10 cm above their set point for at least 2.5 h.

Linear MPC

Because of the prediction step, the NMPC requires high computation times to find optimal control actions. This step can be avoided by working with the same linear model at every time step. To this end, the model constraint Eq. (29) in the NMPC formulation is replaced by

$$\mathbf{x}(k+1) = \bar{\mathbf{A}}\mathbf{x}(k) + \bar{\mathbf{B}}\mathbf{u}(k) + \bar{\mathbf{D}}\mathbf{d}(k) + \bar{\boldsymbol{\beta}} \tag{35}$$

where $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}$, $\bar{\mathbf{D}}$, and $\bar{\boldsymbol{\beta}}$ are found by linearizing the discretized model equations around the nominal steady-state operating point of the river. The algorithm can still be used to find $\underline{\mathbf{u}}(k)$ and $\bar{\mathbf{u}}(k)$ if the time-varying state-space matrices $\mathbf{A}(k)$, $\mathbf{B}(k)$, $\mathbf{D}(k)$, and the vector $\boldsymbol{\beta}(k)$ are replaced with $\bar{\mathbf{A}}$, $\bar{\mathbf{B}}$, $\bar{\mathbf{D}}$, and $\bar{\boldsymbol{\beta}}$, respectively. This controller will be called LMPC (linear MPC).

Because there is still a dependence between $\underline{\mathbf{u}}(k)$ and $\bar{\mathbf{u}}(k)$, the chosen control actions, and the water levels surrounding every gate, in theory several iterations involving the algorithm and solving the QP are needed. However, the simulation results show that good results can already be achieved by performing these operations only once.

State Estimator

The controllers defined in the previous section require the knowledge of the current state of the process x_0. However, only a very limited number of water levels are measured and none of the discharges. Therefore, all the other states need to be estimated. The state estimator used in this paper is the Kalman filter (Kalman 1960). This is explained here for the LMPC. The estimation of the state vector $\mathbf{x}(k)$ will be written as $\hat{\mathbf{x}}(k)$. The Kalman filter uses the linear state-space model in combination with the error between the predicted and the measured water levels to correct the state estimation through a feedback gain matrix $\mathbf{L}$:

$$\Delta\hat{\mathbf{x}}(k+1) = \mathbf{L}[\Delta\mathbf{y}(k) - \Delta\hat{\mathbf{y}}(k)] + \bar{\mathbf{A}}\Delta\hat{\mathbf{x}}(k) + \bar{\mathbf{B}}\Delta\mathbf{u}(k) + \bar{\mathbf{D}}\Delta\mathbf{d}(k) \tag{36}$$

$$\Delta\hat{\mathbf{y}}(k) = C\Delta\hat{\mathbf{x}}(k) \tag{37}$$

where $\Delta\mathbf{x}(k) = \mathbf{x}(k) - \mathbf{x}_{\text{lin}}$; $\Delta\hat{\mathbf{x}}(k) = \hat{\mathbf{x}}(k) - \mathbf{x}_{\text{lin}}$; $\Delta\mathbf{u}(k) = \mathbf{u}(k) - \mathbf{u}_{\text{lin}}$; $\Delta\mathbf{y}(k) = \mathbf{y}(k) - \mathbf{y}_{\text{lin}}$; $\Delta\hat{\mathbf{y}}(k) = \hat{\mathbf{y}}(k) - \mathbf{y}_{\text{lin}}$; $\mathbf{y}(k)$ = measured water levels; $\hat{\mathbf{y}}(k)$ = estimated water levels; and $\mathbf{C}$ = matrix that selects the measured water levels from the states. The Kalman gain $\mathbf{L}$ is found by minimizing the covariance of the estimation error $\mathbf{x}(k) - \hat{\mathbf{x}}(k)$, taking process and measurement noise into account. Both noises are assumed to be independent white noise with a normal distribution. The exact derivation for finding $\mathbf{L}$ can be found in Franklin et al. (1997) and Kwakernaak and Sivan (1972).

Simulation Results

This section describes the simulations results when NMPC, LMPC, and LMPC with Kalman filter are applied to the river system in Fig. 1. For the first two controllers, it is assumed that all the states are measured at every time step. This allows the comparison of the control performance of these two controllers separately. Only the effect of the Kalman filter on LMPC is shown, as this controller has the lowest computation time and a similar performance as NMPC.

The sampling time of every controller is 15 min, whereas the full nonlinear model of the river system generates data every minute. A spatial discretization Δz of 50 m is used. The maximum number of iterations for the NMPC is limited to five or when the Euclidean norm of the difference of the control actions found between two consecutive iterations is smaller than 10^{-3}. Only one iteration is used for LMPC and LMPC with the Kalman filter.

Initially, the discharges in the main river part are all 3 m^3/s, whereas there is no flow in the reservoir and the two side channels. This indicates that the second and the third gates are closed. The most downstream water level of the second and the fourth channels is 5 m, whereas the most downstream water level of the reservoir is 4 m. The set points for all the water levels are equal to the initial water levels. The discharge at the end of the fourth channel should remain between 0 and 5 m^3/s. All the gate openings should be between 0 and 5 m, and they can only be changed over a maximum distance of 20 cm every time step. The differences between the initial water levels and the safety levels are 0.6, 0.6, and 1.2 m for Channels 1, 2, and 5, Channels 3, 4, and 7, and the reservoir, and the differences with respect to the flood levels are 1, 0.8, and 2 m, respectively. The disturbance signal is depicted in Fig. 5. Initially, the discharge entering the river system is 3 m^3/s, which is

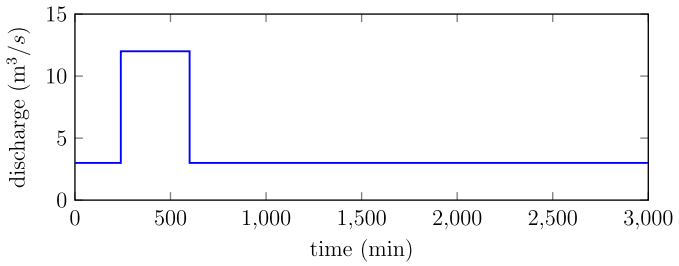

Fig. 5. Disturbance signal $Q_{\text{out}}(t)$

J. Irrig. Drain Eng. 2013.139:532-541.

Table 2. Diagonal Elements for Weight Matrices **W**, **S**, **V**, and **R** and Elements for Weight Vectors **s** and **v**

Elements	Channel 1	2	3	4	5	6	7
$\mathbf{W} \in \mathbb{R}^{333\times333}$							
Water levels	$10 \cdot \mathbf{1}_{17}$	$0.001 \cdot \mathbf{1}_{21}$	$0.001 \cdot \mathbf{1}_{21}$	$1 \cdot \mathbf{1}_{21}$ $1{,}000 \cdot \mathbf{1}_{21}$[a]	$0.001 \cdot \mathbf{1}_{11}$	$1{,}000 \cdot \mathbf{1}_{61}$	$0.001 \cdot \mathbf{1}_{11}$
Discharges	$0.001 \cdot \mathbf{1}_{18}$	$0.001 \cdot \mathbf{1}_{22}$	$0.001 \cdot \mathbf{1}_{22}$	$0.001 \cdot \mathbf{1}_{22}$	$0.001 \cdot \mathbf{1}_{12}$	$0.001 \cdot \mathbf{1}_{62}$	$0.001 \cdot \mathbf{1}_{12}$
$\mathbf{S} \in \mathbb{R}^{7\times7}$							
Safety levels	10^5	10^5	10^5	10^5	10^5	10^6	10^5
$\mathbf{s} \in \mathbb{R}^{7\times1}$							
Safety levels	10^5	10^5	10^5	10^5	10^5	10^6	10^5
$\mathbf{V} \in \mathbb{R}^{7\times7}$							
Flood levels	10^8	10^8	10^8	10^8	10^8	10^8	10^8
$\mathbf{v} \in \mathbb{R}^{7\times1}$							
Flood levels	10^8	10^8	10^8	10^8	10^8	10^8	10^8
$\mathbf{R} \in \mathbb{R}^{4\times4}$	Q_{gate1}	Q_{gate2}	Q_{gate3}	Q_{out}			
Control actions	50	50	50	50			

[a]Value used to recover the buffer capacity of the reservoir. The weighting factor for each of the 21 water levels changes from 1 to 1,000.

increased after 4 h to 12 m^3/s and is decreased back to 3 m^3/s 6 h later.

Table 2 contains the diagonal elements for the weight matrices **W**, **S**, **V**, and **R** and the elements for the weight vectors **s** and **v**. As explained in the "Problem Formulation" section, during normal operation the water levels of the Channels 1 and 4 and of the reservoir are the most important ones. Therefore, the weights of **W** corresponding to these water levels are set to 10, 1, and 1,000, respectively. All the other weights for the other water levels are fixed to 0.001. The weights for the discharges for all the channels are set to 0.001. If the buffer capacity is used to prevent the water levels from violating their safety limits, the set points for the most downstream water levels (Channel 4) are set below the set point of the reservoir, and the corresponding weights in **W** are increased to 1,000 such that the buffer capacity can be recovered. All the diagonal elements of **R** are set to 50, the weights in **S** and **s** for the slack variable $\boldsymbol{\xi}$ associated to the reservoir are set to 10^6, the weights associated to the other channels are set to 10^5, the weights in **V** and **v** for the slack variable $\boldsymbol{\zeta}$ are assumed equal to 10^8, r_c is set to 1.2, and the size of the prediction window N_P is 15.

Fig. 6 shows the evolution of the maximal difference between the water levels and their set points for the three controllers. To keep the number of plots as low as possible, the results for the upstream part of the river (Channels 1, 2 and 5) are plotted together (the top plot) as are the results for the downstream part of the river (Channels 3, 4, and 7, the middle plot). The bottom plot shows the results for the reservoir (Channel 6). Fig. 7 shows the applied control actions for the three gates and $Q_{\text{out}}(t)$ by each of the controllers. The first conclusion that can be made from Figs. 6 and 7 is that the differences in the results between the NMPC and the LMPC are small. The control actions and the water levels have a similar general trend for both controllers. Also, the addition of the estimator in the control loop has only a limited effect on the control performance. There is only a noticeable difference in the control actions for the three controllers during the period of heavy rainfall and right after this period. However, because the trend of the control actions for all the three controllers is similar, the water levels show the same behavior. Initially, all the three controllers try to avoid using the water reservoir and react to the future rainfall by decreasing the upstream water levels. This is done by increasing the gate opening of the first gate and increasing the discharge at the end of the river to its maximal value. Once the water levels risk violating the safety levels, the controllers start using the water reservoir by opening the second and the third gates. Because of the magnitude of the upstream disturbance, the controllers cannot avoid the violation of the safety levels; however, they can prevent the violation of the flood levels or at worst minimize their violation. Table 3 shows the maximal difference between the water levels of each channel and their flood levels for the three controllers. A positive value corresponds to the maximal violation of the flood levels, whereas a negative value corresponds to the minimal margin before a flood level

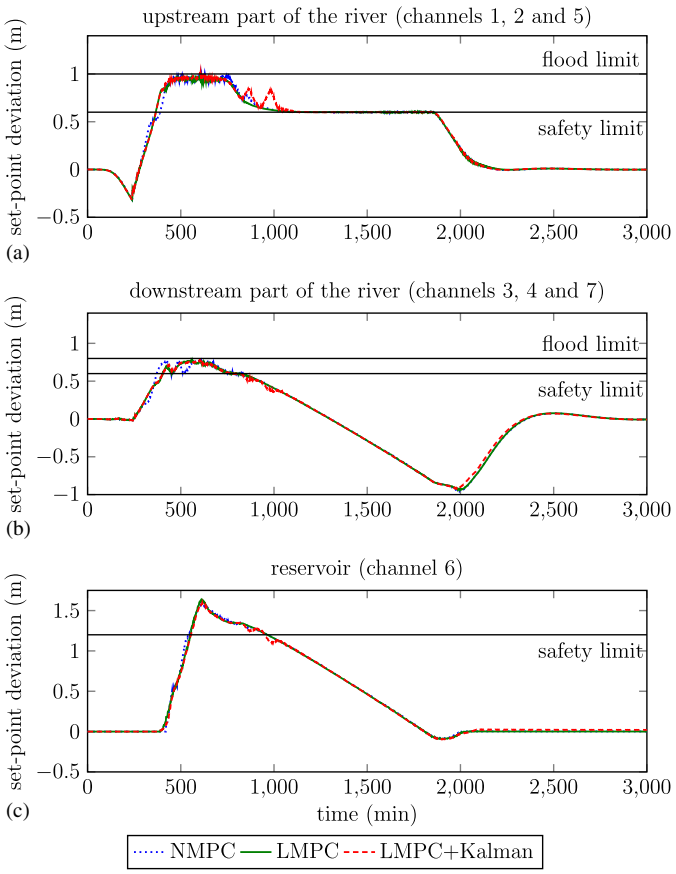

Fig. 6. Evolution of maximal difference between water levels and their set points for three controllers; plots show maximal deviation of water levels at every time step of Channels (a) 1, 2, and 5; (b) 3, 4, and 7; (c) 6; flood and safety levels correspond to horizontal lines

J. Irrig. Drain Eng. 2013.139:532-541.

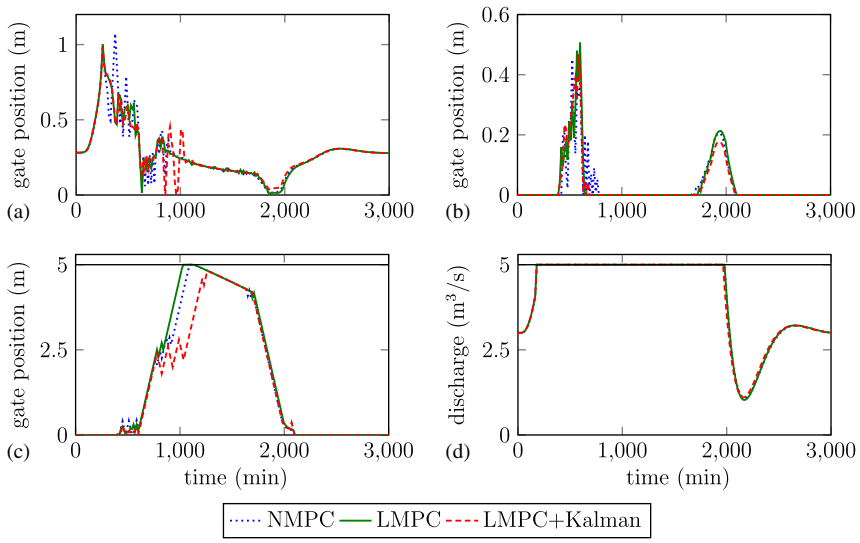

Fig. 7. Control actions of three controllers; horizontal lines are upper limits on control variables: (a) Gate 1; (b) Gate 2; (c) Gate 3; (d) discharge $Q_{\text{out}}(t)$ at end of river system

Table 3. Maximal Flooding of Each Channel for Three Controllers

Channel	NMPC (m)	LMPC (m)	LMPC + Kalman (m)
1	0.0029	−0.0263	0.0160
2	0.0023	−0.0258	0.0090
3	−0.0283	−0.0360	−0.0292
4	−0.0265	−0.0277	−0.0288
5	0.0024	−0.0284	0.0051
6 (reservoir)	−0.4054	−0.3670	−0.4037
7	−0.0290	−0.0293	−0.0249

Note: Negative value indicates that there is no flooding and some buffer capacity is left.

Table 4. Average Deviation in Meters of Each Channel for Three Controllers from Their Set Points at End of Simulation ($t = 3{,}000$ min)

Channel	NMPC (m)	LMPC (m)	LMPC + Kalman (m)
1	−0.0012	−0.0012	−0.0012
2	−0.0012	−0.0012	−0.0012
3	−0.0081	−0.0084	−0.0078
4	−0.0081	−0.0083	−0.0078
5	−0.0012	−0.0012	−0.0012
6 (reservoir)	0.0015	0.0011	0.0200
7	−0.0081	−0.0083	−0.0078

is violated. There is almost no difference between NMPC and LMPC. There is no flooding for any of the channels with LMPC. Three channels are flooded with NMPC. However, these floodings can be neglected: the flood levels are violated with only 3 mm or less, and each flooding lasts for less than 1 min. The remaining buffer capacity in the water reservoir (Channel 6) is larger with NMPC than with LMPC. Adding a state estimator has only a minor negative influence on these results: the maximal flooding is increased to 2 cm, and each flooding lasts for less than 5 min. Once the disturbance is decreased again, the controllers first steer the water levels below the safety limits. Then, they keep the upstream water levels close to the safety limits (the opening of the first gate is decreased, and the second gate is completely closed), whereas the downstream water levels are further decreased below their initial set point because of the change in the matrix **W** and the set point **r**. To decrease these water levels as fast as possible, the controllers keep the downstream discharge at its maximal value while the opening of the third gate is as large as the surrounding water levels. This allows the buffer capacity of the water reservoir to be recovered quickly. Once the buffer capacity is recovered, the matrix **W** and the set point **r** are set to their initial value. The controllers close the second and the third gates to prevent the water levels in the reservoir to increase. The controllers immediately bring the upstream and the downstream water levels to their original set point by opening the first gate and changing $Q_{\text{out}}(t)$. The most significant negative influence of the state estimator can be found in the water level of the water reservoir at the end of the simulation: some part of the buffer capacity is not recovered. This can be seen in Table 4, in which the average difference between the water levels and their set point for each channel at the end of the simulation is shown. This difference for NMPC and LMPC is negligible for every channel. This is also the case when the Kalman filter is used except for the reservoir where there is a small difference of 2 cm.

To verify the generality of the results, a second simulation has been performed for a higher flood with a maximal discharge of 24 m^3/s in combination with a larger reservoir. All the other parameters of the controller, the state estimator, and the river system remained unchanged. The simulation results for this second test case are very similar as the results discussed previously. This indicates that the proposed control scheme can handle different sizes of floodings and reservoirs.

Table 5 contains the mean, the minimal, and the maximal computation times needed for computing the control actions for each step (prediction, linearization, optimization, conversion) separately and the total time by the three controllers. Because NMPC performs the predictions by using a nonlinear model, the time needed to perform this step is much larger than that required for LMPC and LMPC with the Kalman filter. The time needed to solve the optimization problem with the NMPC controller is much larger than that required by the other controllers because of the need to perform several iterations to achieve convergence. The relevance of the conversion step on the total computation time needed by any of the controllers is practically negligible.

J. Irrig. Drain Eng. 2013.139:532-541.

Table 5. Average, Minimal, and Maximal Computation Time (in Seconds) Needed for Each Step Separately and All Together by Three Controllers at Every Time Step

	NMPC			LMPC			LMPC + Kalman		
Steps	Mean	Min.	Max.	Mean	Min.	Max.	Mean	Min.	Max.
Prediction (s)	17.60	7.06	21.81	0.09	0.05	0.12	0.09	0.06	0.14
Linearization (s)	0.06	0.06	0.08	—	—	—	—	—	—
Optimization (s)	6.26	4.53	10.17	4.98	3.50	8.41	5.07	3.31	7.76
Conversion (s)	<0.01	<0.01	0.01	<0.01	<0.01	0.01	<0.01	<0.01	0.01
Total (s)	23.93	12.23	30.59	5.07	3.60	8.50	5.16	3.54	7.69

Note: Simulation was performed on a personal computer with a 2.8 GHz Intel Core i7 CPU and a random-access memory of 8 GB; Min. = minumum; Max. = maximum.

Table 6. Quantification of Total Amount of Control Actions for Three Controllers

Control variable	NMPC	LMPC	LMPC + Kalman
Gate 1 (m)	8.04	4.39	5.47
Gate 2 (m)	3.39	2.29	1.72
Gate 3 (m)	14.16	12.39	15.08
Total gate movement (m)	25.59	19.06	22.26
$\gamma_{Q_{\text{out}}}$ (m^3)	257,936	258,504	255,667

Table 6 shows the quantification of the total amount of control actions for the three controllers. For each gate m, the total gate movement is calculated with the formula

$$\gamma_{\text{gate}_m} = \sum_{k=0}^{N_T-1} |u_m(k+1) - u_m(k)| \tag{38}$$

where N_T = total number of time instants during the simulation. For Q_{out}, the total amount of control action is quantified with the integral of the absolute value of the changes of Q_{out} with respect to its initial value:

$$\gamma_{Q_{\text{out}}} = \int_0^T |Q_{\text{out}}(t) - Q_{\text{out}}(0)| dt \tag{39}$$

where T = end time of the simulation. This indicates by how much the controller let Q_{out} deviate from its normal operation point. The higher these numbers, the more demanding the controller is for the hydraulic structures. The third gate has been used the most by the three controllers. This is for releasing the excess water in the water reservoir. Overall, the NMPC changes the gate opening more frequently than the other two controllers, especially the first gate. The difference in total gate movement for LMPC without or with the Kalman filter is mainly explained by the third gate. This gate shows some oscillations when Kalman filtering is used, which are not present when the state estimator is absent. Because all the controllers use the Q_{out} in a very similar fashion, the difference for $\gamma_{Q_{\text{out}}}$ is negligible. One can always tune the controllers further to eliminate the oscillations or to reduce the total movement of the gates by decreasing the upper limit on the rate of change constraint ($\boldsymbol{\Delta}_{\max}$) or by increasing the value of the diagonal elements in the weight matrix $\mathbf{R}$. This has the advantage that the lifespan of the equipment can be increased, but it can at the same time increase the risk of flooding. One needs to make a trade-off between the life cycle of the structures and the risk of flooding.

Conclusions

This paper has shown that model predictive control can be used for set-point and flood control of river systems with multiple channels, gates, junctions, and a water reservoir. The optimization problem solved by MPC at every time step can be formulated such that the controller only uses the water reservoir when there is a risk of flooding. By modifying some of the parameters of the optimization problem, the same controller can be used to empty the water reservoir quickly when there is no risk of flooding. When the buffer capacity is restored, the controller automatically focuses on set-point control of the other water levels in the river system.

Given that the gate equations are the main sources of nonlinearities, the proposed controllers work with the gate discharges as control variables in the optimization problem instead of the gate positions. Then, a conversion step is performed to calculate the corresponding gate positions. Because the remaining equations of the nonlinear model (the Saint-Venant equations) can accurately be approximated with their linear counterpart (for the problem addressed in this paper) for designing the controllers, the LMPC achieves a similar control performance as the NMPC. The big advantage with LMPC is the reduction in computation time.

Adding a Kalman filter to the control loop to estimate the unmeasured states has only a minor influence on the control performance. Only a small increase in the offset of the water levels in the water reservoir was observed.

Future work will be focused on applying the techniques presented in this paper to a bigger river system, the Demer in Belgium, which has more water reservoirs, channels, gates and junctions. Because of the big number of states needed to model the behavior of this system, model reduction techniques will studied, e.g., proper orthogonal decomposition and Galerkin projection, and distributed control strategies.

Acknowledgments

The authors would like to thank the reviewers of this manuscript for their suggestions and remarks that undoubtedly contributed to the improvement of our initial submission.

Research was supported by Research Council KUL: GOA/10/09 MaNet, PFV/10/002 (OPTEC), FWO: Ph.D./postdoctoral grants and projects: G.0320.08 (convex MPC), G.0558.08 (Robust MHE), Belgian Federal Science Policy Office: IUAP P7/ (DYSCO, Dynamical systems, control and optimization, 2012-2017).

References

Barjas Blanco, T., Willems, P., Chiang, P. P.-K., Haverbeke, N., Berlamont, J., and De Moor, B. (2010). "Flood regulation using nonlinear model predictive control." *Control Eng. Pract.*, 18(10), 1147–1157.

Breckpot, M., Agudelo, O. M., De Moor, B. (2012a). "Control of a single reach with model predictive control." *River Flow 2012, Proc., Int. Conf. on Fluvial Hydraulics*, R. E. Murillo Muñoz, ed., CRC Press, Boca Raton, FL, 1021–1028.

J. Irrig. Drain Eng. 2013.139:532-541.

Breckpot, M., Agudelo, O. M., and De Moor, B. (2012b). "Model predictive control of a river system with two reaches." *Proc., 51st IEEE Conf. on Decision and Control*, IEEE, New York.
Breckpot, M., Barjas Blanco, T., and De Moor, B. (2010). "Flood control of rivers with nonlinear model predictive control and moving horizon estimation." *Proc., 49th IEEE Conf. on Decision and Control*, IEEE, New York, 6107–6112.
Burt, C. M., Mills, R. S., Khalsa, R. D., and Ruiz, V. (1998). "Improved proportional-integral (PI) logic for canal automation." *J. Irrig. Drain. Eng.*, 124(1), 53–57.
Chaudry, M. H. (2008). *Open-channel flow*, 2nd Ed., Springer, New York.
Chow, V. T. (1959). *Open-channel hydraulics*, McGraw-Hill, New York.
Clemmens, A. J., Bautista, E., Wahlin, B. T., and Strand, R. J. (2005). "Simulation of automatic canal control systems." *J. Irrig. Drain. Eng.*, 131(4), 324–335.
Cunge, J. A., Holly, F. M., and Verwey, A. (1980). *Practical aspects of computational river hydraulics*, Pitman, London.
Franklin, G. F., Powell, D. J., and Workman, M. L. (1997). *Digital control of dynamic systems*, 3rd Ed., Addison-Wesley, Boston, MA.
Henderson, F. M. (1966). *Open channel flow*, Macmillan, New York.
Hovd, M., and Braatz, R. D. (2001). "Handling state and output constraints in MPC using time-dependent weights." *Proc., American Control Conf. (ACC)*, IEEE, New York.
Kalman, R. E. (1960). "A new approach to linear filtering and prediction problems." *J. Basic Eng.*, 82, 35–45.
Kwakernaak, H., and Sivan, R. (1972). *Linear optimal control systems*, Wiley-Interscience, New York.
Liggett, J. A., and Cunge, J. A. (1975). *Numerical methods of solution of the unsteady flow equations, unsteady flow in open channels*, Chapter 4, Water Resources Publications, Fort Collins, CO.
Lin, C. H., Yen, J. F., and Tsai, C. T. (2002). "Influence of sluice gate contraction coefficient on distinguishing condition." *J. Irrig. Drain. Eng.*, 128(4), 249–252.
Litrico, X., and Fromion, V. (2009). *Modeling and control of hydrosystems*, Springer-Verlag, New York.
Litrico, X., Fromion, V., and Baume, J. P. (2006). "Tuning of robust distant downstream PI controllers for an irrigation canal pool—II. Implementation issues." *J. Irrig. Drain. Eng.*, 132(4), 369–379.
Malaterre, P. O., and Baume, J. P. (1999). "Optimum choice of control action variables and linked algorithms. Comparison of different alternatives." *Proc., Workshop on Modernization of Irrigation Water Delivery Systems*, USCID, Phoenix, 387–406.
Malaterre, P. O., Rogers, D. C., and Schuurmans, J. (1998). "Classification of canal control algorithms." *J. Irrig. Drain. Eng.*, 124(1), 3–10.
MATLAB 3 (2011a). [Computer software]. MathWorks, Natick, MA.
Mayne, Q., Rawlings, J. B., Rao, C. V., and Scokaert, P. O. M. (2000). "Constrained model predictive control: Stability and optimality." *Automatica*, 36(6), 789–814.
Negenborn, R. R., van Overloop, P.-J., Keviczky, T., and De Schutter, B. (2009). "Distributed model predictive control of irrigation canals." *Networks Heterog. Media*, 4(2), 359–380.
Nocedal, J., and Wright, S. J. (2000). *Numerical optimization*, Springer, New York.
Puig, V., et al. (2009). "Optimal predictive control of water transport systems: Arrêt-Darré/Arros case study." *Water Sci. Technol.*, 60(8), 2125–2133.
Qin, S., and Badgwell, T. (2003). "A survey of industrial model predictive control technology." *Control Eng. Pract.*, 11(7), 733–764.
Romera, J., Ocampo-Martnez, C., Puig, V., Quevedo, J., Garca, P., and Prez, G. (2011). "Flooding management using hybrid model predictive control at the Ebro River in Spain." *Proc., 8th IWA Symp. on System Analysis and Integrated Assessment*, IWA, San Sebastian, Spain, 1–8.
Rossiter, J. (2003). *Model-based predictive control*, CRC, Boca Raton, FL.
Schuurmans, J. (1997). "Control of water levels in open channels." Ph.D. thesis, Delft Univ. of Technology, Delft, Netherlands.
Scokaert, P. O. M., and Rawlings, J. B. (1999). "Feasibility issues in linear model predictive control." *AIChE J.*, 45(8), 1649–1659.
Sepúlveda, C., Gómez, M., and Rodellar, J. (2009). "Benchmark of discharge calibration methods for submerged sluice gates." *J. Irrig. Drain. Eng.*, 135(5), 676–682.
Strelkoff, T. S., and Falvey, H. T. (1993). "Numerical methods used to model unsteady canal flow." *J. Irrig. Drain. Eng.*, 119(4), 637–655.
Sturm, T. W. (2001). *Open channel hydraulics*. McGraw-Hill, New York.
Thai, T. (2005). "Numerical methods for parameter estimation and optimal control for the Red River Network." Ph.D. thesis, Universität Heidelberg, Heidelberg, Germany.
Van Overloop, P. J. (2006). "Model predictive control of open water systems." Ph.D. thesis, Technische Universiteit Delft, Delft, Netherlands.
Van Overloop, P. J., Clemmens, A. J., Strand, R. J., Wagemaker, R. M. J., and Bautista, E. (2010). "Real-time implementation of model predictive control on Maricopa-Stanfield irrigation and drainage district's WM canal." *J. Irrig. Drain. Eng.*, 136(11), 747–756.
Van Overloop, P.-J., Schuurmans, J., Brouwer, R., and Burt, C. M. (2005). "Multiple-model optimization of proportional integral controllers on canals." *J. Irrig. Drain. Eng.*, 131(2), 190–196.
Van Overloop, P.-J., Weijs, S., and Dijkstra, S. (2008). "Multiple model predictive control on a drainage canal system." *Control Eng. Pract.*, 16(5), 531–540.
Wahlin, B., and Clemmens, A. J. (2006). "Automatic downstream water-level feedback control of branching canal networks: Theory." *J. Irrig. Drain. Eng.*, 132(3), 208–219.
Xu, M., van Overloop, P. J., and van de Giesen, N. C. (2011). "On the study of control effectiveness and computational efficiency of reduced Saint-Venant model in model predictive control of open channel flow." *Adv. Water Resour.*, 34(2), 282–290.

J. Irrig. Drain Eng. 2013.139:532-541.