Time-instant optimization for hybrid model predictive control of the Rhine–Meuse delta

H. van Ekeren, R. R. Negenborn, P. J. van Overloop and B. De Schutter

ABSTRACT

In order to ensure safety against high sea water levels, in many low-lying countries, water levels are maintained at certain safety levels, and dikes have been built, while large control structures have been installed that can also be adjusted dynamically after they have been constructed. Currently, these control structures are often operated purely locally, without coordination of actions being taken at different locations. Automatically coordinating these actions is difficult, as open water systems are complex, hybrid dynamical systems, in the sense that continuous dynamics (e.g. the evolution of the water levels) appear mixed with discrete events (e.g. the opening or closing of barriers). In low lands, this complexity is increased further due to bi-directional water flows resulting from backwater effects and interconnectivity of flows in different parts of river deltas. In this paper, we propose a model predictive control (MPC) approach that is aimed at automatically coordinating the actions of control structures. The hybrid dynamical nature of the water system is explicitly taken into account. In order to relieve the computational complexity involved in solving the MPC problem, we propose TIO-MPC, where TIO stands for time-instant optimization. Using this approach, the original MPC optimization problem that uses both continuous and integer variables is transformed into a problem involving only continuous variables. Simulation studies of current and future situations are used to illustrate the behavior of the proposed scheme.

Key words | hybrid systems, model predictive control, open water systems, operational water management

H. van Ekeren
B. De Schutter
Delft Center for Systems and Control,
Delft University of Technology,
Delft,
The Netherlands

R. R. Negenborn (corresponding author)
Department of Marine and Transport Technology,
Delft University of Technology,
Delft,
The Netherlands
E-mail: *r.r.negenborn@tudelft.nl*

P. J. van Overloop
Department of Water Management,
Delft University of Technology,
Delft,
The Netherlands

INTRODUCTION

Floods are one of the most common types of natural disasters that Europe has to face. In the period between 1998 and 2004, there were more than 100 major floods in Europe. As a result, 700 people died, 250,000 people lost their home, and an economic loss of 25×10^9 euros was incurred. More recently, in 2010, several major floods (e.g. in Pakistan, China, Brazil and east/central Europe) showed again the importance of good flood prevention. Moreover, due to the changing climate, flood prevention is expected to become even more important, as sea levels rise, precipitation intensifies and river flows fluctuate more (Intergovernmental Panel on Climate Change (IPCC) 2007; Deltacommissie 2008). These changes will, in particular, affect low lands and river deltas, such as those found in The Netherlands.

One of the areas where increased problems are expected is the highly populated Rhine–Meuse delta in The Netherlands, including the large cities of Rotterdam and Dordrecht, and the largest port of Europe (see Figure 1) (Deltacommissie 2008). The Rhine–Meuse delta consists of a large number of rivers and sea outlets. The boundary in the east consists of the rivers Lek, Waal and Meuse. The boundary in the west consists of the connections of the Nieuwe Waterweg and the Hartelkanaal and one outlet, the Haringvliet, to the North Sea.

To protect this area against floods, dunes, dikes and storm surge barriers have been constructed. There are currently four main barriers (see Figure 2): the Maeslant barrier, the Hartel barrier, the Hollandsche IJssel barrier

doi: 10.2166/hydro.2013.177

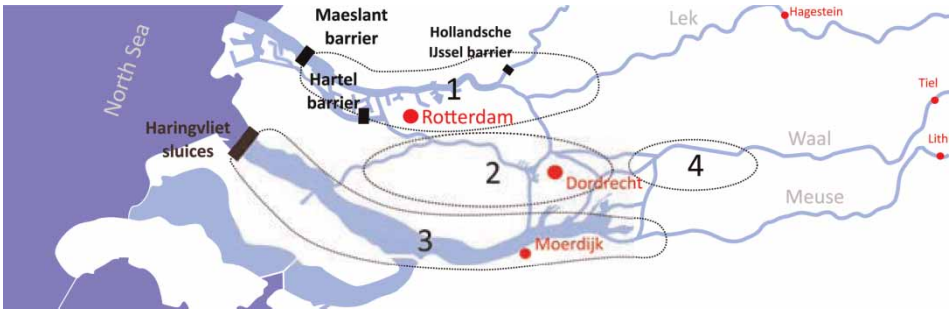

Figure 1 | The Rhine–Meuse delta and its division into four reservoirs. Reservoirs 1, 2 and 4 are located in the province of Zuid-Holland; reservoir 3 is located in the province of Zeeland.

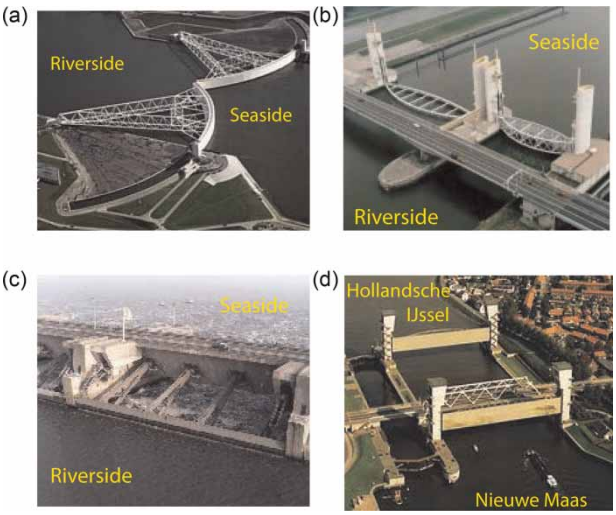

Figure 2 | Storm surge barriers in the Rhine–Meuse delta (van Overloop 2009). (a) Maeslant barrier, (b) Hartel barrier, (c) Haringvliet sluices and (d) Hollandsche IJssel barrier.

and the Haringvliet sluices. The first three barriers are designed to be either completely open or completely closed. In the open state, the rivers in which these barriers are built can flow freely, and ships can pass without any disturbance. In the closed state, river flows and navigation are blocked. The last barrier consists of 17 gates that can move independently between a maximum and a minimum height. At this point, ships can pass via a lock.

Whereas the dunes and dikes cannot be changed from minute to minute after they have been constructed, it is possible for the storm surge barriers to be changed (see Figure 2). Each barrier therefore has a local control system that determines when the barrier should be closed or opened. Locally, these rules may work well; however, since they do not take into account actions taken by other control structures, no guarantees can be given in the event of large disturbances. Therefore, the Delta commission, established by the Dutch government in September 2007 strongly advises investigation of the methods in which adequate operation of the water system can be ensured in the future. One of these plans, recommendation 10 of the Delta commission, is called 'Afsluitbaar Open Rijnmond' (Deltacommissie 2008). The idea of this plan is to build extra barriers that can block the river flows of the Lek, the Rhine and the Meuse. Together with the existing barriers at the sea side of the Rhine–Meuse delta, the complete Rotterdam/Dordrecht area can then be isolated from all disturbances. By closing the new barriers, the water of the rivers is directed to the south, towards the Haringvliet. Another plan, recommendation 8 of the Delta commission, is to use water bodies in Zeeland (the Grevelingen, the Volkerak and possibly the Oosterschelde) as temporary water storage bodies for water in critical situations (Deltacommissie 2008). The two plans merged together are depicted in Figure 3. Moreover, the Delta commission advises making the water system more flexible and operating it more dynamically. In this article, we investigate how coordination of the actions of the existing and the new structures, as well as the use of temporary storage could be realized.

The control goal in the Rhine–Meuse delta can be formulated as balancing the trade-off between keeping water levels low and minimizing the cost associated with using the storm surge barriers. The storm surge barriers and the water system of the Rhine–Meuse delta can be considered as hybrid dynamical systems, i.e. systems in which continuous dynamics and discrete events interact. In this case, some of the barriers are designed to move at once into a fully opened or fully closed state, and dikes can overflow (discrete events), while at the same time, other barriers are

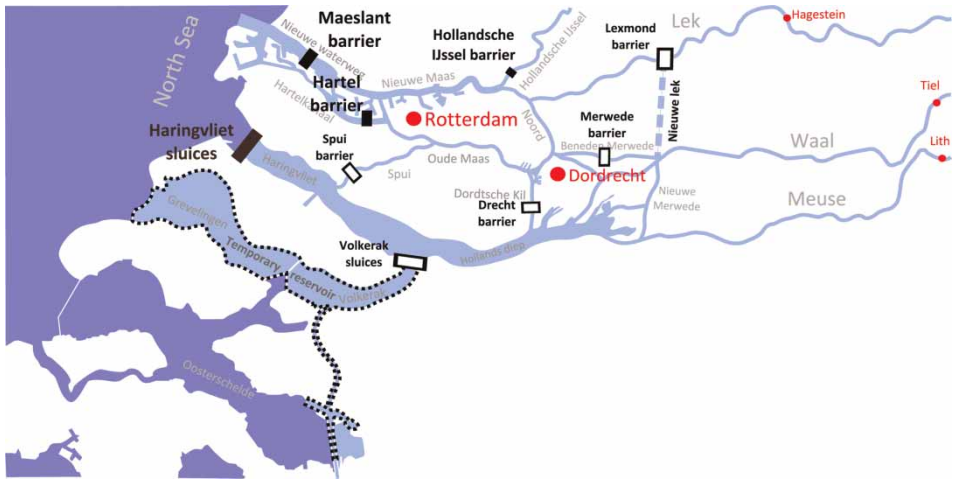

Figure 3 | A future situation of the Rhine–Meuse delta water system. The Volkerak sluices and the temporary reservoir, as well as the Spui, Drecht, Merwede and Lexmond barriers, provide new control options.

operated with continuous actions, and the water system itself involves continuous evolution of water flows and levels (continuous dynamics).

The decision whether or not to close the barriers depends on water levels, water flows and weather conditions in the near future. Currently, local rule-based control is used for controlling this system, although several more advanced methods have been proposed in the literature for control related systems. Several previously published works focus on the use of different types of optimization problem solvers for determining which action to take. For example, genetic or evolutionary search methods are used in Nixon *et al.* (2001) and Farmani *et al.* (2006), and Reed *et al.* (2001) and Farmani *et al.* (2006) discuss multi-objective optimization approaches. Other work considers the problem of deciding which action to take from a system and control perspective. Malaterre *et al.* (1998) give an overview of methods founded in control theory, including feedback control, feedforward control and combinations of feedback and feedforward control.

We propose to use model predictive control (MPC), an optimization-based control technique originally proposed in the process industry (Camacho & Bordons 2004), that combines feedback and feedforward control, and which is now also gaining increasing attention in other fields, including the field of open water systems. MPC is designed for handling multi-variable systems, constraints and multiple objectives, and therefore seems promising for also coordinating actions in water systems. Recently, van Overloop *et al.* (2008) proposed MPC for the control of a drainage canal system, Barjas Blanco *et al.* (2008) proposed MPC for control of a part of the river Demer in Belgium, Wahlin & Clemmens (2006) discussed the use of an MPC controller for a network of branching canals, Ruiz & Ramirez (1998) considered an MPC scheme for control of irrigation canals, Gómez *et al.* (2002) proposed a decentralized scheme for control of such a system, Negenborn *et al.* (2009) proposed a distributed MPC scheme, Begovich *et al.* (2007) discussed the actual application of MPC for a physical prototype of an irrigation canal and Malaterre & Rodellar (1997) focused on design and evaluation aspects of an MPC controller for a two-pool system.

Although MPC of water systems has been proposed previously, the existing applications of MPC for open water systems, however, do not consider the hybrid dynamical nature that can be present explicitly. Here, we propose to use an MPC technique that explicitly takes into account the hybrid dynamical nature of the system, a so-called *hybrid* MPC technique. We propose, in particular, the TIO-MPC technique, where TIO stands for time-instant optimization. Contrary to other MPC techniques for control of hybrid dynamical systems (such as the well-known MPC technique based on the mixed-logical dynamic (MLD) modeling framework (Bemporad & Morari 1999; Morari & Baric 2006)), TIO-MPC optimizes time instants. The MPC optimization problem originally involving both discrete and continuous variables is thereby transformed into an optimization problem with only

continuous variables. This has an advantage that computational time requirements are reduced, as mixed-integer nonlinear optimization problems are generally more complex to solve (Köppe 2012). Such a technique has been previously used for traffic control (De Schutter & De Moor 1998); here, we investigate its use for water control.

The contributions of this article include the following:

- A generic description of TIO-MPC, an MPC approach that uses the technique of TIO for control of hybrid dynamical systems, is proposed.
- A hybrid dynamical model that can be used for simulating and predicting the dynamics in the Rhine–Meuse delta for the current situation and a possible future situation based on recommendations of the Delta commission (Deltacommissie 2008) is proposed.
- The behavior of the use of TIO-MPC is illustrated when controlling the Rhine–Meuse delta water system in current and possible future situations.

This article extends the previously published conference article (van Ekeren *et al.* 2011) by providing more background and context on the control problem and the current control approaches available in the considered area, as well as more details of the control approach used. In addition, the hybrid dynamical model is presented for both current and possible future situations, and more simulation scenarios are provided to illustrate the behavior of the proposed scheme.

This article is organized as follows. First, the dynamics in the Rhine–Meuse area are formalized in a hybrid dynamical model, and a discussion of the currently used local control systems is given. Then, the details of MPC using TIO are discussed, and a TIO-MPC controller is designed for the area under study. The behavior of the proposed approach is subsequently illustrated in simulation studies representing current and future situations. Conclusions and directions for future research end this article.

RHINE–MEUSE DELTA

Model of the dynamics

We first describe the model of the dynamics for the current situation in the Rhine–Meuse delta. This model builds upon the model proposed in Roeleveld (2007) by including a more detailed representation of the Maeslant barrier, the Hartel barrier and the Haringvliet sluices. We then describe an extension of this dynamical model to represent a possible future setup of the Rhine–Meuse delta.

Later, the dynamical model will be used for representing reality (i.e. as a simulation model) and for representing inside the proposed MPC controller (i.e. as a model for making predictions). This model will be evaluated many times during the process of action determination. It is therefore necessary that the dynamical model is as fast and compact as possible. The model used is of the diffusive wave type and hence does not describe the full dynamic Saint-Venant equations. This reduces the accuracy of the model and, inherently, the calculated control actions. The same can be said about a large time step or even the assumption of one-dimensional flow and perfect mixing of fresh and saline water. It is important to realize, though, that the model requires sufficient accuracy while still being fast enough to be used in the real-time optimization inside the MPC controller. Moreover, the MPC controller features a strong feedback functionality where, at each control time step, the states in the internal model are taken as the actual states. This feedback reduces the uncertainty in the internal model. In Roeleveld (2007), the trade-off between model reduction and computational time has been extensively studied, resulting in the lumped hydraulic model that still captures the main dynamics relevant for control as described below.

Dynamical model of the current situation

The study area is represented by four large reservoirs that are interconnected by rivers, see Figures 1 and 4. The states x_1, x_2, x_3 and x_4 represent the water levels in reservoirs 1, 2, 3 and 4, respectively. The change in each of these water levels is determined using a discretized mass balance equation:

$$\begin{aligned} x_1(k+1) = x_1(k) + \frac{T_s}{A_{s1}(u_{\text{hijb}}(k))}[&q_{12}(x_1(k), x_2(k)) \\ &+ q_{1d}(k) - q_{nw}(x_1(k), h_{hvh}(k), u_{mb}(k)) \\ &- q_{hk}(x_1(k), h_{hvh}(k), u_{hb}(k))] \end{aligned} \tag{1}$$

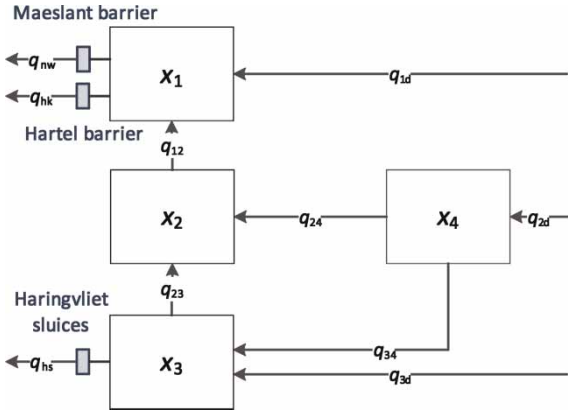

Figure 4 | Structure of the Rhine–Meuse delta model.

$$x_2(k+1) = x_2(k) + \frac{T_s}{A_{s2}}[-q_{12}(x_1(k), x_2(k)) + q_{23}(x_2(k), x_3(k)) + q_{24}(x_2(k), x_4(k))] \quad (2)$$

$$x_3(k+1) = x_3(k) + \frac{T_s}{A_{s3}}[-q_{23}(x_2(k), x_3(k)) + q_{34}(x_3(k), x_4(k)) + q_{3d}(k) - q_{hs}(x_3(k), h_{hs}(k), u_{hs}(k))] \quad (3)$$

$$x_4(k+1) = x_4(k) + \frac{T_s}{A_{s4}}[-q_{24}(x_2(k), x_4(k)) - q_{34}(x_3(k), x_4(k)) + q_{2d}(k)] \quad (4)$$

where k is a discrete time step; T_s (s) is the sample time; $A_{s1}(\cdot)$, A_{s2}, A_{s3} and A_{s4} (m^2) are the surface areas of reservoir 1, 2, 3 and 4, respectively; $x_1(k)$, $x_2(k)$, $x_3(k)$ and $x_4(k)$ (m) are the water levels of reservoir 1, 2, 3 and 4, respectively; $q_{1d}(k)$, $q_{2d}(k)$, and $q_{3d}(k)$ (m^3/s) are disturbance inflows from the rivers Lek, Waal and Meuse, respectively; $q_{nw}(\cdot)$, $q_{hk}(\cdot)$ and $q_{hs}(\cdot)$ (m^3/s) are disturbance inflows from the sea, controlled by the Maeslant barrier, the Hartel barrier and the Haringvliet sluices, respectively; $q_{12}(\cdot)$, $q_{23}(\cdot)$, $q_{24}(\cdot)$ and $q_{34}(\cdot)$ (m^3/s) are the flows between the reservoirs, described by $q_{ij}(\cdot) = f_{Chézy}(\cdot)$, where $f_{Chézy}$ is the Chézy formula (Brouwer 2001):

$$f_{Chézy}(x_i(k), x_j(k)) = A_{c,ij}(x_i(k), x_j(k)) C_{ij}\,\mathrm{sign}(x_j - x_i)\sqrt{\frac{R_{ij}(x_i(k), x_j(k))|x_j - x_i|}{l_{ij}}} \quad (5)$$

where $q_{ij}(\cdot)$ is the flow between reservoirs i and j; $A_{c,ij}(\cdot)$ (m^2) is the (smallest) water cross-section of the transport region of flow $q_{ij}(\cdot)$; C_{ij} ($m^{1/2}/s$) is the Chézy roughness coefficient of flow $q_{ij}(\cdot)$; $R_{ij}(\cdot)$ (m) is the hydraulic radius of flow $q_{ij}(\cdot)$ and l_{ij} (m) is the length of the river between reservoir i and reservoir j. Note that in (5), the sign function is used to indicate the direction of the flow.

The water cross-sectional area $A_{c,ij}(\cdot)$ and the hydraulic radius $R_{ij}(\cdot)$ are variables that depend on the water level in the river that connects reservoir i with reservoir j. This river water level is approximated by $(x_i + x_j)/2$, the average of the water levels of reservoirs i and j. The water cross-sectional area and the hydraulic radius depend also on the physical structure of the river cross-section. The river cross-sections are approximated with straight lines. As a result, $A_{c,ij}(\cdot)$ and $R_{ij}(\cdot)$ are nonlinear functions of $x_i(k)$ and $x_j(k)$. For more details, see Roeleveld (2007).

The flows $q_{nw}(\cdot)$ and $q_{hk}(\cdot)$ are determined using (5):

$$q_{nw}(x_1(k), h_{hvh}(k), u_{mb}(k)) = u_{mb}(k) f_{Chézy}(x_1(k), h_{hvh}(k)) \quad (6)$$

$$q_{hk}(x_1(k), h_{hvh}(k), u_{hb}(k)) = u_{hb}(k) f_{Chézy}(x_1(k), h_{hvh}(k)) \quad (7)$$

where $h_{hvk}(k)$ (m) is the water level of the North Sea at Hoek van Holland and $u_{mb}(k)$ represents the state of the Maeslant barrier, defined as:

$$u_{mb}(k) = \begin{cases} 0 & \text{if the barrier is closed at time step } k \\ 1 & \text{otherwise} \end{cases} \quad (8)$$

The state of the Hartel barrier is represented by $u_{hb}(k)$ and the state of the Hollandsche IJssel barrier $u_{hijb}(k)$ is defined similarly.

The flow $q_{hs}(\cdot)$ through the Haringvliet sluices depends on the water level of the North Sea near these sluices $h_{hs}(k)$, the water level $x_3(k)$ in reservoir 3 and the opening height of the gates of the sluices $u_{hs}(k)$. The flow is determined by using the equations for free and submerged orifice flow and the equations for free and submerged weir flow, as given in Roeleveld (2007).

The effect of the Hollandsche IJssel barrier is modeled via the surface area of reservoir 1 as:

$$A_{s1}(u_{hijb}(k)) = A_{s1,normal} - A_{s1,hij}(1 - u_{hijb}(k)) \quad (9)$$

where $A_{s1,normal}$ is the surface area of reservoir 1 (m^2) when the Hollandsche IJssel barrier is open ($u_{hb}(k) = 1$) and where $A_{s1,hij}$ is the reduction in surface area caused by closure of the Hollandsche IJssel barrier ($u_{hb}(k) = 0$).

Dynamical model of a future situation

In the considered future situation, five new barriers are present: the Lexmond barrier, the Merwede barrier, the Drecht barrier, the Spui barrier and the Volkerak sluices. All these barriers are modeled as barriers that can be in two discrete modes: fully open or fully closed. The dynamical model presented above for the current setup is extended to create a model for this future setup. A schematic view of this extended model is shown in Figure 5. The two water storage bodies in Zeeland are modeled by adding an extra reservoir, reservoir 5. By opening the Volkerak sluices, this additional reservoir is filled with water from reservoir 3. The dynamical model for the future setup therefore has one extra mass balance equation:

$$x_5(k+1) = q_{35}(x_3(k), x_5(k), u_{vs}(k))\frac{T_s}{A_{s5}} + x_5(k) \quad (10)$$

where $q_{35}(\cdot)$ is the flow through the Volkerak sluices in m^3/s; A_{s5} is the surface area of reservoir 5 in m; and $u_{vs}(k)$ is the state of the Volkerak sluices. The flow $q_{35}(\cdot)$ through the Volkerak sluices is determined in a similar way to how the flow $q_{hs}(\cdot)$ is determined. To take into account the addition of the Volkerak sluices, the mass balance (3) of reservoir 3 is changed into:

$$\begin{aligned} x_3(k+1) = [&- q_{23}(x_2(k), x_3(k), u_{sdb}(k)) \\ &+ q_{34}(x_3(k), x_4(k)) + q_{3d}(k) \\ &- q_{hs}(x_3(k), h_{hs}(k), u_{hs}(k)) \\ &- q_{35}(x_3(k), x_5(k), u_{vs}(k))]\frac{T_s}{A_{s3}} + x_3(k) \end{aligned} \quad (11)$$

where $u_{sdb}(k)$ is the state of the Drecht barrier and the Spui barrier (these barriers are restricted to open/closed simultaneously).

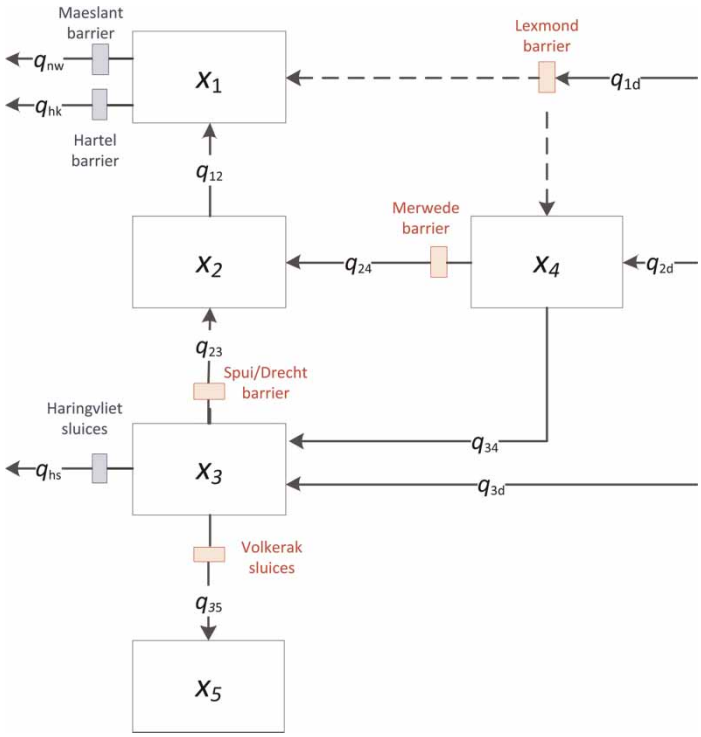

Figure 5 | Structure of the Rhine–Meuse delta model for the future setup.

The Lexmond barrier is modeled as a barrier that can direct the disturbance inflow $q_{1d}(k)$ to reservoir 4 instead of to reservoir 1. Therefore, the mass balances (1) and (4) of the dynamical model of the current setup are replaced by:

$$x_1(k+1) = [q_{12}(x_1(k), x_2(k)) + q_{1d}(k)u_{1b}(k) - q_{nw}(x_1(k), h_{hvh}(k)u_{mb}(k)) \quad (12)$$

$$- q_{hk}(x_1(k), h_{hvh}(k), u_{hb}(k))]\frac{T_s}{A_{s1}(u_{hijb}(k))} + x_1(k) \quad (13)$$

$$x_4(k+1) = [-q_{24}(x_2(k), x_4(k), u_{mwb}(k)) - q_{34}(x_3(k), x_4(k)) + q_{1d}(k)u_{lb}(k) + q_{2d}(k)]\frac{T_s}{A_{s4}} + x_4(k) \quad (14)$$

where $u_{mwb}(k)$ is the state of the Merwede barrier and $u_{lb}(k)$ is the state of the Lexmond barrier, which is similarly defined as $u_{mb}(k)$ in (8).

The Spui barrier and the Drecht barrier can together block the flow from reservoir 2 to reservoir 3. The Merwede barrier can block the flow from reservoir 2 to reservoir 4. These flows $q_{23}(\cdot)$ and $q_{24}(\cdot)$ are influenced by the state of these barriers as follows:

$$q_{23}(x_2(k), x_3(k), u_{sdb}(k)) = u_{sdb}(k)f_{Chézy}(x_2(k), x_3(k)) \quad (15)$$

$$q_{24}(x_2(k), x_4(k), u_{mwb}(k)) = u_{mwb}(k)f_{Chézy}(x_2(k), x_4(k)) \quad (16)$$

Constructing the Volkerak sluices and the temporary reservoir in Zeeland requires relatively little adjustment of the existing infrastructure. Constructing the other components requires significantly more investments. Therefore, in the simulation section below, we will focus on the current situation extended with the Volkerak sluices and the temporary reservoir.

Currently used control systems

The control systems currently used for the barriers consist of simple if–then–else rules. The goal of the local controllers is to achieve the following objectives (van Overloop 2009):

1. To prevent the water level at Rotterdam rising above 3.87 mNAP (m Normaal Amsterdams Peil, i.e. m above the Dutch mean sea level) and at Dordrecht to rise above 3.25 mNAP.
2. To prevent water levels in the Hollandsche IJssel to rise above 2.25 mNAP, while preventing saline water to flow into this river.
3. To maintain a minimum water level of 0.00 mNAP at Moerdijk (in the Hollandsche Diep).
4. To maintain a minimum discharge, averaged over a tide, through the Nieuwe Waterweg of 1,500 m^3/s.
5. To prevent water flowing directly from the North Sea into the Haringvliet.

When the water levels at Rotterdam and Dordrecht stay below their critical value (i.e. the dike height), the whole area is safe, since the most critical (i.e. lowest) dikes are located at these locations (Roeleveld 2007). Currently, the Maeslant barrier and the Hartel barrier are used mostly for objective 1; the Hollandsche IJssel barrier for objective 2; and the Haringvliet sluices for objectives 1 and 3–5.

Maeslant barrier

The Maeslant barrier (Figure 2(a)) is a storm surge barrier that can completely block the Nieuwe Waterweg. The Nieuwe Waterweg is a connection between the North Sea and the port of Rotterdam. It acts as a highway for large ocean vessels. Closing the Nieuwe Waterweg can lead to large economic losses (of the order of a million euros per day). The Maeslant barrier consists of two large gates (about 240 m each) that are usually located in two dry docks, one on each side of the river. The control system of the Maeslant barrier determines when the gates are floated out of the dry docks and are sunk down to close the Nieuwe Waterweg.

The control system of the Maeslant barrier currently consists of one simple rule subject to some constraints (van Overloop 2009). Every 10 min, the control system makes a prediction of the water levels over the coming 24 hours based on actual weather forecasts (Rijkswaterstaat 2009). The closing rule of the Maeslant barrier based on these predictions is as follows: if a water level of 3.00 or 2.90 mNAP is predicted for, respectively, Rotterdam or Dordrecht (usually caused by a storm at sea), the closing

procedure of the Maeslant barrier is started. The closing procedure starts by filling the dry docks of the gates with water. The gates are then floated out of the dry docks directly after filling of the dry docks in situations when the discharge of the river Rhine measured at Lobith is higher than 6,000 m^3/s and the water level at the structure exceeds 2.00 mNAP. Otherwise, the gates are floated out of the dry docks when the direction of the flow at the structure changes from downstream into upstream. (Lobith is a measurement location in The Netherlands close to the border with Germany. Depending on the flow, it takes up to 48 hours for the water at Lobith to reach the Rhine–Meuse delta. Therefore, it is a (rough) measure of the amount of river water that will flow into the Rhine–Meuse delta in the near future.)

The two gates are sunk when the water level at the sea side is 0.05 m higher than the water level on the river side. During the sinking of the gates, the flow potential under these gates is monitored. The flow potential is used as an indication of the rate of erosion at the floor structure. If the underside of the gates is 5 m above the floor and if the flow potential is too high, the sinking is stopped.

With the gates sunk to the bottom, the Nieuwe Waterweg is closed. In the gates, there are ballast tanks. The amount of water in these tanks is controlled in order to float up the gates as soon as the water level at the river side equals or surpasses the water level at the sea side. It takes 60 min to lift the gates up. After this procedure, the gates float on the river and do not close the Nieuwe Waterweg anymore. If the predictions of the water levels at Rotterdam and Dordrecht are not critical, the gates are floated back into their dry docks.

The implementation of the control system of the Maeslant barrier that is used in this article for simulations is a simplified version of the control system used in reality. The model of the Rhine–Meuse delta that is used models the Maeslant barrier (and the same holds for the Hartel barrier and the Hollandsche IJssel barrier), as if it can be either open or closed. Thus, the closing/opening procedures are neglected. This means that the constraints on the closing/opening procedures are not considered. It is assumed that the closing procedure is started early enough to let the physical closing coincide with the start of the closed state of the barrier in the model. The implemented control system of the Maeslant barrier is therefore the following: if a water level of 3.00 m or 2.90 mNAP is predicted for, respectively, Rotterdam or Dordrecht in the coming 24 hours, the Maeslant barrier is closed as soon as the water level at the sea side (i.e. the water level at Hoek van Holland) is 0.05 m higher than the water level on the river side (i.e. the water level at Rotterdam). The Maeslant barrier is opened again as soon as the water level at the river side equals or surpasses the water level at the sea side.

Hartel barrier

The Hartel barrier (Figure 2(b)) is situated in the Hartelkanaal, parallel to the Nieuwe Waterweg. In comparison with the Nieuwe Waterweg, this is a small canal. Unlike in the Nieuwe Waterweg, no navigation of large ocean vessels is possible in the Hartelkanaal. The Hartel barrier consists of two gates that are normally lifted high above the water.

The control system of the Hartel barrier consists of the same closing rule and similar constraints as for the Maeslant barrier (van Overloop 2009). However, the closing procedure is different. When a critical water level is predicted, the gates are lowered until the undersides of the gates are at 3.50 mNAP. This takes about 10 min. The start of lowering the gates to close the Hartelkanaal depends on the same constraints as for the closing of the Maeslant barrier, i.e. when the discharge at Lobith is high and the water level at the structure is higher than 2.00 mNAP.

With both gates lowered, the Hartelkanaal is completely closed. During the closure, the water levels on both sides of the barrier are monitored. When the water level at the river side equals or surpasses the water level at the sea side, the gates are lifted one after another. When a water level of 3.00 mNAP at Rotterdam or of 2.90 mNAP at Dordrecht is predicted to appear in the coming 24 hours, the gates stay in this lifted, not blocking, position. When the water level at the structure stays constant or increases continuously for half an hour, or increases at least 10 cm continuously, the gates will close the Hartelkanaal again. When the predicted water levels are lower, the gates are lifted up until the storm surge barrier has been completely opened.

As already explained above, the Hartel barrier is modeled as if it can be either fully open or fully closed. Therefore, constraints and procedures concerning the

opening and the closing procedures of the Hartel barrier are neglected. The implementation of the control system of the Hartel barrier for simulations is the same as the implementation used for the Maeslant barrier, as presented above.

Haringvliet sluices

The Haringvliet sluices (Figure 2(c)) are located in the former estuary of the Haringvliet (van Overloop 2009). The 17 sluices, each 56.5 m wide, determine how much water is exchanged between the North Sea and the Haringvliet. Each sluice consists of two gates, which makes it possible to block water from the sea side as well as from the river side. The gates can be lifted partially, which makes different discharges through the sluices possible. The Haringvliet sluices have three purposes: closing off the Haringvliet for flood prevention, keeping the water in the Haringvliet fresh by blocking the water of the North Sea and holding the water level at Moerdijk above 0.00 mNAP.

A control program called LPH'84 determines the height of the sluice gates based on the measured discharge of the river Rhine at Lobith (van Overloop 2009). The operation of this control program is depicted in Figure 6. The vertical axis in Figure 6 shows the area of the surface gate opening as a function of the river discharge. This area can be translated straightforwardly into gate heights by using the width of the gates. The sluices are only opened when the water pressure at the river side is larger than the water pressure at the sea side. When the discharge sluices are open and the water pressure at the sea side becomes larger than the water pressure at the river side, the gates are closed. As a consequence, the sluices are closed almost every high tide. Opening or closing the gates takes about 20 min.

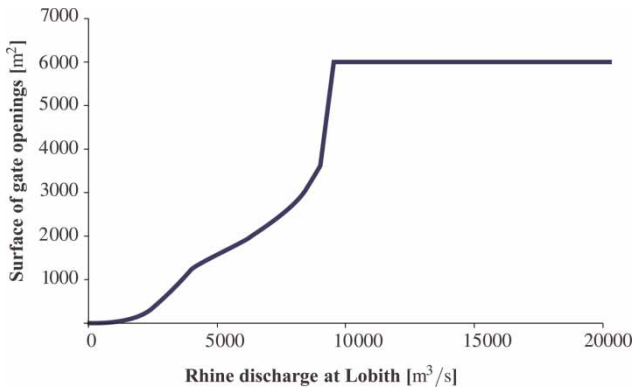

Figure 6 | The relation that determines the gate openings of the Haringvliet sluices depending on the discharge of the river Rhine at Lobith (van Overloop 2009).

The implementation of the control system of the Haringvliet sluices for simulations is a translation of the description presented in this section.

Hollandsche IJssel barrier

The Hollandsche IJssel barrier (Figure 2(d)) consists of two large gates in series. The gates can close off the mouth of the river Hollandsche IJssel. This protects the low-level area behind the barrier against high water levels in the river Nieuwe Maas.

The two large gates of the barrier close the Hollandsche IJssel when a water level of 2.25 mNAP is measured at the Nieuwe Maas. The procedure for lowering the gates starts directly when this critical water level is measured. The gates are lifted when the water pressure at the side of the Hollandsche IJssel becomes larger than the water pressure at the side of the Nieuwe Maas.

The implementation of the control system of the Hollandsche IJssel barrier for simulations is a translation of the description above.

TIO-MPC

The currently used control systems described in the previous section operate independently of one another. There is no coordination between the control systems at the different structures. Hence, the evolution of the water levels when different control actions determined by the current control systems are applied in the same period is not taken into account. This may lead to low performance of the local control systems in extreme conditions. The control system proposed here could improve this performance by employing MPC as the control strategy.

MPC

Using an MPC approach, the control goal to be achieved is described mathematically. The computational power we

have nowadays is then used to find optimal control actions with respect to this control goal. However, these control actions are only optimal with respect to the available information given to the optimizer. This information is, in practice, never completely correct (e.g. due to modeling errors and predictions errors) and only valid over a finite time window (prediction horizon). Therefore, at every control step, this optimization is repeated to obtain control actions that are as close to optimal as possible to a control system that works over an infinite time window. This is briefly how an MPC controller works.

The general MPC scheme is depicted in Figure 7. The MPC controller has three types of inputs. The first one is the input of the operator. The operator's task is to formulate the control goal in an objective function and to specify the operational constraints. The second input consists of the measurable or predictable disturbances. The MPC controller anticipates these a priori known disturbances. The last input is the current state of the system. The output of the MPC controller consists of the optimized control actions that are sent to the real system. Actions are implemented for the upcoming time step. The real system then transitions to a new state, and the MPC controller then determines new actions to be taken at the next time step.

As addressed in the Introduction, no publications exist on so-called hybrid MPC techniques for the field of large open water systems, i.e. MPC for systems comprising both discrete events and continuous dynamics. Hybrid MPC is a technique that is, in particular, interesting for this field because of the several hybrid dynamical aspects that are present. With hybrid dynamical aspects in water systems, we refer in this paper, to a combination of continuous dynamics of the water with discrete events. Examples of discrete events in water systems are: storm surge barriers that can be opened or closed, pumps that can switched to only work at maximum capacity or not at all, and different modes in which a water system will all of a sudden have different dynamics (e.g. depending on the weather conditions). Examples of continuous dynamics of water systems are the continuously changing water levels and water flows. Hybrid MPC explicitly deals with the combination of these continuous dynamics and discrete events.

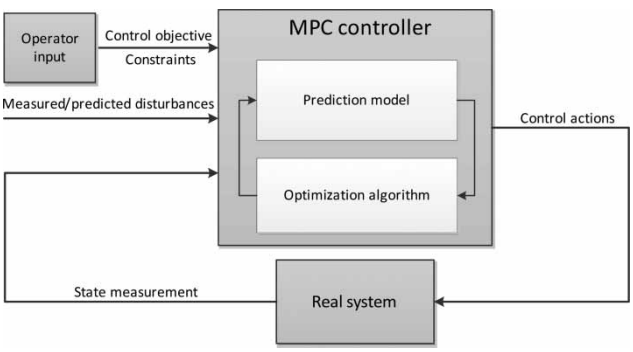

Figure 7 | General MPC scheme (Negenborn 2007).

One common method uses binary variables, which indicate at each time step whether a discrete event takes place (such as switching from open to closed or vice versa). Using these binary variables, the MPC optimization problem typically becomes a mixed-integer nonlinear optimization problem, generally requiring a significant computational load to solve (Köppe 2012), as the number of binary variables can be large. Below, we discuss how the discrete events can be dealt with in an alternative way: by using real-valued time instants at which events take place as decision variables, rather than binary variables that, at each time step, indicate the occurrence of an event.

The principle of TIO

Figure 8 is used to explain the principle of TIO-MPC. Figure 8(a) shows how a binary input (e.g. the state of the Maeslant barrier) would be modeled in the commonly used MLD modeling framework for hybrid dynamical systems. In the MPC optimization problem formulation, there would be a binary variable to model the state of the barrier as a function of time for each control step. Hence, when the prediction horizon is 16 hours and the control step is 1 hour, there are $16/1 = 16$ binary variables for that specific binary input signal. With a high enough number of binary inputs and complexity of the model, an optimization problem that cannot be solved in a reasonable time (within the control step length) will result. Therefore, the technique illustrated in Figure 8(b) is proposed. Instead of having a binary variable that indicates, for each control step, whether or not there are time instants that indicate at what time step an event takes

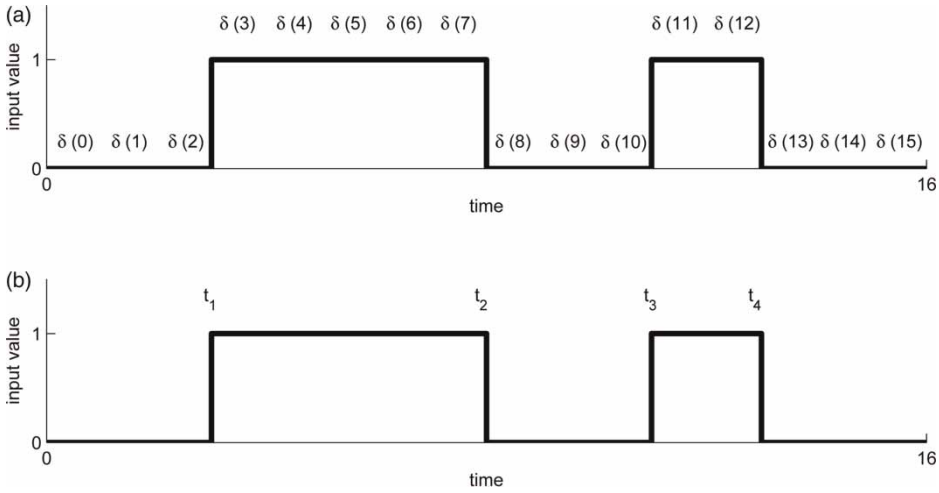

Figure 8 | MLD-MPC versus TIO-MPC. (a) MLD-MPC (b) TIO-MPC.

place. Hence, the link between the binary variables and the continuous variables is that a time instant t_i indicates the moment at which the value of a binary input changes its value. The time instants are continuous optimization variables, rather than binary variables as used in the MLD-MPC formulation. Moreover, the amount of optimization variables could be lower with this technique. If the number of time instants is lower than the number of control steps in the prediction horizon, the optimization problem has fewer degrees of freedom. For example, the event represented by the MLD-MPC binary variable of Figure 8(a) can happen 16 times, whereas the event represented by the TIO-MPC time instants of Figure 8(b) can take place at most twice.

Prediction model

TIO-MPC requires a model that maps the time instants, possibly additional 'regular' inputs (i.e. inputs that consist of a sequence of input variables, one variable for each (control) time step) and the actual state of the system to the future state/output variables of the system. The time instants can normally not directly be applied to the actuators of the system. Therefore, a converter is needed to translate the determined time instants into input sequences that are appropriate for the actuators.

The TIO prediction model can be formulated as:

$$\tilde{\mathbf{x}}(k) = f(\tilde{\mathbf{t}}(k), \tilde{\mathbf{u}}_r(k), \mathbf{x}(k)) \tag{17}$$

with

$$\tilde{\mathbf{x}}(k) = \left[\mathbf{x}^{\mathrm{T}}(k+1)\,\mathbf{x}^{\mathrm{T}}(k+2)\cdots\mathbf{x}^{\mathrm{T}}(k+N)\right]^{\mathrm{T}}$$
$$\tilde{\mathbf{u}}_r(k) = \left[\mathbf{u}_r^{\mathrm{T}}(k)\,\mathbf{u}_r^{\mathrm{T}}(k+1)\cdots\mathbf{u}_r^{\mathrm{T}}(k+N-1)\right]^{\mathrm{T}}$$
$$\tilde{\mathbf{t}}(k) = \left[t_{1,k}\,t_{2,k}\cdots t_{p,k}\right]^{\mathrm{T}}$$

where $\tilde{\mathbf{x}}(k)$ and $\tilde{\mathbf{u}}_r(k)$ contain all the state variables and all the regular input variables over the prediction horizon N, respectively. The vector $\tilde{\mathbf{t}}(k)$ consists of all the (continuous) time instants relative to kT_s to be optimized over the prediction horizon.

In general, there is no restriction on the prediction model structure as long as it relates to time instants and regular inputs to the state variables. One possible structure that is used for the case study below is given in Figure 9. The time instants $\tilde{\mathbf{t}}(k)$ are translated into regular input sequences similar to $\tilde{\mathbf{u}}_r(k)$ using a converter. These input sequences are used to calculate the state evolution $\tilde{\mathbf{x}}(k)$ with a discrete-time state-space model.

Optimization problem

The optimization problem to be solved at every time step of the TIO-MPC system has some specific properties.

- *Nonlinear optimization*. The model (17) is always a nonlinear model. This is a consequence of using time instants as input variables. The relations between the time instants and the future state values are nonlinear. These

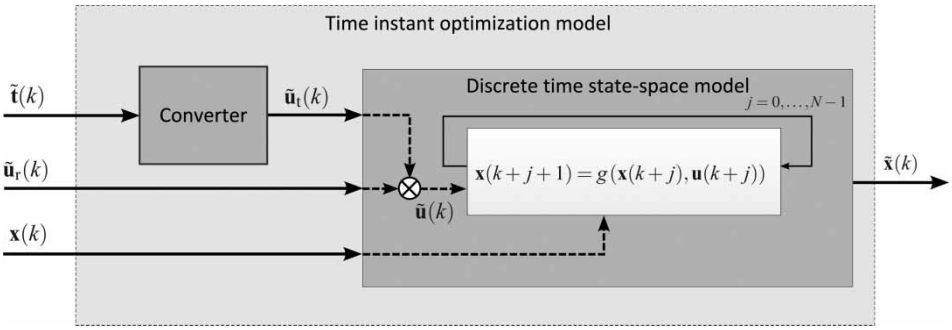

Figure 9 | A possible prediction model structure for TIO-MPC.

relations make it impossible to use fast linear or convex optimization algorithms.

- *Constraint optimization*. The optimization problem is always constrained because the time instants have a predefined order (e.g. $t_1(k) < t_2(k)$).
- *Derivative-free optimization*. In general, it is not possible to derive analytical expressions for the derivatives of the problem with respect to the time instants. Optimization algorithms, like sequential quadratic programming that use numerical approximations of derivatives by performing several function evaluations, are in practice not so efficient, due to the underlying model structure (Figure 9) of the TIO-MPC optimization problem. The discrete-time state-space model inside the model of (17) rounds the time instants to the closest discrete time step. This causes the objective function to be non-smooth. As a result, derivative-based methods cannot be employed.

Because of these three properties, we propose to use derivative-free or direct-search methods, such as pattern search (Lewis *et al.* 2000), which do not require any knowledge about the problem structure. By performing several function evaluations with different values for the optimization variables, such algorithms aim at finding an optimal solution.

A schematic representation of the TIO-MPC optimization is given in Figure 10. The optimization problem consists of the nonlinear model (17) and the objective function of the control system. Together, they form the function $J\{\tilde{\mathbf{t}}(k), \tilde{\mathbf{u}}_n(k)\}$ to be minimized.

It is not guaranteed that the optimum found by a direct-search optimization algorithm is the global optimum.

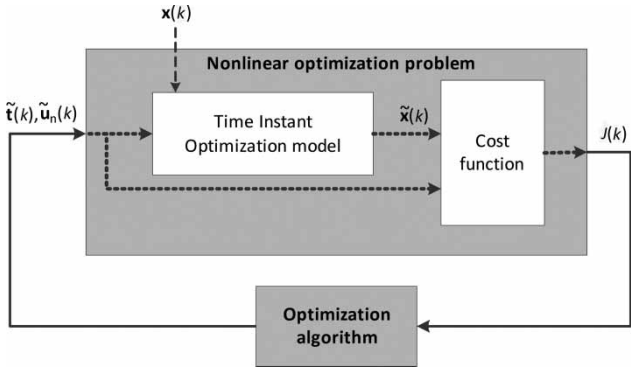

Figure 10 | Optimization scheme for TIO-MPC.

Frequently, such an algorithm converges to a local optimum. We therefore employ a multi-start approach. As an example of a direct-search method, we will now discuss multi-start pattern searches. This method is also used in the simulation experiments later on.

Multi-start pattern search

A basic, single-start pattern search searches at every iteration for an improved solution of the optimization problem. The search is done by exploring points (alternative solutions) around the current solution. These points, which are called mesh points, are created by adding a set of scaled vectors to the current point. The scaling is called the mesh size and the set of vectors is called the pattern. The pattern vectors could be random vectors, but are usually fixed-direction vectors. A possible set of pattern vectors for an optimization problem with two variables is:

$$[1 \quad 0]^{\mathrm{T}} \quad [0 \quad 1]^{\mathrm{T}} \quad [-1 \quad 0]^{\mathrm{T}} \quad [0 \quad -1]^{\mathrm{T}}$$

and is called the maximal basis. At every iteration, the function values of the mesh points are calculated one by one. Depending on the settings, all the mesh points are explored at every iteration, or an iteration finishes after an improved mesh point has been found. In the last case, the algorithm switches directly to this improved mesh point. At every iteration of the pattern search algorithm, there are two possibilities.

1. The search fails in finding an improvement of the current solution, i.e. the mesh points all have a higher cost function value. The current solution stays the same in the next iteration and a new search is performed. The mesh size for this new search is decreased by multiplying it with a contraction factor (a factor between 0 and 1). This results in mesh points closer to the current point.
2. The search finds an improvement of the current solution, i.e. one of the mesh points has a lower cost function value than the current point. The current point (solution) is replaced by this mesh point. A new search is performed around this new current solution. The mesh size of this new search is increased by multiplying it with an expansion factor (i.e. a factor larger than 1).

The solution at iteration i will always be at least equal or better than the one at previous iteration $i-1$. Therefore, the cost function value of the optimization problem will decrease until it converges close to a local (possibly global) optimum. The algorithm stops when the stopping criterium is satisfied (e.g. a minimum mesh size, a maximum number of function evaluations, a minimum change in function value between two successive iterations, a maximum number of iterations, or a time limit).

Pattern search in its basic single-start version does not guarantee finding a globally optimal solution. Large-scale nonlinear problems can have a large number of local optima. Therefore, we propose multi-start pattern search as the TIO-MPC optimization method. The pattern search algorithm is then started several times from different initial points. These points can be generated randomly, but good initial guesses based on expert knowledge can make the convergence of the algorithm considerably faster. The probability of finding the global optimum increases when the number of starting points increases. However, there is a finite amount of time to perform the optimizations because the TIO-MPC controller has to apply the control actions after the control time is over. The multi-start optimizations can be performed in series (one after another), but also in parallel by using several processors. Either way, the pattern search optimizations are restarted several times until the maximum optimization time is reached and possibly with several processors in parallel. The best of the local optima is selected and applied to the system.

TIO-MPC design for the Rhine–Meuse delta

To design a TIO-MPC controller for the Rhine–Meuse delta, we have to define the objective function, the prediction model, the constraints and the solution method. We explicitly provide the design of the controller for the current situation. The design of the controller for the future situation considered later on in the simulation studies is straightforward (primarily involving the replacement of the prediction model and the adequate redefinition of the vectors and matrices.)

Objective function

Input effort and costs on (too) high water levels are considered as control objectives. We adopt the commonly used weighted-sum strategy to obtain an objective function that merges these objectives into one combined objective. For time step k, the objective function is defined as:

$$J(k) = J_{x_1}(\tilde{\mathbf{x}}_1(k)) + J_{x_2}(\tilde{\mathbf{x}}_2(k)) + J_{\mathrm{mb}}(\tilde{\mathbf{u}}_{\mathrm{mb}}(k)) + J_{\mathrm{hb}}(\tilde{\mathbf{u}}_{\mathrm{hb}}(k)) + J_{\mathrm{hs}}(\tilde{\mathbf{u}}_{\mathrm{hs}}(k)) \quad (18)$$

where

$$\begin{aligned}
\tilde{\mathbf{x}}_1(k) &= [x_1(k+1)\, x_1(k+2) \cdots x_1(k+N)]^{\mathrm{T}} \\
\tilde{\mathbf{x}}_2(k) &= [x_2(k+1)\, x_2(k+2) \cdots x_2(k+N)]^{\mathrm{T}} \\
\tilde{\mathbf{u}}_{\mathrm{mb}}(k) &= [u_{\mathrm{mb}}(k)\, u_{\mathrm{mb}}(k+1) \cdots u_{\mathrm{mb}}(k+N-1)]^{\mathrm{T}} \\
\tilde{\mathbf{u}}_{\mathrm{hb}}(k) &= [u_{\mathrm{hb}}(k)\, u_{\mathrm{hb}}(k+1) \cdots u_{\mathrm{hb}}(k+N-1)]^{\mathrm{T}} \\
\tilde{\mathbf{u}}_{\mathrm{hs}}(k) &= [u_{\mathrm{hs}}(k)\, u_{\mathrm{hs}}(k+1) \cdots u_{\mathrm{hs}}(k+N-1)]^{\mathrm{T}}
\end{aligned}$$

with N the length of the prediction horizon in discrete time steps. The first part of the objective function consisting of the terms $J_{x_1}(\cdot)$ and $J_{x_2}(\cdot)$ describes the damage and flood risk of high water levels. This part of the objective function

is illustrated for $J_{x_1}(\cdot)$ in Figure 11. When the maximum of water level x_1 stays below a reference level r_{11}, there will be no damage at all. In this situation, the river is in its summer bed. Exceeding reference level r_{11} can lead to some damage (e.g. damaged houses, cattle and fields on the floodplains) and flood risk (e.g. risk of collapsing dikes), depending on the water level. Therefore, a water level exceeding reference level r_{11} is penalized with a quadratically increasing cost. The quadratic cost form represents that a small increase with relatively high water levels yields larger costs than the same small increase at a relatively low water level. The dike height r_{12} is the most important reference level. Exceeding reference level r_{12} will suddenly lead to huge (economic and social) costs caused by flooding of the crowded area of Rotterdam. Therefore, exceeding level r_{12} is penalized with a constant cost value, as well as with a quadratically increasing cost. In case it is impossible to prevent the water level exceeding the dike height, this quadratic cost ensures that the controller still minimizes the magnitude of the flood. The cost function that we propose is now defined on the cumulative exceedance of the critical water level by x_1 as follows:

$$J_{x_1}(\tilde{\mathbf{x}}_1(k)) = \alpha_{11} e_{\text{cum},11}(k) + \alpha_{12} e_{\text{cum},12}(k) + \alpha_{13}\tilde{e}_{13}(k)$$

where

$$e_{\text{cum},11}(k) = \sum_{j=1}^{N} (\max(x_1(k+j) - r_{11}, 0))^2$$

$$e_{\text{cum},12}(k) = \sum_{j=1}^{N} (\max(x_1(k+j) - r_{12}, 0))^2$$

$$\tilde{e}_{13}(k) = \begin{cases} 1 & \text{if } \max(\tilde{\mathbf{x}}_1(k)) > r_{12} \\ 0 & \text{otherwise} \end{cases}$$

where $e_{\text{cum},11}(k)$ and $e_{\text{cum},12}(k)$ are the cumulative exceedance for reference levels r_{11} and r_{12}, respectively. The parameters α_{11}, α_{12} and α_{13} are cost weights. The cost function $J_{x_2}(\tilde{\mathbf{x}}_2(k))$ is defined in a similar way.

The second part of the objective function (18) consisting of $J_{\text{mb}}(\tilde{\mathbf{u}}_{\text{mb}}(k))$, $J_{\text{hb}}(\tilde{\mathbf{u}}_{\text{hb}}(k))$ and $J_{\text{hs}}(\tilde{\mathbf{u}}_{\text{hs}}(k))$ describes the cost of closing and moving the storm surge barriers. Closure of the Maeslant barrier or the Hartel barrier blocks the navigation in the corresponding canals. Secondly, movement of a barrier also costs money due to wear and tear and energy costs. The cost function of the Maeslant barrier is therefore defined as:

$$J_{\text{mb}}(\tilde{\mathbf{u}}_{\text{mb}}(k)) = \alpha_{\text{mb1}} \sum_{j=1}^{N} [1 - u_{\text{mb}}(k+j-1)] + \alpha_{\text{mb2}} \sum_{j=1}^{N} |u_{\text{mb}}(k+j-1) - u_{\text{mb}}(k+j-2)| \quad (19)$$

with α_{mb1} the cost of closing the Maeslant barrier for one time step and α_{mb2} the cost of changing the state of the Maeslant barrier. The cost function of the Hartel barrier $J_{\text{hb}} = (\tilde{\mathbf{u}}_{\text{hb}}(k))$ is defined similarly. The cost function for the Haringvliet sluices $J_{\text{hs}} = (\tilde{\mathbf{u}}_{\text{hb}}(k))$ is also defined similarly, except for that only costs on the movements are considered (i.e. a term similar to the second term in (19)).

The Hollandsche IJssel barrier and its local control system are included in the TIO-MPC prediction model. Thus, the control actions for this barrier are not optimized. The control objective of keeping the water level in the Hollandsche IJssel below 2.25 mNAP results in trivial control actions: closing the Hollandsche IJssel barrier when the water level $x_1(k)$ at Rotterdam exceeds 2.25 mNAP. This is also the local control rule of the current control system of the Hollandsche IJssel barrier.

Prediction model

The complete nonlinear reservoir model is used as a prediction model for the TIO-MPC approach. The gate positions of the Haringvliet sluices are regular inputs of the TIO-MPC prediction model. The nonlinear relation between the

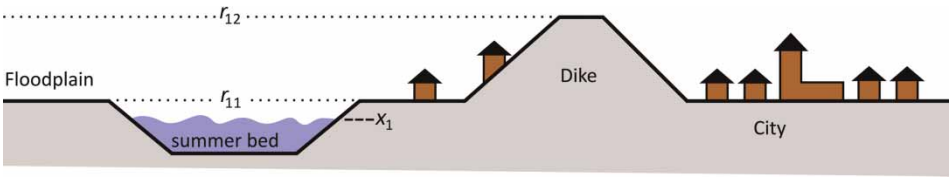

Figure 11 | A schematic view of a river and its surrounding area.

sluice gates, the water levels and the flow through these sluices fits inside the nonlinear optimization problem.

The TIO-MPC prediction model has both time instants and regular inputs. The state (open or closed) changes of the Maeslant barrier and the state changes of the Hartel barrier are both modeled with four time instants. These time instants represent the moments at which the state of the barriers change. The positions of the gates of the Haringvliet sluices are modeled with one regular input. The result is the following TIO-MPC prediction model:

$$\tilde{\mathbf{x}}(k) = f\big(\tilde{\mathbf{t}}(k), \tilde{\mathbf{u}}_{\mathrm{hs}}(k), \mathbf{x}(k)\big) \tag{20}$$

with

$$\begin{aligned}
\tilde{\mathbf{x}}(k) &= \left[\mathbf{x}^{\mathrm{T}}(k+1)\, \mathbf{x}^{\mathrm{T}}(k+2) \cdots \mathbf{x}^{\mathrm{T}}(k+N)\right]^{\mathrm{T}} \\
\tilde{\mathbf{u}}_{\mathrm{hs}}(k) &= \left[u_{\mathrm{hs}}(k)\, u_{\mathrm{hs}}(k+1) \cdots u_{\mathrm{hs}}(k+N-1)\right]^{\mathrm{T}} \\
\tilde{\mathbf{t}}(k) &= \left[t_{1,\mathrm{mb},k}\, t_{2,\mathrm{mb},k}\, t_{3,\mathrm{mb},k}\, t_{4,\mathrm{mb},k}\, t_{1,\mathrm{hb},k}\, t_{2,\mathrm{hb},k}\, t_{3,\mathrm{hb},k}\, t_{4,\mathrm{hb},k}\right]^{\mathrm{T}}
\end{aligned}$$

where $u_{\mathrm{hs}}(k)$ is the gate position (m) of the Haringvliet sluices at time step k. The time instants $t_{1,\mathrm{mb},k}$, $t_{2,\mathrm{mb},k}$, $t_{3,\mathrm{mb},k}$ and $t_{4,\mathrm{mb},k}$ (s) are the moments at which the Maeslant barrier changes its state. Similarly, the time instants $t_{1,\mathrm{hb},k}$, $t_{2,\mathrm{hb},k}$, $t_{3,\mathrm{hb},k}$ and $t_{4,\mathrm{hb},k}$ (s) are the moments at which the Hartel barrier changes its state. The inputs of the TIO-MPC model are illustrated in Figure 12.

The time instants are possibly beyond the length of the prediction horizon, which ensures that no discrete state changes take place over the prediction horizon. The discrete-time nonlinear reservoir model of the Rhine–Meuse delta requires regular input sequences for the state of the Maeslant barrier and the Hartel barrier. Therefore, a transformation is needed from the time instants into regular input sequences. This transformation is made as follows:

$$\tilde{\mathbf{u}}_{\mathrm{mb}}(k) = \left[u_{\mathrm{mb}}(k)\, u_{\mathrm{mb}}(k+1) \cdots u_{\mathrm{mb}}(k+N-1)\right]^{\mathrm{T}}$$

with

$$u_{\mathrm{mb}}(k+j) = \begin{cases} u_{\mathrm{mb}(k-1)} & \text{if } j \le \mathrm{round}\big(t_{1,\mathrm{mb},k}\big) \\ & \quad \text{or } \mathrm{round}\big(t_{2,\mathrm{mb},k}\big) \le j \le \mathrm{round}\big(t_{3,\mathrm{mb},k}\big) \\ & \quad \text{or } j \ge \mathrm{round}\big(t_{4,\mathrm{mb},k}\big) \\ 1 - u_{\mathrm{mb}}(k-1) & \text{otherwise} \end{cases}$$

for $j = 0, \ldots, N-1$, where $\tilde{\mathbf{u}}_{\mathrm{mb}}(k)$ is the regular input sequence created from the time instants.

Optimization

The TIO-MPC optimization problem consists of the model and the objective function that were discussed in the previous paragraphs. The model (20) relates the inputs of the water system to the evolution of the states (water levels) over the prediction horizon. The objective function (18) is a function of the inputs and the state evolution. These two relations together form a function that relates the inputs (degrees of freedom) to the value of the objective function:

$$J\big(\tilde{\mathbf{t}}(k), \tilde{\mathbf{u}}_{\mathrm{hs}}(k)\big) = f_{\mathrm{opt}}\big(\tilde{\mathbf{t}}(k), \tilde{\mathbf{u}}_{\mathrm{hs}}(k)\big) \tag{21}$$

In fact, the actual state $\mathbf{x}(k)$ of the system and the predicted disturbances $q_{1\mathrm{d}}(k+j)$, $q_{2\mathrm{d}}(k+j)$, $q_{3\mathrm{d}}(k+j)$,

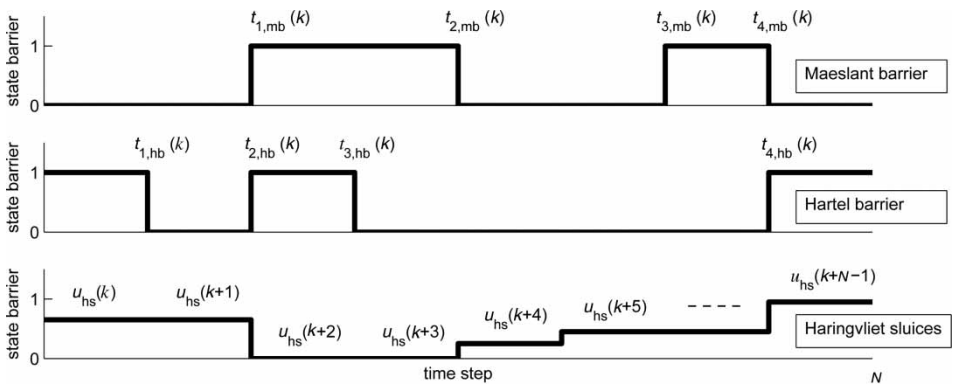

Figure 12 | Illustration of typical TIO-MPC inputs.

$h_{\mathrm{hs}}(k+j)$ and $h_{\mathrm{hs}}(k+j)$, for $j=0,\ldots,N-1$, are also inputs of this function. However, since these inputs are constant in the optimization problem, they are left out of (21). Function $f_{\mathrm{opt}}(\cdot)$ is the function to be minimized by the optimization algorithm, subject to the following constraints:

$$0 \le t_{1,\mathrm{mb}}(k) \tag{22}$$

$$t_{1,\mathrm{mb}}(k) - t_{\min} \le t_{2,\mathrm{mb}}(k) \tag{23}$$

$$t_{2,\mathrm{mb}}(k) - t_{\min} \le t_{3,\mathrm{mb}}(k) \tag{24}$$

$$t_{3,\mathrm{mb}}(k) - t_{\min} \le t_{4,\mathrm{mb}}(k) \tag{25}$$

$$t_{4,\mathrm{mb}}(k) \le t_{\max} \tag{26}$$

$$0 \le t_{1,\mathrm{hb}}(k) \tag{27}$$

$$t_{1,\mathrm{hb}}(k) - t_{\min} \le t_{2,\mathrm{hb}(k)} \tag{28}$$

$$t_{2,\mathrm{hb}}(k) - t_{\min} \le t_{3,\mathrm{hb}(k)} \tag{29}$$

$$t_{3,\mathrm{hb}}(k) - t_{\min} \le t_{4,\mathrm{hb}(k)} \tag{30}$$

$$t_{4,\mathrm{hb}(k)} \le t_{\max} \tag{31}$$

$$u_{\mathrm{hs,min}} \le u_{\mathrm{hs}}(k+j) \le \tilde{u}_{\mathrm{hs,max}}(k+j) \tag{32}$$

for $j=0,\ldots,N-1$, with

$$\tilde{u}_{\mathrm{hs,max}}(k+j) = \begin{cases} u_{\mathrm{hs,max}} & \text{if } x_3(k+j) \ge h_{\mathrm{hs}}(k+j) \\ u_{\mathrm{hs,min}} & \text{otherwise} \end{cases} \tag{33}$$

where $t_{\min}$(s) is the minimum time between two state changes; $t_{\max}$(s) is the maximum value of $t_{4,\mathrm{mb}(k)}$ and $t_{4,\mathrm{hb}(k)}$ and is larger than the prediction horizon, $u_{\mathrm{hs,min}}$ and $u_{\mathrm{hs,max}}$ (m) are, respectively, the minimum and maximum gate positions of the Haringvliet sluices and h_{hs} is the water level at the sea side of the Haringvliet sluices. The relation in (33) is the constraint of a one-directional flow through the Haringvliet sluices. The constraints (22)–(30) are constraints for the Maeslant barrier and the Hartel barrier, and describe the order of the time instants.

As mentioned, the cost function is minimized subject to the constraints using the nonlinear derivative-free optimization algorithm pattern search. The pattern search algorithm is started i times from i different initial points (i.e. multi-start optimization) until the end of the control step length.

Three different approaches can be used for determining appropriate initial points used for the multi-start pattern search optimization.

1. Initial solutions based on expert knowledge. The initial time instants of the Maeslant barrier and Hartel barrier are determined in the method shown in Figure 13. The time instants of the two highest peaks of h_{hvh} (predicted sea water level at Hoek van Holland) in the prediction horizon are determined. The initial time instants are based on these two peak times. For example, when the barriers are initially open, there are four obvious possibilities for each barrier: closing at peak 1, closing at peak 2, closing at peak 1 and peak 2, or staying open. These initial solutions are logical because the barriers are designed to block peak water levels. When the state of a barrier is not initially open, four different logical possibilities are determined. Combining these different possibilities of two barriers leads to $4^2=16$ initial solutions based on expert knowledge. The Haringvliet sluices are maximally open at every initial solution

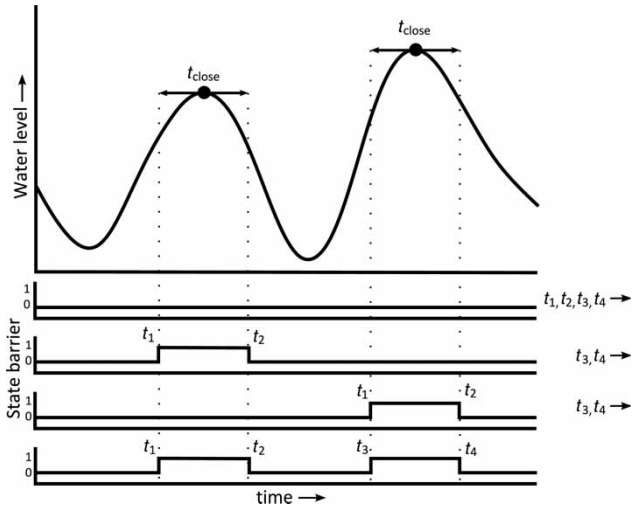

Figure 13 | This figure shows how the four initial points are determined based on expert knowledge, for a barrier that is initially open.

based on expert knowledge. This is normally optimal in critical conditions because the maximum amount of water is discharged out of the Rhine–Meuse delta.
2. Initial solutions that are generated randomly. The inclusion of a set of random initial solutions prevents the pattern search algorithm from converging to possibly suboptimal solutions based on expert knowledge only.
3. An initial solution that is based on the solution of the previous TIO-MPC optimization. The optimization problem of control step k is usually comparable to the optimization problem at control step $k-1$. Therefore, an initial solution based on the previous TIO-MPC optimization is usually relatively close to an optimum.

The TIO-MPC procedure at a time step is now the following:

1. A large set of initial solutions is created. The initial solution set has to be larger than the number of pattern search optimizations that can be performed in the control step length. This ensures that the multi-start pattern search optimization algorithm uses the complete control step length to search for good control actions.
2. The cost function values $f_{\text{opt}}(\cdot)$ of the initial solutions of step 1 are calculated.
3. The initial solutions are ranked based on the cost function values calculated in the previous step. An initial solution with a lower cost function value is expected to be more promising than an initial solution with a higher cost function value.
4. A pattern search optimization is started with the most promising initial solution based on the ranking calculated in step 3. After convergence of the pattern search optimization, a new optimization is started with the next most promising initial solution. This procedure is repeated until time runs out.
5. The best solution calculated in step 4 is selected as the output of the multi-start pattern search optimization.

RESULTS AND DISCUSSION

In order to illustrate the behavior of the proposed control approach, we consider simulation studies of the current and future situations of the Rhine–Meuse delta. The nonlinear reservoir models proposed in this paper are used both for simulations and predictions inside the MPC controller. Simulations are performed using the software package Matlab. Control actions are computed using the pattern search function as implemented in Mathwork's Genetic Algorithms and Direct Search Toolbox for Matlab (MathWorks 2007) in combination with multi-start. At each time step, the pattern search function solves the optimization problem (21)–(33).

The simulation time step is 2 min. The control time steps are set to 10 and 30 min, respectively, for the current control systems and the MPC system (10 min is the control step length used in the current control systems; 30 min is the control step length for the MPC system, providing a trade-off between fast control and lower quality versus slower control and higher quality). A prediction horizon of 24 hours is considered, as it is equal to the current practice (van Overloop 2009). It is assumed that the controllers have perfect predictions of the boundary conditions (the three river inflows and the two sea water levels) over the prediction horizon. The total simulation time span is 48 hours.

Several scenarios have been investigated in order to investigate the behavior of the proposed control system (van Ekeren 2010). These scenarios have been created using historical measurement data of 7–9 November 2007 provided by the Dutch national water body, Rijkswaterstaat (Rijkswaterstaat 2010). This period also includes the period in which the Maeslant barrier was closed due to storm conditions at sea. We consider the following two scenarios.

- Scenario 1 involves conditions due to a storm surge at sea and a sea level rise of 0.65 m. The flow of the river Rhine at Lobith (which gives an indication of the amount of water flowing into the Rhine Meuse delta) is 1,600 m^3/s. This results in a maximum sea water level of 3.81 mNAP with relatively low discharges of the three rivers.
- Scenario 2 involves a very extreme scenario: extremely high water level peaks of 4.96, 4.23 and 4.56 mNAP at Hoek van Holland are considered. The scenario has been created by combining historical data sets (of periods in February 1989 and November 2009) and adding effects due to a sea water level rise of 1.8 m.

Scenario 1

Figure 14 shows the results of the simulation using the current local control systems of the Rhine–Meuse delta. As can be observed, the Maeslant barrier and Hartel barrier are both closed for 20 hours. These long closures, in combination with the relatively low inflows of the rivers Lek, Waal and Meuse, keep the water levels at Rotterdam (y_1) and Dordrecht (y_2) very low. The area is therefore well protected against floods. However, the long closure is highly undesirable, since ocean vessels are blocked for more than 24 hours (navigation is not allowed for 4 hours before closure of the Maeslant barrier). The discharge volume through the Haringvliet sluices is quite low (a discharge volume of $15 \times 10^6\,\mathrm{m}^3$ over a time span of 48 hours), since it is related to the relatively low flow of the river Rhine at Lobith. The performance (computed as the evaluation of the cost function over the complete simulation) is 2.42×10^6.

Figure 15 shows the results of the simulation when using the TIO-MPC approach. We observe that instead of closing both the Maeslant and the Hartel barrier for a long period, the TIO-MPC approach only closes the Maeslant barrier for two short periods (3 hours in total). The maximum water levels at Rotterdam and Dordrecht are just above the first reference levels r_{11} and r_{12}, respectively. This illustrates the trade-off that the controller makes between exceeding the first reference levels (where damage starts) and input effort (cost on closing the barriers) that the TIO-MPC controller considers. The Haringvliet sluices are maximally open when possible (i.e. when constraint $y_3 > y_{\mathrm{hvh}}$ is not violated), resulting in a large discharge volume of $705 \times 10^6\,\mathrm{m}^3$ over the time span of 48 hours. The performance is 4.40×10^5.

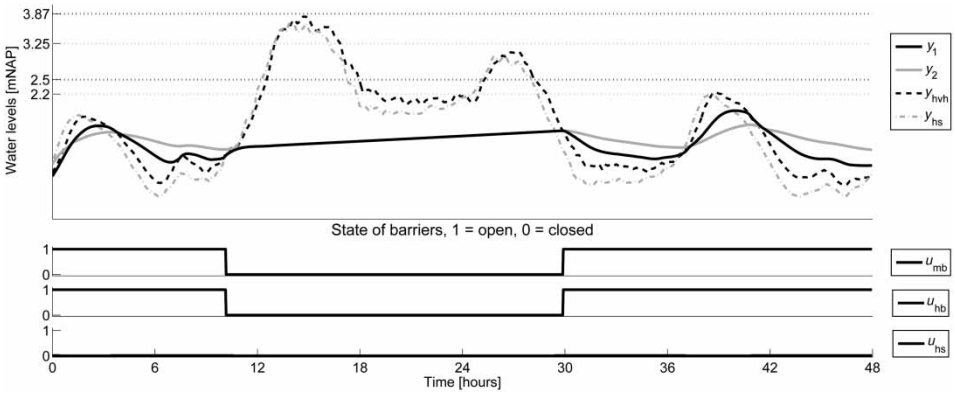

Figure 14 | Simulation results of the current control systems for scenario 1.

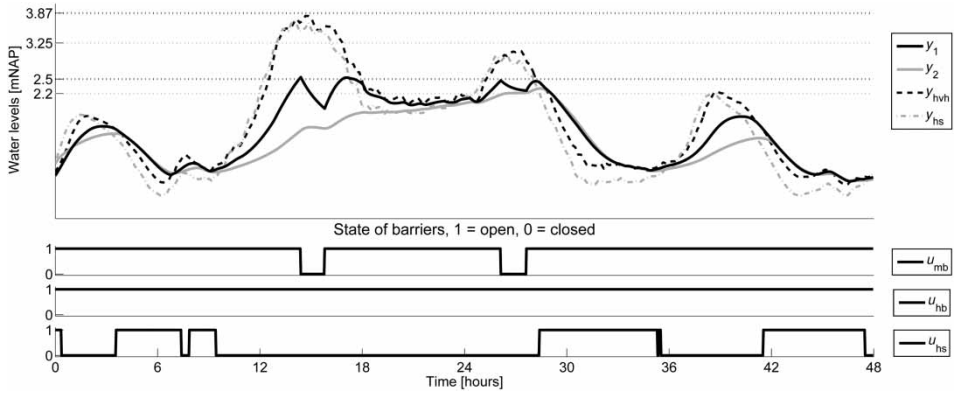

Figure 15 | Simulation results of the TIO-MPC approach for scenario 1. The dotted horizontal lines represent respectively from bottom to top the reference levels r_{21}, r_{11}, r_{22} and r_{12}.

Scenario 2

The simulation results for scenario 2 with the current control systems are shown in Figure 16. The simulation results for the TIO-MPC approach are shown in Figure 17. The simulation results for the current control systems show comparable results (i.e. state changes of the barriers occur mostly at approximately the same time instants). It is observed that the TIO-MPC approach, as well as the current control system, cannot prevent Rotterdam and Dordrecht being flooded. As can be seen in Figure 17, the peak water levels at Rotterdam and Dordrecht do not occur at the first and highest sea water level peak. After the second sea water level peak, the water levels y_1 and y_2 do not have enough time to decrease. Therefore, the high discharges of the rivers Rhine, Waal and Meuse result in extremely high water levels in the neighborhood of sea water level peak 3. Hence, in this case, the control approaches are not able to prevent the flooding. The performance when using the current control systems is 1.57×10^{10}. The performance when using the TIO-MPC approach is 1.53×10^{10}.

Since, with the current infrastructure, the controllers are not capable of preventing flooding, it is interesting to explore whether this will be possible in the case that additional infrastructure is built, in accordance with the plans of the Dutch government. Therefore, we next illustrate the behavior that emerges when two possible future situations are considered, which include possibilities for additional storage. First, an extension of the current system in which a small additional reservoir is present, and the Volkerak sluices are constructed to control this reservoir; and second, an extension in which the current system is extended with a large additional reservoir, again with the Volkerak sluices present for controlling this reservoir.

The simulation results of the TIO-MPC approach when using a future setup with a small additional reservoir are

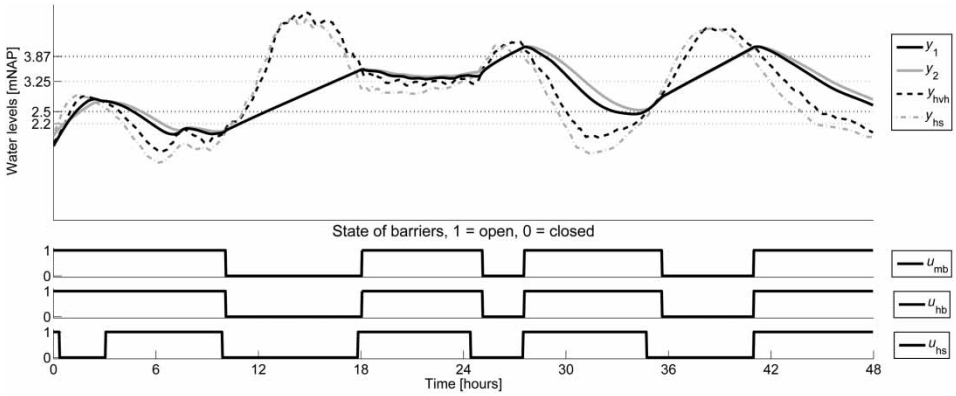

Figure 16 | Simulation results of the current control systems for scenario 2.

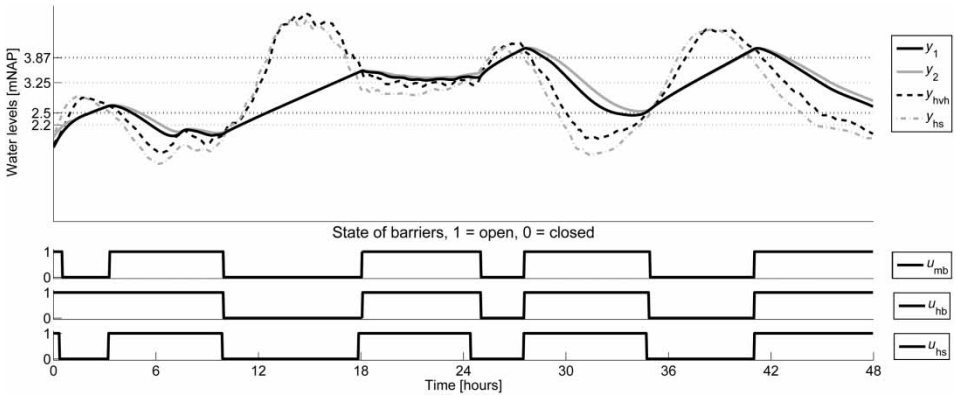

Figure 17 | Simulation results of the TIO-MPC approach for scenario 2.

shown in Figure 18. The water levels y_1 and y_2 are kept at 69 and 78 cm lower than the TIO-MPC approach of the current Rhine–Meuse delta setup. However, the water level at Dordrecht still becomes 9 cm higher than the dike height. Moreover, the critical water level r_{52} of Zeeland is exceeded by 50 cm. As can be seen in Figure 19, the TIO-MPC approach with a large reservoir has a significantly better performance. The peak water levels at Rotterdam, Dordrecht and Zeeland are respectively 111, 61 and 59 cm lower than the critical levels r_{12}, r_{22} and r_{52}. The performance of TIO-MPC with a small reservoir is 3.34×10^8. The performance of TIO-MPC with a large reservoir is 1.98×10^7.

CONCLUSIONS

In this paper, we have proposed an MPC approach for water systems represented as hybrid systems (i.e. combing both

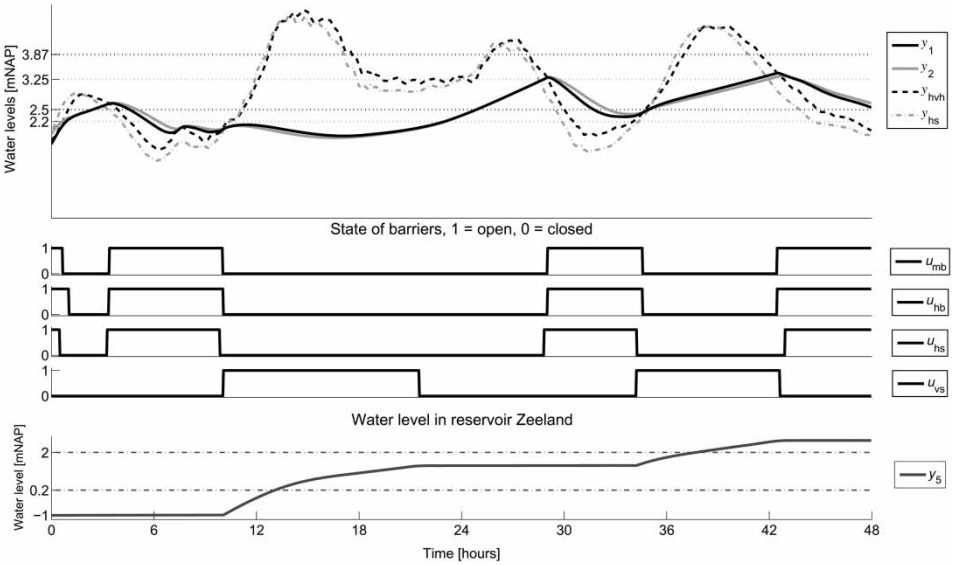

Figure 18 | Simulation results of the TIO-MPC approach, future setup with a small reservoir, for scenario 2.

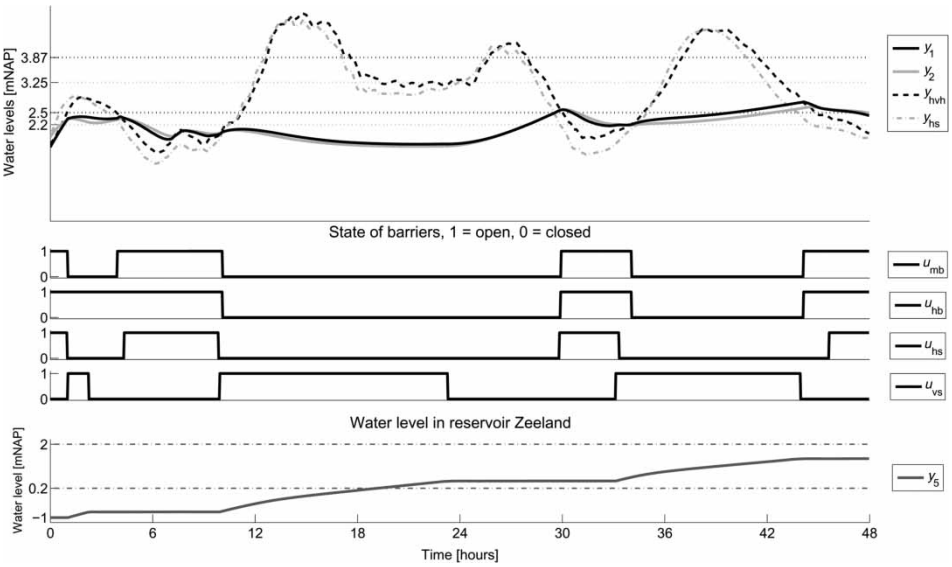

Figure 19 | Simulation results of the TIO-MPC approach, future setup with a large reservoir, for scenario 2.

continuous dynamics and discrete events). The approach proposed is based on so-called TIO. The idea of TIO-MPC is that the moments at which events should take place are determined, rather than that, for each time step, whether an event should take place or not is determined (as is typically the case in more conventional predictive control approaches for hybrid dynamical systems). When considering hybrid MPC problems, this approach can be promising in terms of reduced computational requirements. This is achieved by transforming the originally mixed-integer MPC optimization problem into an optimization problem with only real-valued variables. Using simulation studies based on the Rhine–Meuse delta in The Netherlands, the behavior of the approach has been illustrated, in particular, when a trade-off has to be made between input effort and damage costs. Current and future situations have been investigated.

Future research will be carried out on the application side and on the control and optimization theoretic side. Future research focuses on evaluating the performance of the proposed control scheme when there is a mismatch between the model used by the controller to make predictions and the dynamics of the water systems to be controlled (e.g. when the dynamics of the water system to be controlled are represented by a Sobek model). Moreover, future research will address the computational performance analysis of the proposed approach for future Rhine–Meuse delta setups and coordination among MPC controllers that control different, but interconnected, parts of large-scale water systems. From the control and optimization point of view, an extensive assessment and further comparison with alternative mixed-integer optimization methods from the literature (such as Bäck & Schütz (1995) and Li *et al.* (2006)) will be made. Moreover, a comparison will be made between the currently used approach of integrating the multiple objectives in one objective function and alternatives, such as epsilon constraint (Haimes 1973) and goal attainment (Gembicki & Haimes 1975).

ACKNOWLEDGEMENTS

This research is supported by the VENI project 'Intelligent multi-agent control for flexible coordination of transport hubs' (project 11210) of the Dutch Technology Foundation STW, a subdivision of The Netherlands Organisation for Scientific Research (NWO), and the EU Network of Excellence 'Highly-complex and networked control systems (HYCON2)'.

REFERENCES

Bäck, T. & Schütz, M. 1995 Evolution strategies for mixed-integer optimization of optical multilayer systems. In: *Evolutionary Programming: Proceedings of the Fourth Annual Conference on Evolutionary Programming* (J. R. McDonnell, R. G. Reynolds & D. B. Fogel, eds). MIT Press, Cambridge, Massachusetts, pp. 33–51.

Barjas Blanco, T., Williams, P., De Moor, B. & Berlamont, J. 2008 Flood prevention of the Demer using model predictive control. In: *Proceedings of the 17th IFAC World Congress*, Seoul, Korea, pp. 3629–3634.

Begovich, O., Ruiz, V. M., Besançon, G., Aldana, C. I. & Georges, D. 2007 Predictive control with constraints of a multi-pool irrigation canal prototype. *Latin American Applied Research* **37** (3), 177–185.

Bemporad, A. & Morari, M. 1999 Control of systems integrating logic, dynamics, and constraints. *Automatica* **35** (3), 407–427.

Brouwer, R. 2001 Operational Water Management. Technical report, Delft University of Technology, Delft, The Netherlands.

Camacho, E. F. & Bordons, C. 2004 *Model Predictive Control*. Springer-Verlag, New York.

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

Deltacommissie 2008 Samen werken met water. http://www.deltacommissie.com/advies. In Dutch. (Accessed on 1 November 2011.)

Farmani, R., Walters, G. & Savic, D. 2006 Evolutionary multi-objective optimization of the design and operation of water distribution network: total cost vs. reliability vs. water quality. *Journal of Hydroinformatics* **8** (3), 165–179.

Gembicki, F. & Haimes, Y. 1975 Approach to performance and sensitivity multiobjective optimization: the goal attainment method. *IEEE Transactions on Automatic Control* **20** (6), 769–771.

Gómez, M., Rodellar, J. & Mantecón, J. A. 2002 Predictive control method for decentralized operation of irrigation canals. *Applied Mathematical Modelling* **26** (11), 1039–1056.

Haimes, Y. 1973 Integrated system identification and optimization. *Control and Dynamic Systems: Advances in Theory and Applications* **10**, 435–518.

Intergovernmental Panel on Climate Change 2007 IPCC Fourth Assessment Report: Climate Change 2007 (AR4). Technical report.

Köppe, M. 2012 On the complexity of nonlinear mixed-integer optimization. *Mixed Integer Nonlinear Programming* **154**, 533–557.

Lewis, R. M., Torczon, V. & Trosset, M. W. 2000 Direct search methods: then and now. *Journal of Computational and Applied Mathematics* **124** (1–2), 191–207.

Li, R., Emmerich, M., Eggermont, J. & Bovenkamp, E. 2006 Mixed-integer optimization of coronary vessel image analysis using evolution strategies. In: *Proceedings of the 8th Annual Conference on Genetic and Evolutionary Computation*, New York, NY, pp. 1645–1652.

Malaterre, P. O. & Rodellar, J. 1997 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, pp. 239–248.

Malaterre, P. O., Rogers, D. C. & Schuurmans, J. 1998 Classification of canal control algorithms. *Journal of Irrigation and Drainage Engineering* **124** (1), 3–10.

MathWorks 2007 Genetic Algorithm and Direct Search Toolbox 2 – User's Guide.

Morari, M. & Baric, M. 2006 Recent developments in the control of constrained hybrid systems. *Computers and Chemical Engineering* **30** (10–12), 1619–1631.

Negenborn, R. R. 2007 Multi-Agent Model Predictive Control with Applications to Power Networks. PhD thesis, Delft University of Technology, Delft, The Netherlands.

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

Nixon, J. B., Dandy, G. C. & Simpson, A. R. 2001 A genetic algorithm for optimizing off-farm irrigation scheduling. *Journal of Hydroinformatics* **3** (1), 11–22.

Reed, P., Minsker, B. S. & Goldberg, D. E. 2001 A multiobjective approach to cost effective long-term groundwater monitoring using an elitist nondominated sorted genetic algorithm with historical data. *Journal of Hydroinformatics* **3** (2), 71–89.

Rijkswaterstaat 2009 Keringhuis. http://www.keringhuis.nl/. (Accessed on 1 November 2011).

Rijkswaterstaat 2010 Waterbase. http://live.waterbase.nl/. (Accessed on 1 November 2011).

Roeleveld, P. H. 2007 A New Control System for the Rhine–Meuse Delta. Master's thesis, Delft University of Technology, Delft, The Netherlands.

Ruiz, V. M. & Ramirez, L. 1998 Predictive control in irrigation canal operation. In: *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, San Diego, CA, pp. 3897–3901.

van Ekeren, H. 2010 Hybrid Model Predictive Control of the Rhine–Meuse Delta. Master's thesis, Delft University of Technology, Delft, The Netherlands.

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

van Overloop, P. J. 2009 Operational water management of the main waters in The Netherlands. Technical report, Delft University of Technology, Delft, The Netherlands.

van Overloop, P. J., Weijs, S. & Dijkstra, S. 2008 Multiple model predictive control on a drainage canal system. *Control Engineering Practice* **16** (5), 531–540.

Wahlin, B. T. & Clemmens, A. J. 2006 Automatic downstream water-level feedback control of branching canal networks: Theory. *Journal of Irrigation and Drainage Engineering* **132** (3), 198–207.

First received 15 November 2011; accepted in revised form 21 November 2012. Available online 4 January 2013