AGU PUBLICATIONS

Water Resources Research

RESEARCH ARTICLE
10.1002/2015WR017595

High-Performance Integrated Control of water quality and quantity in urban water reservoirs

S. Galelli[1], A. Castelletti[2,3], and A. Goedbloed[4]

[1]Pillar of Engineering Systems and Design, Singapore University of Technology and Design, Singapore, Singapore, [2]Department of Electronics, Information, and Bioengineering, Politecnico di Milano, Milan, Italy, [3]Institute of Environmental Engineering, ETH Zurich, Zurich, Switzerland, [4]Department of Civil and Environmental Engineering, National University of Singapore, Singapore, Singapore

Key Points:
- A novel computational framework to real-time operation of urban reservoirs
- The approach combines model reduction and model predictive control
- Water quality and drinking supply are optimized with an hourly frequency

Supporting Information:
- Supporting Information S1
- Movie S1

Correspondence to:
S. Galelli,
stefano_galelli@sutd.edu.sg

Citation:
Galelli, S., A. Castelletti, and A. Goedbloed (2015), High-Performance Integrated Control of water quality and quantity in urban water reservoirs, *Water Resour. Res.*, *51*, 9053–9072, doi:10.1002/2015WR017595.

Received 26 MAY 2015
Accepted 22 OCT 2015
Accepted article online 29 OCT 2015
Published online 19 NOV 2015

Abstract This paper contributes a novel High-Performance Integrated Control framework to support the real-time operation of urban water supply storages affected by water quality problems. We use a 3-D, high-fidelity simulation model to predict the main water quality dynamics and inform a real-time controller based on Model Predictive Control. The integration of the simulation model into the control scheme is performed by a model reduction process that identifies a low-order, dynamic emulator running 4 orders of magnitude faster. The model reduction, which relies on a semiautomatic procedural approach integrating time series clustering and variable selection algorithms, generates a compact and physically meaningful emulator that can be coupled with the controller. The framework is used to design the hourly operation of Marina Reservoir, a 3.2 Mm^3 storm-water-fed reservoir located in the center of Singapore, operated for drinking water supply and flood control. Because of its recent formation from a former estuary, the reservoir suffers from high salinity levels, whose behavior is modeled with Delft3D-FLOW. Results show that our control framework reduces the minimum salinity levels by nearly 40% and cuts the average annual deficit of drinking water supply by about 2 times the active storage of the reservoir (about 4% of the total annual demand).

1. Introduction

Large metropolitan areas, or "megapolitan regions," are undergoing a dramatic expansion driven by unprecedented rates of urbanization—globally, the number of people living in cities is expected to surpass 6 billion by 2050 [*United Nations Population Fund*, 2011]. This fast urban growth is putting increased demands on basic infrastructure services, presenting a major design challenge for planners and policy makers. In particular, the increase in drinking water demand, combined with climate-change-driven shift in the hydrologic cycle, could cause repeated water shortages unless adequate adaptation measures are taken. A study by *McDonald et al*. [2011] shows that about 150 million people are presently living in cities with perennial water shortage. This figure is expected to increase to almost 1 billion people by 2050 [*McDonald et al*., 2011].

In this context, the conventional urban water management approach—based on large-scale, centrally operated infrastructures for water supply, storm water drainage and wastewater disposal—is often criticized because it fails to make best use of the different components of the urban water cycle [*Marlow et al*., 2013]. These shortcomings have led to the development of the Sustainable Urban Water Management paradigm, which proposes the adoption of decentralized infrastructures, e.g., rainwater and storm water harvesting, as a means to augment water supplies within the urban environment [*Brown and Farrelly*, 2009]. Retention basins and urban reservoirs are increasingly used to support drinking water supply, since they make use of a resource, i.e., storm water, that would be otherwise wasted, thus limiting the amount of water abstracted from natural systems or produced with energy-intensive techniques, such as desalination. Yet, the operation of these infrastructures is problematic, for two main reasons. First, the presence of large impervious areas in urban catchments results in high discharge peaks and a fast runoff response to rainfall, with consequent short times of concentration [*Fletcher et al*., 2013]. Second, storm water transports a large amount of pollutants to the receiving water bodies [*Barbosa et al*., 2012].

A critical role to manage water quantity and quality in intensively developing catchments is played by the integration of real-time operation (control) techniques with high-fidelity simulation models that describe

the hydrological and biological processes in great details [*Beck*, 2005]. The main advantage of High-Performance Integrated Control (HPIC)—using Beck's terminology—is that it enhances engineering and ecological resilience by adapting the system's operation to rapidly propagating disturbances, such as rainfall events. Furthermore, HPIC does not require any structural engineering measures and thus retains the existing infrastructures [*Butler and Schütze*, 2005]. Yet such integration cannot be simply accomplished by combining individual models, but must account for the interaction between the different model components [*Bach et al*., 2014]. This aspect is particularly relevant with regard to retention basins and urban reservoirs, where the presence of spatially distributed and nonlinear processes requires the adoption of complex, 3-D hydroecological simulation models that cannot be directly combined with real-time operation models. In fact, the computational requirements of these 3-D models may prevent their uptake for optimization-based decision making, especially when short decision time steps (e.g., daily or hourly) must be adopted.

A viable strategy to handle this problem is *surrogate modeling* (or *computer model emulation*) [*Santner et al*., 2003], that is, the development and adoption of a simpler, cheaper-to-run model of the original high-fidelity simulation model. Model emulation has become very popular in several applications of water systems analysis, such as optimization and sensitivity analysis, with particular emphasis given to the response surface approach [*Mugunthan and Shoemaker*, 2006; *Castelletti et al*., 2010; *Yan and Minsker*, 2010]. This approach, also called metamodeling [*Blanning*, 1975], includes a broad variety of data-driven techniques, all aimed at finding a direct mapping between multiple explanatory variables (e.g., model parameters) and a model response variable (see the review by *Razavi et al*. [2012]). The response variable is commonly a statistics of model simulations (e.g., average value of a water quality parameter), although some response surface techniques that emulate an output time series have been recently proposed [*Wu et al*., 2014, 2015]. A key feature of the response surface approach is that it identifies an empirical, input-output relationship that does not explicitly emulate the main process dynamics and does not represent the internal components of the original simulation model. This last property is guaranteed by model reduction or reduced-order modeling, which aims at identifying a low-order, dynamic emulator of the original model [*Castelletti et al*., 2012a]. Most of the existing model reduction procedures are based on Proper Orthogonal Decomposition (POD), which uses data simulated with the original model—or "snapshots"—to construct a set of basis functions via singular value decomposition. The reduction is then achieved by retaining only those few basis functions that account for most of the variability in the data [*Willcox and Peraire*, 2002; *Antoulas*, 2005]. During the last decade, model reduction has been successfully applied to a variety of groundwater problems, including inverse modeling [*Vermeulen et al*., 2006; *Liu et al*., 2013; *Boyce and Yeh*, 2014; *Lin and McLaughlin*, 2014a], simulation [*Vermeulen et al*., 2004; *Siade et al*., 2010; *Cardoso and Durlofsky*, 2010], design of monitoring networks [*Ushijima and Yeh*, 2013], stochastic flow problems [*Pasetto et al*., 2011], and optimal management [*McPhee and Yeh*, 2008; *Lin and McLaughlin*, 2014b] (see the review by *Asher et al*. [2015]). More recently, model reduction has been employed for the optimal real-time operation of surface water resources systems, such as water flooding systems [*van Doren et al*., 2006] and polders [*Xu et al*., 2010, 2013].

Model reduction is particularly useful for meeting water quantity and quality targets in urban reservoirs for two reasons. First, it reduces the computational requirements of 3-D hydroecological models, thus making their integration into real-time operation frameworks effective, even when very short decision time steps are adopted—see, for example, *Joseph-Duran et al*. [2014], where an emulator of a sewer network is used to solve an optimal control problem with a 5 minute time step. Second, model reduction yields a low-order, dynamic model emulating only the internal components of the original model that are relevant to the decisions that must be optimized. In other words, the dynamic emulator issues at each time step an estimate of the current state of the system (e.g., water temperature and salinity) upon which the decisions (e.g., pumping and discharging) are made.

Following this research stream, we propose a novel HPIC framework to support the operation of urban water reservoirs by combining model reduction of computationally intensive 3-D models and real-time operation techniques. The model reduction relies on the semiautomatic procedural approach introduced by *Castelletti et al*. [2012b], named Dynamic Emulation Modeling (DEMo). Unlike traditional POD-based procedures, which generally project the entire state vector of the original model onto a reduced-space basis, the DEMo procedure reduces the dimensionality of the original simulation model with an alternative, "output-oriented" approach. First, snapshots are used to identify spatial clusters in which a specific state variable assumes a similar pattern over time. Then, only the clusters relevant to the output of interest are

selected—as reduced state variables—with the aid of a variable selection algorithm. This output-oriented approach ensures adaptability to nonlinear processes, high efficiency in terms of complexity reduction, physical interpretability of the emulator structure, and compatibility with real-time control. The emulator is used to inform a real-time Model Predictive Control (MPC) scheme that exploits a short-term prediction of the reservoir inflow to optimize the control decisions over the prediction horizon [*Camacho and Alba*, 2013]. The HPIC framework is applied to the real-time operation of Marina Reservoir—a storm-water-fed reservoir recently constructed in the central business area of Singapore—which is occasionally affected by flood events and high salinity levels resulting from the recent formation of the impoundment from a former estuary.

2. Study Site

2.1. Water Management in Singapore

Singapore is a city-state with an area of 700 km^2, limited natural water resources, and a population of about 5.5 million people, which makes it one of the most densely populated regions in South-East Asia. To ensure a stable and sustainable water supply, the government has recently adopted the *Four National Taps Strategy*, in which water is supplied from four different sources: treated storm water harvested from local catchments, imported water from Malaysia, desalinated water, and reclaimed water (NEWater) [*Tortajada*, 2006]. The core of the first tap is the use of Singapore's largest catchment, Marina Reservoir catchment, as a source of water supply. The reservoir—created in late 2008 with the construction of a 350 m-wide barrage that closed the former marina from the sea (Figure 1, top)—meets about 10% of the current water demand [*Kristiana et al*., 2011].

2.2. Marina Reservoir

Marina Reservoir has a surface area of 2.45 km^2 and an active storage of 3.2 Mm^3. It is fed by five main tributaries draining a catchment of approximately 100 km^2 that produces a mean annual inflow of 180 Mm^3. The catchment is drained by concrete lined canals, which make the time of concentration extremely short and the base flow almost null (about 1 h and 2.9 m^3/s, respectively). Because of the high-intensity rainfall events characterizing the region, peak discharges up to 750 m^3/s can occur. This combination of tropical climate, short time of concentration, and small active storage makes the reservoir storage dynamics very sensitive to peak discharges. The reservoir is also operated to prevent flooding of the surrounding lowland areas. To meet these operating objectives, the barrage is equipped with surface gates, bottom pipes, and a pumping station—all operated with an hourly decision time step accordingly to the time of concentration of the catchment and the semidiurnal tidal frequency. Gates and pipes discharge water into the sea during low-tide events, while pumps are operated during high tides, when the sea water level is higher than in the reservoir (Figure 1, bottom). With an installed capacity of 280 m^3/s, the pumping station is very energy intensive and its use ought to be limited to exceptional cases. The energy cost is considered as the third operation target; activating the pumps should be kept at a minimum [*Galelli et al*., 2014a].

The preservation of water quality standards in Marina Reservoir is challenged by the considerable load of nutrients (mostly phosphates and nitrates), which, combined with high water temperature and light intensity, often leads to in-reservoir eutrophic conditions [*Antenucci et al*., 2013]. Besides, the recent formation of the impoundment from a former estuary makes high salt concentrations another important issue. Since the permanent closure of the barrage in April 2009, salinity levels have dropped to typical fresh water standards in many areas of the reservoir [*Antenucci et al*., 2013]. However, residual sources of saline water persist in the form of groundwater seepage driven by the water level gradient between sea and reservoir during high tide events. The accumulation of salt has resulted in a thin saline layer on the reservoir bed—especially near the barrage—of about 1–2 m thickness, equivalent to 10% of the reservoir volume, where the salinity is approximately 30 parts per thousand (ppt) [*Antenucci et al*., 2013]. During dry periods, salinity levels increase in the upper layers through diffusion and mixing, resulting in high salinity at the drinking water intake point, with potential consequences on water supply and treatment costs. (The drinking water supply is stopped when the salinity at the drinking water intake is higher than 0.25 ppt). The seawater intrusion process could be controlled, in the first instance, by resorting to structural engineering measures. For example, the injection of freshwater along the barrage area would create a number of hydraulic barriers and prevent seawater from infiltrating into the reservoir. This solution, which is often adopted to protect coastal

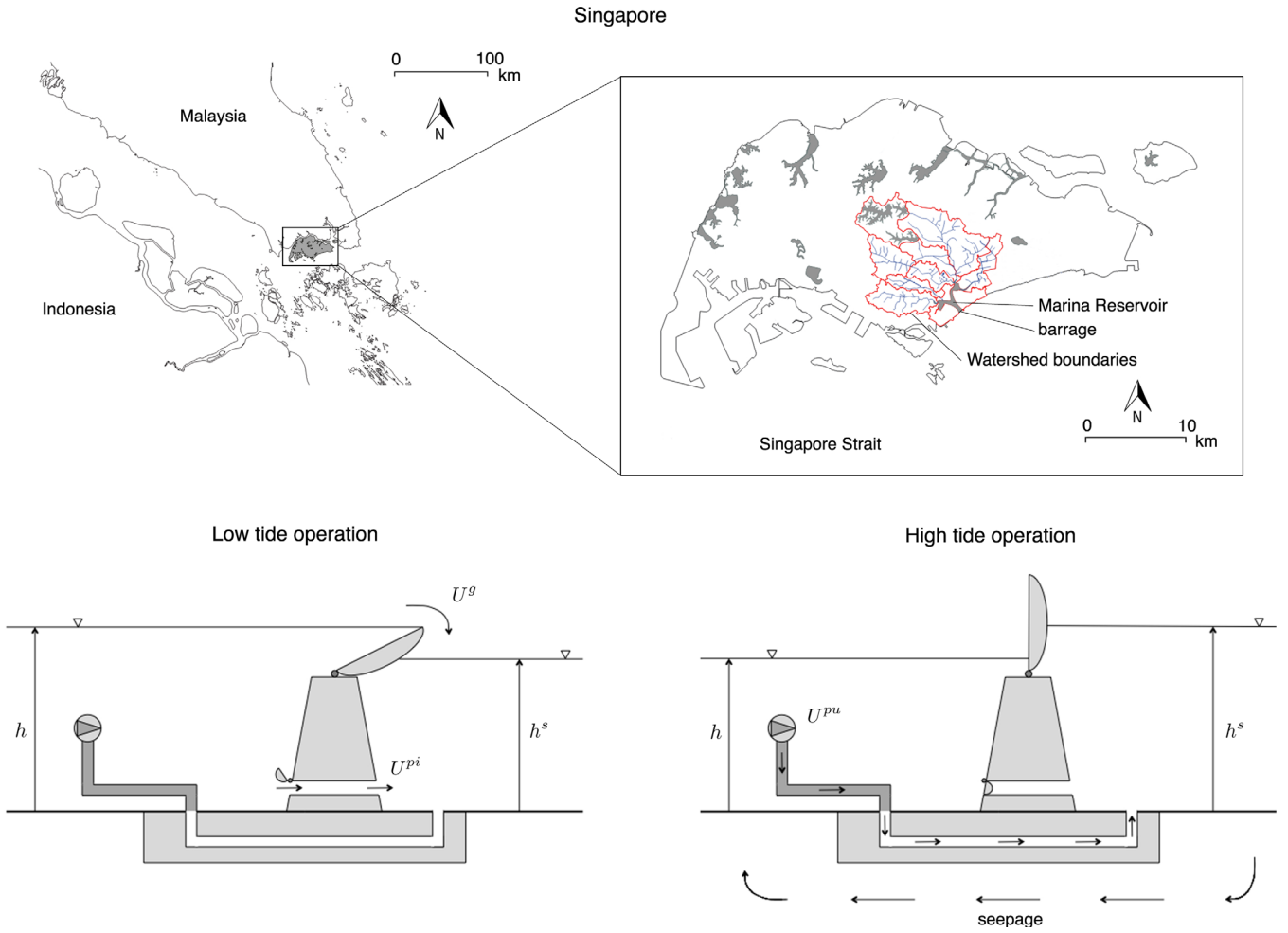

Figure 1. (top) Marina Reservoir location in Singapore; (bottom) schematization of Marina barrage operation during (left) low and (right) high tide conditions with the decision variables adopted in this study.

aquifers [*Bray and Yeh*, 2008], could be complemented by a system designed to continuously pump seawater from the reservoir bottom. A more invasive measure would be the deployment of physical subsurface barriers, such as concrete grout, bituminous substances, or sheet piles, which retain groundwater and inhibit seawater intrusion [*Luyun et al*., 2011]. However, these are invariably costly options that would require a physical modification of the barrage [*Werner et al*., 2013]. In this work, we study an economically viable, "soft-path" alternative to infrastructural interventions, namely the mechanical control of the salinity levels obtained by operating the bottom pipes during low-tide events. This requires redesigning the real-time operation of the barrage to include the salinity control target.

3. Models, Methods, and Tools

3.1. Models Available

Because of its strategic role and peculiar location, Marina Reservoir has been studied intensively in the last few years. Various models have been developed to describe both the hydrodynamics of the whole reservoir (Delft3D-FLOW) [*Zijl and Twigt*, 2007] and the associated ecological and water quality processes (ELCOM-CAEDYM) [*Antenucci et al*., 2013]. Comparatively, fewer studies have focused on the operational aspects. The real-time operation model developed by *Galelli et al*. [2014a] and Delft3D-FLOW are adopted in this work, as they are being considered for adoption by the barrage operators and were tested in a recent research project [*Galelli et al*., 2012].

3.1.1. Real-Time Operation Model

To simulate the reservoir operation, a model that calculates the release from gates, pumps, bottom pipes, and drinking water intake is needed. Since the storage dynamics is sensitive to rainfall-induced peak discharges, the reservoir operation benefits from a proactive decision-making strategy based on real-time hydrometeorological predictions. This is the rationale behind the adoption of a deterministic MPC scheme, which relies on a short-term prediction of the reservoir inflow and the optimization of the release decisions over the inflow prediction horizon [*Castelletti et al.*, 2008].

The MPC scheme is based on the *receding horizon principle*. At each decision time step t, a hydrological model issues a prediction of the reservoir inflow—over the horizon $[t, t+h]$—that is used by an optimization algorithm to design the release sequence. Only the release decision $\mathbf{U}_t$ for the first time step $[t, t+1)$ is implemented, and, at time $t+1$, a new problem is formulated and solved over the horizon $[t+1, t+h+1]$ with a new prediction of the inflow updated to the information available at time $t+1$ [*Castelletti et al.*, 2008]. When accounting for water quantity objectives (i.e., drinking water supply, flood control, and energy cost), the MPC scheme is coupled with the following lumped model of the reservoir mass balance that describes the storage dynamics and allows computing the operating objectives:

$$s_{t+1} = s_t + \left[a_{t+1} - R(s_t, \mathbf{U_t}, a_{t+1}, h^s_{t+1})\right] \cdot \Delta \tag{1}$$

where s_t is the storage (m^3) at time t; a_{t+1} the inflow (m^3/s) in the interval $[t, t+1)$; $\mathbf{U}_t$ the decision vector comprising the release decision from gates (U^g_t), pumps (U^{pu}_t), bottom pipes (U^{pi}_t), and drinking water intake (U^d_t) (m^3/s); h^s_{t+1} the sea level (m) at time $t+1$; Δ the integration time step (equal to 3600 s). $R(\cdot)$ is the release (m^3/s) that actually occurs in the interval $[t, t+1)$ due to physical and operational constraints [*Soncini-Sessa et al.*, 2007].

The presence of different water quantity targets is formalized through the definition of immediate cost functions. The *drinking water supply* cost function $g^w_t(\cdot)$ is a function of the supply deficit with respect to the drinking water pumping capacity w (equal to 4.75 m^3/s), and it is defined as

$$g^w_t(s_t, U^w_t, a_{t+1}) = \begin{cases} 0 & \text{if } r^w_{t+1} \geq w \\ (w - r^w_{t+1}) & \text{otherwise} \end{cases} \tag{2a}$$

where r^w_{t+1} is the actual volume pumped in the interval $[t, t+1)$ to the upstream water treatment facility.

The *flood control* cost function $g^f_t(\cdot)$ expresses the deviation from the safety water level range $[-0.2, 0.3]$ m. It is expressed as

$$g^f_t(s_t, \mathbf{U}_t, a_{t+1}, h^s_{t+1}) = \begin{cases} (h_t + 0.2)^2 & \text{if } h_t < -0.2 \\ (h_t - 0.3)^2 & \text{if } h_t > +0.3 \\ 0 & \text{otherwise} \end{cases} \tag{2b}$$

where h_t is the reservoir level (m), computed as the ratio between reservoir storage s_t and surface S.

The *energy cost* function $g^p_t(\cdot)$ accounts for the pumps usage during high tide events, and it is defined as

$$g^p_t(s_t, U^p_t, a_{t+1}, h^s_{t+1}) = (r^p_{t+1})^2 \tag{2c}$$

where r^p_{t+1} is the actual volume pumped in the interval $[t, t+1)$ from the barrage to the open sea.

According to the multiobjective nature of the problem, the three immediate cost functions $g^w_t(\cdot)$, $g^f_t(\cdot)$, and $g^p_t(\cdot)$ are aggregated to provide the cost function $g_t(\cdot)$:

$$g_t(\cdot) = \lambda^w \cdot g^w_t(\cdot) + \lambda^f \cdot g^f_t(\cdot) + \lambda^p \cdot g^p_t(\cdot) \tag{3}$$

with $\lambda^w + \lambda^f + \lambda^p = 1$. The value of the weights λ^w, λ^f, and λ^p can be varied—in order to study the trade-off between the different operating objectives—or kept fixed, as in the current study (see next).

The real-time optimization problem thus takes the following form:

$$\min_{\mathbf{U}_t, \ldots, \mathbf{U}_{t+h-1}} \left[\sum_{\tau=t}^{t+h-1} g_\tau(\cdot) + \bar{g}_{t+h}(s_{t+h}) \right] \tag{4a}$$

subject to

$$s_{\tau+1} = f(s_\tau, \mathbf{U}_\tau, a_{\tau+1}, h^s_{\tau+1}) \qquad \tau = t, \ldots t+h-1 \tag{4b}$$

$$\mathbf{U}_\tau \in \mathcal{U}_\tau \qquad \tau = t, \ldots t+h-1 \tag{4c}$$

$$\mathbf{a}^{t+h}_{t+1} \text{ given scenario} \qquad \tau = t, \ldots t+h-1 \tag{4d}$$

$$\mathbf{h}^{s,t+h}_{t+1} \text{ given scenario} \qquad \tau = t, \ldots t+h-1 \tag{4e}$$

$$s_t \text{ given} \tag{4f}$$

where s_τ is the storage at time τ, whose dynamics is described by the mass conservation equation (equation (1)); $\mathbf{U}_t$ the vector of release decisions belonging to the set of the feasible decisions $\mathcal{U}_\tau$; $g_\tau(\cdot)$ the immediate cost function expressing the cost associated with the system's transition; and $\bar{g}_{t+h}(s_{t+h})$ the penalty function associated with the final storage s_{t+h}. The vector of inflow and sea level predictions, $\mathbf{a}_{t+1}$ and $\mathbf{h}^s_{t+1}$, is issued with an hourly resolution by a M5 model tree and a dynamic tidal model over a horizon h equal to 3 h [*Galelli et al.*, 2014a]. The release decisions are optimized with the trust region method based on interior point techniques [*Waltz et al.*, 2006]. This operation model is inherently parsimonious, so the problem of optimizing the release decisions is solved in real time in a computationally efficient way.

Accounting for seawater intrusion requires an additional model—such as Delft3D-FLOW—mapping the release decisions and hydrometeorological boundaries into the saline layer dynamics. To control salt intrusion, an additional cost function is added to the problem formulation, i.e.,

$$g^s_t(\cdot) = Y_t \tag{5}$$

where Y_t is the output simulated by Delft3D-FLOW (see next section). In this case, the aggregated cost function $g_t(\cdot)$ is given by

$$g_t(\cdot) = \lambda^w \cdot g^w_t(\cdot) + \lambda^f \cdot g^f_t(\cdot) + \lambda^p \cdot g^p_t(\cdot) + \lambda^s \cdot g^s_t(\cdot) \tag{6}$$

with $\lambda^w + \lambda^f + \lambda^p + \lambda^s = 1$. The real-time optimization problem (equation (4a)) is then modified to (1) account for the different aggregated cost function $g_t(\cdot)$ (equation (6)) and (2) adding a process model describing the salinity dynamics.

3.1.2. 3-D Hydrodynamic Model

Delft3D-FLOW simulates three-dimensional unsteady flow and transport phenomena in response to the releases from the barrage and hydrometeorological boundaries—including the effect of density differences due to nonuniform temperature and salinity distribution. The numerical modeling system solves the Navier-Stokes equations for an incompressible fluid, under the shallow water and Boussinesq assumptions. The system of equations consists of the horizontal equation of motion, continuity equation, and transport equations for conservative constituents. These partial differential equations are solved on a finite difference grid (see *Deltares* [2014] for further details).

The horizontal spatial resolution varies throughout the domain, with an average grid size of 25 by 25 m. The model has a maximum of 12 vertical layers, each with a specified height, so that the amount of layers depends on the local depth. The average vertical resolution is about 0.5 m. This results in a smooth representation of the bottom topography and accurate calculation of stratification. Overall, the model has a total of about 2500 computational cells. Delft3D-FLOW has five state variables for each cell, namely horizontal and vertical components of velocity, water temperature, and salinity (Table 1). Their value is saved on a coarser grid of 111 nodes and along a south-north cross section, represented in Figure 2 along with the reservoir bathymetry. Considering a total of 111 nodes and an average of 10 vertical layers, the state vector $\mathbf{X}_t$ has ~5500 variables. The input to Delft3D-FLOW consists of two vectors $\mathbf{U}_t$ and $\mathbf{W}_t$ of release decisions and hydrometeorological boundaries (see Tables 2 and 3). The output Y_t is the salinity at the control point, namely the deepest layer (depth of 6.5 m) of a node located a few hundred meters from the barrage (see Figure 2). The model has a temporal resolution of 30 s and an associated real-to-run time ratio of about 100:1 (estimate obtained with an Intel Xeon 2.53 GHz Eight-Core with 24 GB RAM); for instance, it takes 1 computing day to simulate 100 real days.

The model was manually calibrated with a trial-and-error procedure aimed at adequately reproducing the temporal and spatial dynamics of flow and transport phenomena. The groundwater seepage W^{gs}_t is

Table 1. Components of Delft3D-FLOW State Vector $\mathbf{X}_t^i$ for the *i*th Node

Variable	Unit of Measurement	Description
$X_t^{i,hs1}$	m/s	Horizontal speed (first component)
$X_t^{i,hs2}$	m/s	Horizontal speed (second component)
$X_t^{i,vs}$	m/s	Vertical speed
$X_t^{i,wt}$	°C	Water temperature
$X_t^{i,s}$	ppt	Salinity

described as an empirical, linear relation calibrated on a set of salinity measurements. With this relation, the intrusion rate is described as a function of the difference between sea and reservoir water level.

3.2. High-Performance Integrated Control Framework

With a real-to-run time ratio of 100:1, Delft3D-FLOW cannot be practically interlaced to the real-time operation model, which requires thousands of simulations to explore the decision space and determine the optimal control. Consider, for example, the problem of simulating 1000 alternative operation strategies over a horizon of 3 h. This would require a computing time of about 30 h, which is more than 1 order of magnitude larger than the decision time step adopted by the barrage operators. To solve this problem, the HPIC framework combines a dynamic low-order emulator and a real-time operation model to design the hourly operation of the barrage (Figure 3).

First, historical observations of the hydrometeorological boundaries $\mathbf{W}_t$ and stochastically generated release decisions $\mathbf{U}_t$ are used to run Delft3D-FLOW (Design of Experiments in the figure), generating a data set $\mathcal{F}$ of observations $\{\mathbf{X}_t, \mathbf{W}_t, \mathbf{U}_t, Y_t, \mathbf{X}_{t+1}\}$. This high-dimensional data set is then employed by model reduction algorithms to identify a low-order, computationally efficient emulator of the saline layer dynamics. Second, the emulator is embedded within the operation model to design in real-time release decisions that balance water quantity and quality objectives. The effect of these decisions is quantitatively assessed via high-fidelity simulation with Delft3D-FLOW, whose high spatial and temporal resolution allows studying the dynamics of the saline layer throughout the reservoir domain.

3.3. Dynamic Emulation Modeling

The purpose of DEMo is to identify a low-order, computationally efficient emulator on the data set $\mathcal{F}$ generated via simulation with Delft3D-FLOW. The emulator must be such that its output y_t reproduces Delft3D-FLOW output

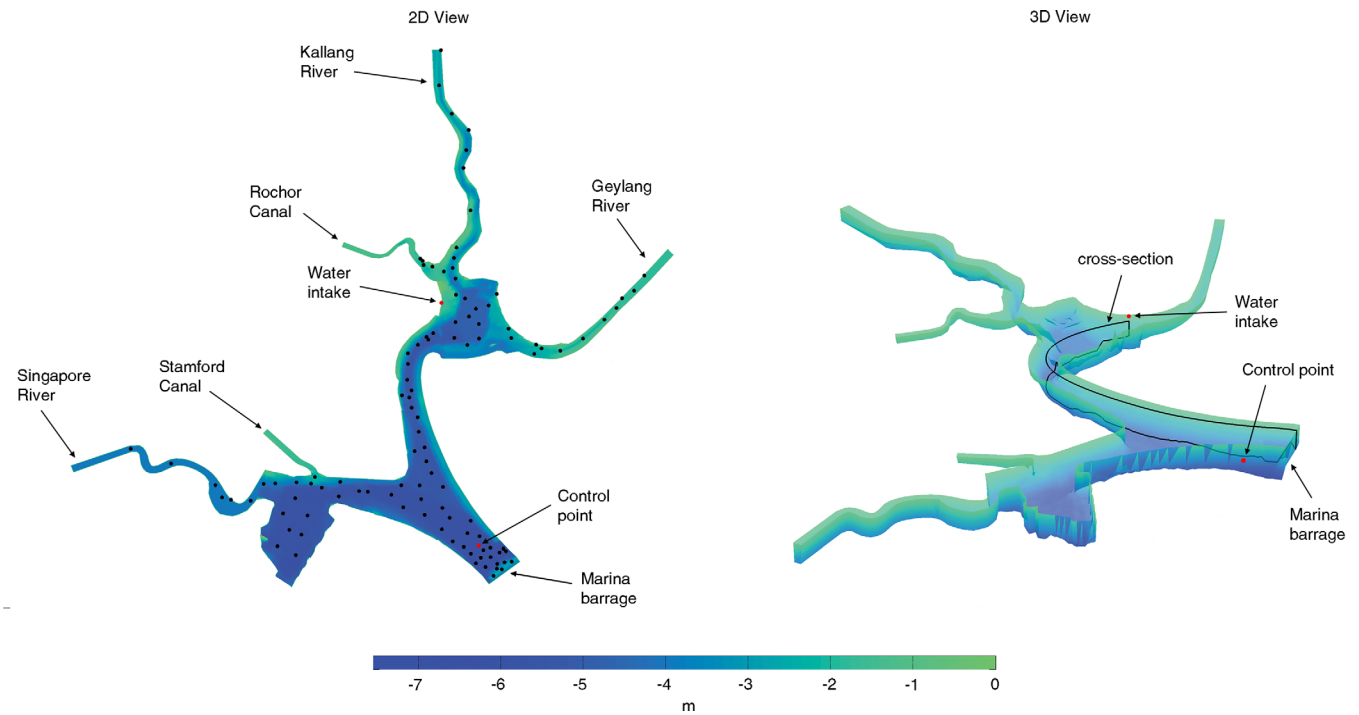

Figure 2. (left) 2-D view of Marina Reservoir bathymetry, with the black dots representing the grid adopted to save the output of Delft3D-FLOW; (right) 3-D view of the bathymetry, with the black line representing the south-north cross section used to analyze the model results. In both parts, the red dots correspond to the water intake and control point.

Table 2. Components of Delft3D-FLOW Release Decisions Vector $\mathbf{U}_t$

Variable	Unit of Measurement	Description
U_t^g	m^3/s	Release from gates
U_t^{pu}	m^3/s	Release from pumps
U_t^{pi}	m^3/s	Release from bottom pipes
U_t^w	m^3/s	Release from drinking water pump

Y_t, but has lower dimensional state, hydrometeorological boundaries and release decision vectors $\mathbf{x}_t$, $\mathbf{w}_t$, and $\mathbf{u}_t$. The emulator takes the following state-space form:

$$\mathbf{x}_{t+1} = \mathbf{f}(\mathbf{x}_t, \mathbf{w}_t, \mathbf{u}_t) \quad (7a)$$

$$y_t = h(\mathbf{x}_t, \mathbf{w}_t, \mathbf{u}_t) \quad (7b)$$

where $\mathbf{f}(\cdot)$ is a nonlinear function that models the state transitions and $h(\cdot)$ the output transformation function. Since the emulator is used to control seawater intrusion in real time, the output Y_t we want to reproduce is the immediate cost function $g_t^s(\cdot)$ associated with the water quality objective (equation (5)). The adoption of a state-space formulation is particularly useful for the HPIC framework, as the resolution of the control problem requires to know the system state $\mathbf{x}_t$—the release decision is based on the value of the state variables at time t. The generation of a high-dimensional data set with Delft3D-FLOW and model reduction are the two components of the DEMo procedure [*Castelletti et al.*, 2012b].

3.3.1. Design of Experiments

The DoE consists of a sampling in the space of the 3-D hydrodynamic model inputs to design a sequence of trajectories for the release decisions $\mathbf{U}_t$ and hydrometeorological boundaries $\mathbf{W}_t$. The model is run on these management and hydrometeorological scenarios to generate the data set $\mathcal{F}$. This data set must reproduce all dynamic behaviors of the model, excited and forced by the spectrum of inputs. This can be done by adopting techniques for the *design of dynamic experiments* [*Forssell and Ljung*, 2000; *Young*, 2011], which activate the dynamical modes that are of interest to the DEMo problem. However, if the computational requirements of the model limit the feasible number of simulation runs, a less formal DoE may need to be adopted. For example, historical observations available for the hydrometeorological boundaries could be used for $\mathbf{W}_t$, while the release decisions $\mathbf{U}_t$ could be generated with pseudo-random number generators [*MacWilliams and Sloane*, 1976]. The latter approach is adopted in this study (see section 4 for further details).

3.3.2. Model Reduction

A common approach to the model reduction problem is to project the entire state vector $\mathbf{X}_t$—along with the inputs $\mathbf{W}_t$ and $\mathbf{U}_t$—onto a lower dimensional subspace, where the model equations are solved. In practice, samples of the state and input variables generated with the DoE are used by an interpolation scheme that finds a subspace of the state vector space, where the full model solution resides. This subspace is evaluated so that it captures the majority of the variation in the model solution [*Siade et al.*, 2010]. A limit of this approach is that it conditions the emulator on a projection of the entire state vector, while only a few state variables might be relevant with respect to the output of interest. In water quality models, for example, different processes might be well approximated with a small subset of the variables simulated by the original model, thus adding to the emulator's simplicity and physical interpretability. This is the rationale for adopting a three-step model reduction approach based on variable aggregation, selection, and model calibration [*Castelletti et al.*, 2012b].

3.3.2.1. Variable Aggregation

The purpose of variable aggregation is to transform the state vector $\mathbf{X}_t$ into a lower dimensional vector $\tilde{\mathbf{X}}_t$. The transformation is obtained by processing the data in $\mathcal{F}$ with a suitable aggregation scheme, so that the majority of the variation in $\mathbf{X}_t$ is captured. This yields a lower dimensional data set $\tilde{\mathcal{F}}$ equal to $\{\tilde{\mathbf{X}}_t, \mathbf{W}_t, \mathbf{U}_t, Y_t, \tilde{\mathbf{X}}_{t+1}\}$. The aggregation scheme can rely on fully automatic techniques, such as locally linear embedding [*Roweis and Saul*, 2000], self-organizing maps [*Kalteh et al.*, 2008], or time series clustering [*Liao*, 2005], which is adopted for the first time in this study. An interesting feature of clustering is that it organizes the state variables into

Table 3. Components of Delft3D-FLOW Hydrometeorological Boundaries Vector $\mathbf{W}_t$

Variable	Unit of Measurement	Description
a_t	m^3/s	Inflow
W_t^{gs}	m^3/s	Groundwater seepage
W_t^{cc}	%	Fractional cloud cover
W_t^{rh}	%	Relative humidity
W_t^{at}	°C	Air temperature
W_t^{ws}	m/s	Wind speed
W_t^{wd}	°	Wind direction

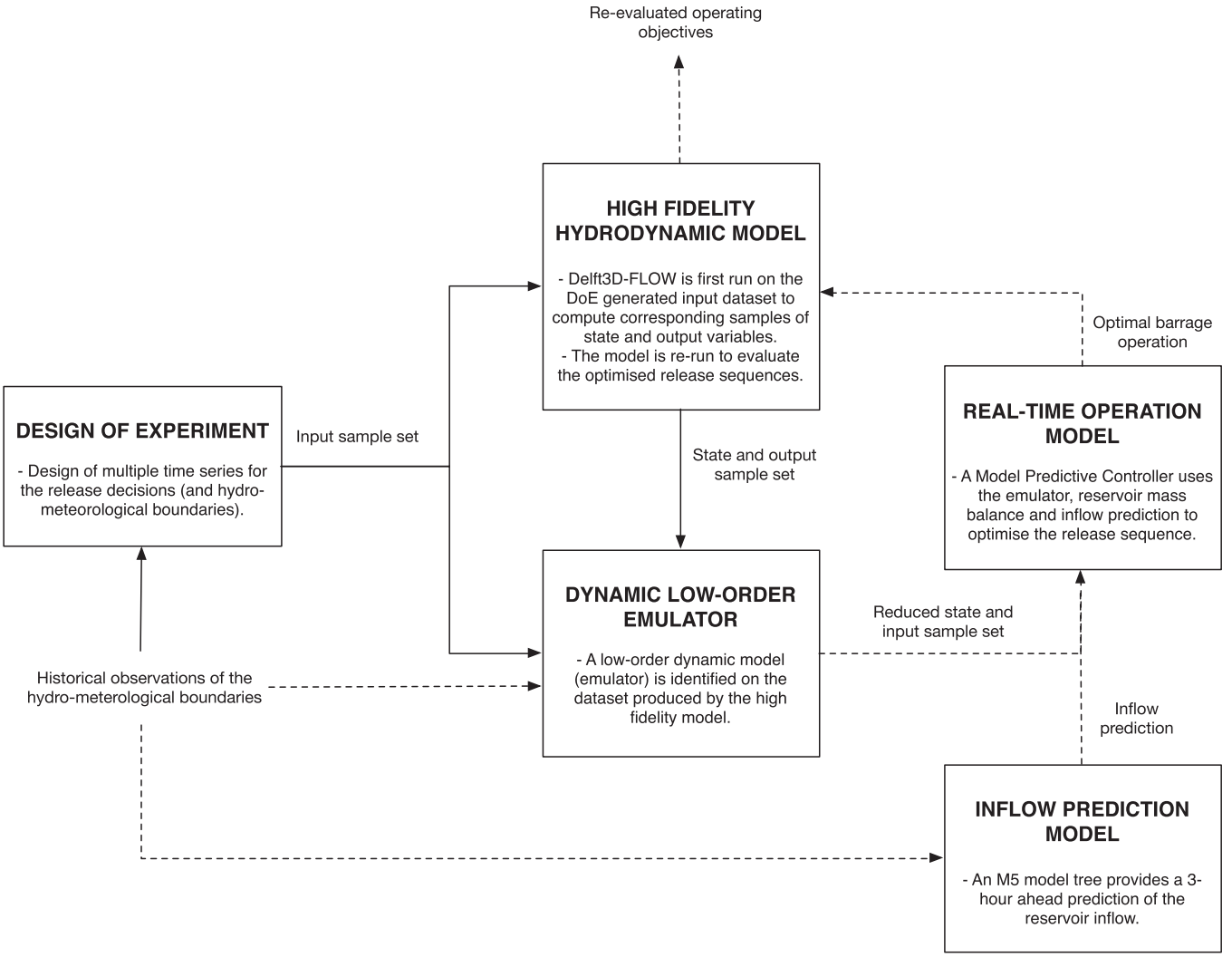

Figure 3. Schematization of the High-Performance Integrated Control framework adopted to combine Delft3D-FLOW, a low-order emulator and the Model Predictive Control scheme. The solid line is the DEMo workflow, the dashed line is the evaluation via optimization-simulation of the optimal barrage operation.

homogeneous clusters; so state variables belonging to a given cluster are more similar to each other than to variables belonging to a different cluster. Clusters can be seen as macroareas in the reservoir domain where the within-group variance is minimized and the between-group variance is maximized. In this study, we adopt the Hierarchical Agglomerative Clustering algorithm introduced by *Magni et al*. [2008]. Apart from its intrinsic simplicity, the algorithm has the characteristic of working directly on the "raw" data. This gives the twofold advantage of not requiring any preprocessing and preserving the physical interpretability of the reduced state variables. Unlike other algorithms, e.g., *k*-means [*Hartigan and Wong*, 1979], the desired number of clusters does not have to be specified a priori, but it is determined with an ex-post evaluation of the results. In this work, the evaluation is based on the Davies-Bouldin Index (DBI) [*Davies and Bouldin*, 1979], which favors clustering configurations with small within-cluster variance and high between-clusters variance. Overall, the optimal configuration is the one that minimizes the DBI value. Further details about the algorithm are given in section A1.

3.3.2.2. Variable Selection and Model Calibration

Based on the information content of $\tilde{\mathcal{F}}$, the dimensionality of the hydrodynamic model is further reduced by selecting the components of $\tilde{\mathbf{X}}_t$, $\mathbf{W}_t$, and $\mathbf{U}_t$ that will constitute the emulator's state $\mathbf{x}_t$, hydrometeorological boundaries $\mathbf{w}_t$ and release decision $\mathbf{u}_t$ vectors. This operation relies on variable selection algorithms, which can effectively handle large, multivariate data sets [*Guyon and Elisseeff*, 2003; *Galelli et al*., 2014b]. In this study, we adopt the Iterative Input variable Selection (IIS) algorithm—a stepwise, forward-selection

method introduced by *Galelli and Castelletti* [2013a]. The algorithm starts with an empty set of input variables, and then selects one variable per iteration according to a given measure of input significance. The selection process is stopped when the accuracy of an underlying data-driven model built with the selected variables does not significantly improve—according to a specific tolerance ε.

The IIS algorithm is applied in a recursive manner. First, the most relevant variables in explaining the output Y_t are selected among the vectors $\tilde{\mathbf{X}}_t$, $\mathbf{W}_t$, and $\mathbf{U}_t$. This gives the arguments $\mathbf{x}_t$, $\mathbf{w}_t$, and $\mathbf{u}_t$ of the function $h(\mathbf{x}_t, \mathbf{w}_t, \mathbf{u}_t)$ that yields the emulator's output y_t. For each state variable selected in the previous step, a new variable selection is performed to select the variables relevant to describe its dynamics. This gives the arguments of the function $\mathbf{f}(\mathbf{x}_t, \mathbf{w}_t, \mathbf{u}_t)$ associated with the considered state vector $\mathbf{x}_t$. If the second step leads to the selection of further variables from the vector $\tilde{\mathbf{X}}_t$—i.e., state variables not yet included in $\mathbf{x}_t$—the variable selection is recursively repeated, until all the selected state variables are given a dynamic description. Once this process is over, the arguments $\mathbf{x}_t$, $\mathbf{w}_t$, and $\mathbf{u}_t$ of the output transformation function $h(\cdot)$ and state transition equation $\mathbf{f}(\cdot)$ are known.

The output transformation and state transition equation are eventually validated on an independent set of observations generated with the DoE. Extra-Trees—a tree-based regression method introduced by *Geurts et al*. [2006]—are the model class for $\mathbf{f}(\cdot)$ and $\mathbf{h}(\cdot)$, since they are used by the IIS algorithm during the variable selection process. Further details about IIS algorithm and Extra-Trees, along with the adopted parameterization, are given in section A2.

4. Setting the Experiments

The generation of the data set $\mathcal{F}$ via simulation of Delft3D-FLOW requires determining the input $\mathbf{W}_t$ and $\mathbf{U}_t$. As far as the hydrometeorological boundaries are concerned, the data available for this study span a period of 33 months (all data were observed with hourly resolution during April 2009 to December 2011). A 21 month time series (April 2009 to December 2010) for each component of $\mathbf{W}_t$ is used in the DoE. Considering that Singapore is characterized by no distinctive dry and wet periods and that storm events have similar probability of occurrence throughout the year, the available data are considered representative of the seasonal hydrometeorological conditions across the bay [*NEA*, 2012]. The barrage operation is instead represented by 20 time series for each component of the release decision vector $\mathbf{U}_t$ generated with the aid of a pseudo-random number generator. To ensure that the release decisions depend on the hydrometeorological conditions, the following two-step strategy is adopted. First, the MPC scheme is used to compute the optimal release decisions at each decision time step t—with water quantity objectives only (equation (3))—on the basis of the current system's state and the predicted hydrometeorological conditions. Then, the value of these release decisions is modified by adding a disturbance created with the pseudo-random number generator. In other words, the release decisions exploration is enlarged to a small set of different values around the historical optimal ones. The combination of hydrometeorological and operating scenarios gives a total of 20 simulation scenarios and a data set $\mathcal{F}$ of about 300,000 hourly samples. The data corresponding to the period April–December 2009 are used for identifying the emulator (i.e., variable aggregation, selection, and model calibration), while those corresponding to January–December 2010 are used for validation. Finally, the hydrometeorological boundaries measured over the period January–December 2011 are used for an additional evaluation of the entire HPIC framework. This period is characterized by slightly wetter conditions (the average monthly rainfall is 221 mm, while it is 182 and 184 mm in 2009 and 2010, respectively), so this allows testing the framework on unseen inflow conditions.

5. Results and Discussion

5.1. Model Reduction

Following the procedure described in section 3.3, the high-dimensional data set produced by Delft3D-FLOW is used to identify a dynamic emulator of the salinity simulated at the control point.

5.1.1. Variable Aggregation

The time series clustering algorithm is applied to five groups of state variables simulated by Delft3D-FLOW, each containing salinity, temperature, and speed (horizontal and vertical components). This allows identifying different macroareas in which a specific state variable, for example salinity, assumes similar values over the simulation period. The clustering analysis is carried out by varying the final number of clusters between

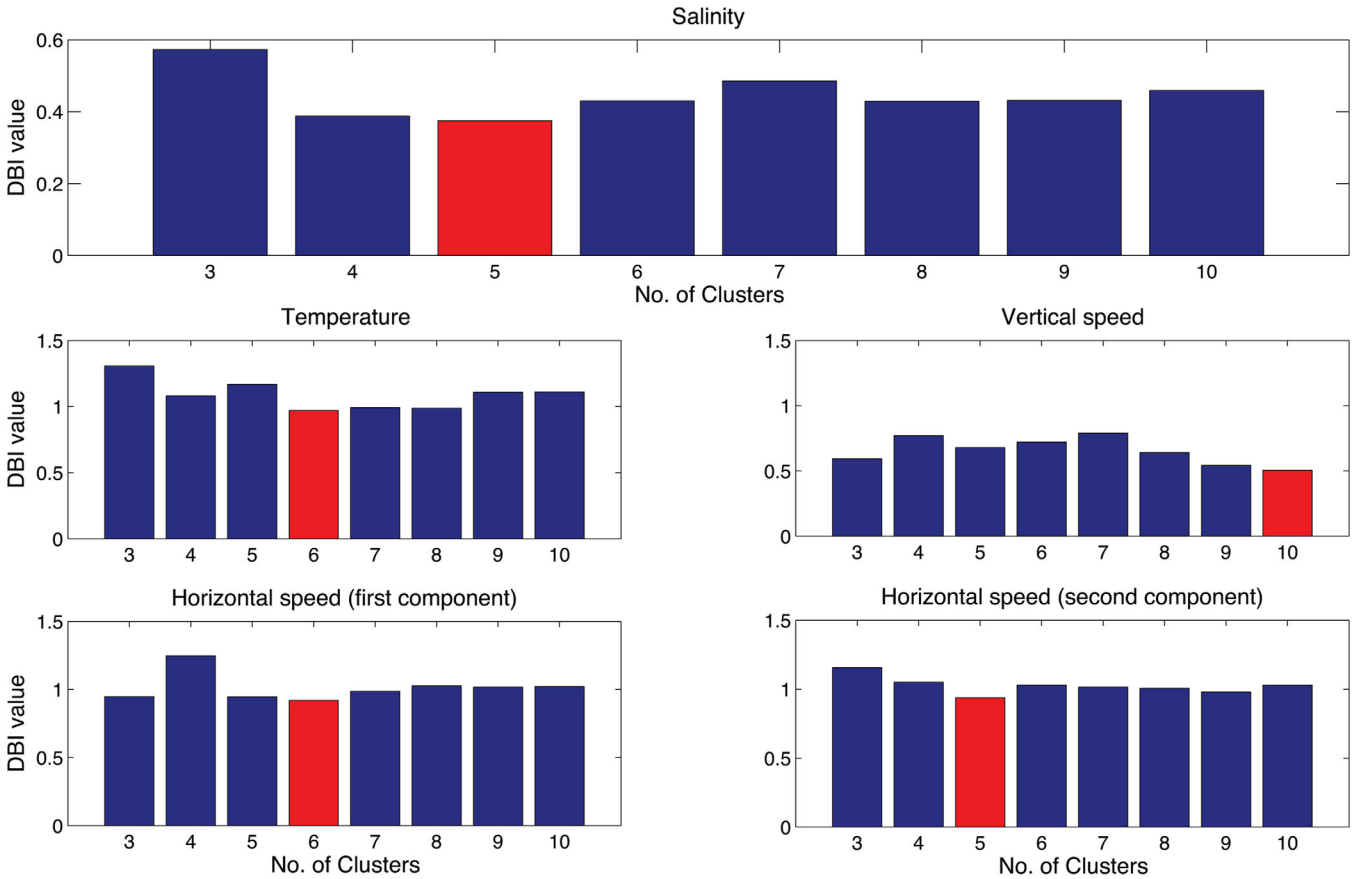

Figure 4. Value of the Davies-Bouldin Index (DBI) for a number of clusters included between 3 and 10; the red bars correspond to the optimal number of clusters. The time series clustering analysis is performed individually for the five state variables simulated by Delft3D-FLOW.

3 and 10, which represent the lower and upper bound of the desired number of clusters. The lower bound corresponds to a clustering configuration that favors dimensionality reduction, while the upper bound penalizes reduction in favor of a more accurate description of the physical processes. The accuracy of each clustering configuration is measured with the DBI, whose values are reported in Figure 4 for each group of state variables. Results show that the optimal number of clusters for salinity is equal to 5, which represents the best trade-off between dimensionality reduction and the characterization of the original physical process. Similar results are obtained for temperature and the horizontal components of water speed, while a higher number of clusters are found for vertical speed. This aggregation leads to an overall dimensionality reduction of about 2 orders of magnitude, since the state vector $\mathbf{X}_t$ is transformed into a lower dimensional vector $\tilde{\mathbf{X}}_t$ with 32 components.

A spatial representation of the five clusters identified for salinity is given in Figure 5, which shows the correspondence between the clustering configuration and the stratification conditions of the reservoir. Cluster 5 contains all points belonging to the upper layers of the reservoir, until a maximum depth of about 4 m from the surface. This is the largest area of the reservoir, characterized by freshwater conditions. Cluster 4 represents a sort of "buffer zone" at 4–5 m depth, located between this freshwater area and the thin saline layer on the reservoir bottom represented by Clusters 1–3. In particular, Clusters 1 and 2 contain the deepest points located in proximity of the barrage, where the groundwater seepage occurs. The reader is referred to the supporting information for a detailed description of the clustering results on water temperature and speed.

5.1.2. Variable Selection and Model Calibration

The candidate input variables considered for the variable selection process are the 32 components of the state vector $\tilde{\mathbf{X}}_t$ and the seven and four components of the hydrometeorological boundaries and release decisions vectors $\mathbf{W}_t$ and $\mathbf{U}_t$ (Tables 1 and 2). This results in a total of 43 candidate inputs. The IIS algorithm

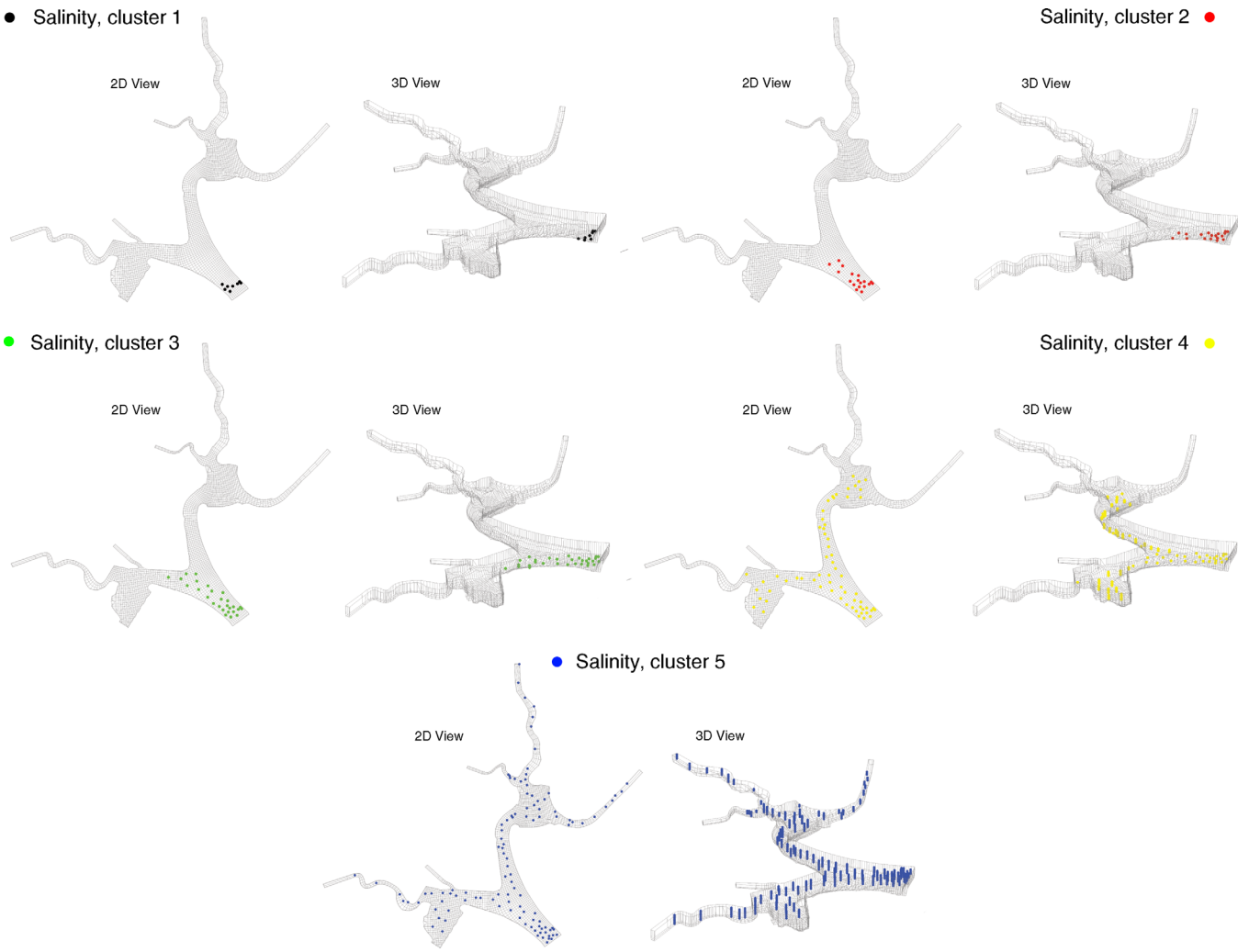

Figure 5. 2-D and 3-D spatial representation of the five clusters identified for the salinity values simulated by Delft3D-FLOW.

is first used to select the most relevant variables in explaining the output Y_t over a prediction horizon of 3 h. The results obtained are reported in Figure 6a, which shows the accuracy—measured in terms of coefficient of determination R^2—of the underlying Extra-Trees model as well as the contribution ΔR^2 of each selected variable. The model performance increases sharply with the selection of just one variable, and the selection of an additional variable (the groundwater seepage W_t^{gs}) does not lead to any further improvement. Unsurprisingly, the first variable is the state variable $\tilde{X}_t^{C2,s}$—the salinity in the second cluster (see Figure 5), where the control point is located. In other words, the dynamic behavior of the output Y_t can be accurately described (with an R^2 of 0.989, cross validation) as a function of the state variable $\tilde{X}_t^{C2,s}$, as graphically confirmed by the scatter plot of Figure 6b.

A second variable selection round is then necessary to identify the variables relevant to the dynamics of the state variable $\tilde{X}_{t+1}^{C2,s}$. Unlike the former case, the performance of the underlying model increases slowly with the number of selected variables, up to the fourth, when the IIS algorithm tolerance ε is reached (see Figure 6c). A high percentage of the salinity variance in the second cluster is described by three variables that relate to the main driving forces of salinity behavior: (1) the autoregressive term $\tilde{X}_t^{C2,s}$ represents the state of the system and, as such, is characterized by a relevant contribution ΔR^2 of about 0.778; (2) the release from the bottom pipes U_t^{pi} has a key role, since it contributes to discharge water with high salinity; and (3) the groundwater seepage W_t^{gs} has a direct, positive impact on the accumulation of salt in the bottom layers of Marina Reservoir. The individual effect of the selected variables with respect to $\tilde{X}_{t+1}^{C2,s}$ dynamics is shown in the scatter plots of Figures 6d–6f.

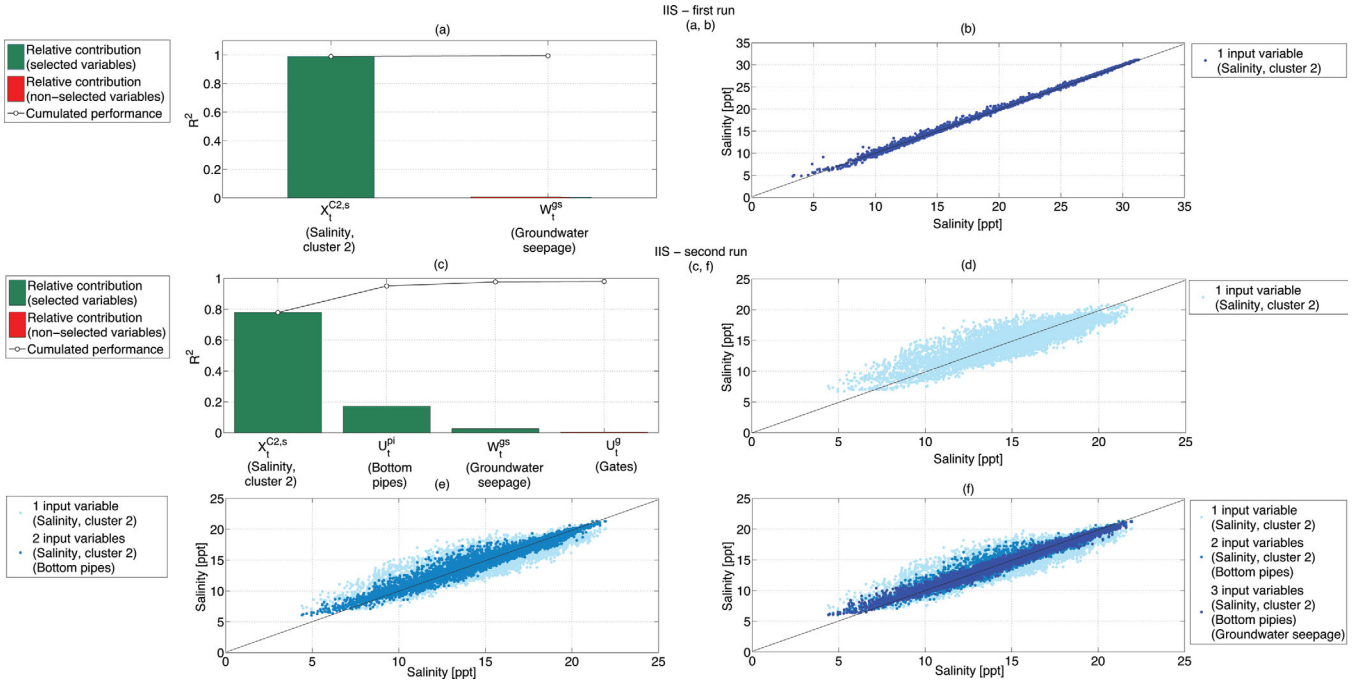

Figure 6. Performance improvement by each selected variable in the (a, c) first and second run of the IIS algorithm, with the continuous line representing the cumulated performance of the underlying models. Comparison between the measured and predicted salinity (b) at the control point and (d, e, and f) at the second cluster.

In synthesis, the selection of the variables to appear in the emulator takes two calls of the IIS algorithm to single out a state, hydrometeorological boundaries and release decision vectors $\mathbf{x}_t$, $\mathbf{w}_t$, and $\mathbf{u}_t$ with one component each. These are the salinity $\tilde{X}_t^{C2,s}$ in the second cluster, the groundwater seepage W_t^{gs}, and the release U_t^{pi} from the bottom pipes. The relation between these variables is illustrated in the causal network of Figure 7. The performance of the emulator is finally assessed over the validation period (January–December 2010) by means of the coefficient of determination R^2, which is equal to 0.970 for the output transformation and 0.965 for the state transition equation.

5.2. Real-Time Control

The emulator is employed for the operation of Marina Reservoir with water quantity and quality objectives. In particular, the emulator is used in combination with the reservoir mass balance equation to describe the salinity and storage dynamics over the 3 h ahead prediction horizon adopted by the MPC scheme. The water quality and quantity objectives are aggregated into a single operating objective with a convex combination giving the same importance to each (normalized) objective—that is, λ^w, λ^f, λ^P, and λ^s are all equal to 0.25 in equation (6). The effect of the release decisions is assessed via high-fidelity simulation with Delft3D-FLOW over three different periods (April–December 2009, January–December 2010, and January–December 2011) and compared against the effect of the release decisions designed by considering water quantity objectives only. With the adopted

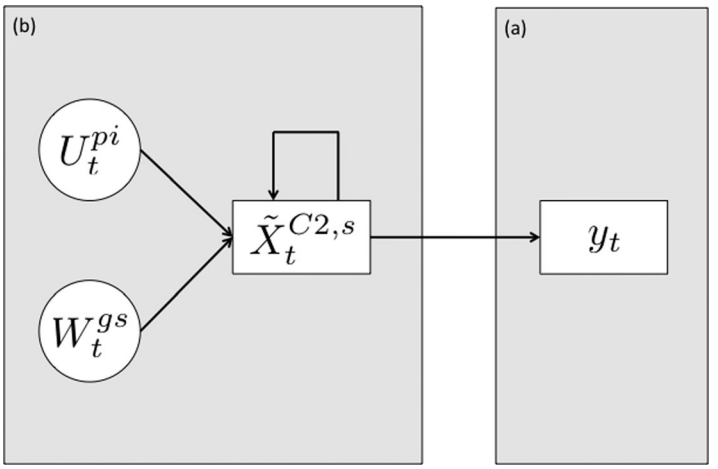

Figure 7. Graph representation of the variables involved in the (a) emulator output transformation function and (b) state transition equation.

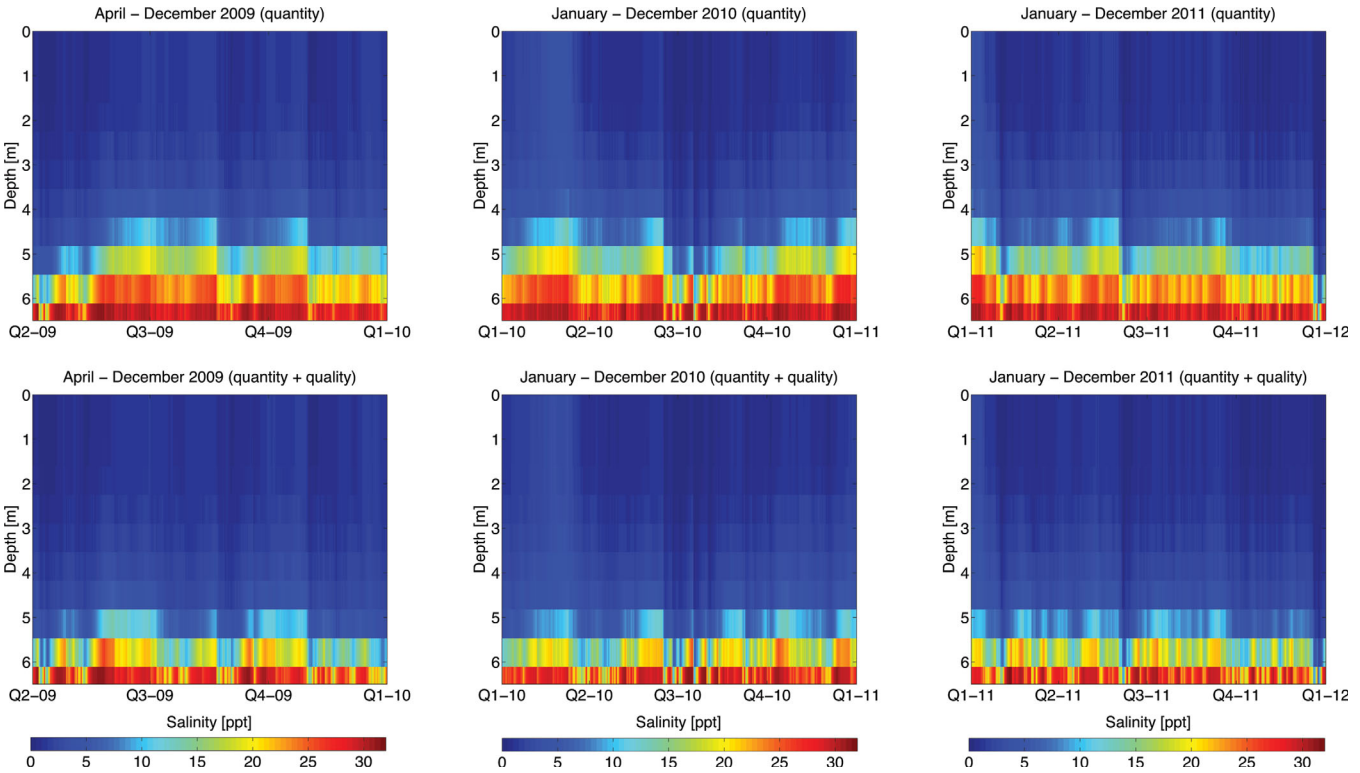

Figure 8. Salinity profiles across all layers of the control point for (left) April–December 2009, (middle) January–December 2010, and (right) January–December 2011 obtained by considering (top) water quantity and (bottom) both quality and quantity targets.

setting, the MPC scheme takes about 10 s to optimize the release decisions over a 3 h horizon. Considering that the adopted optimization algorithm is set to a maximum of 1000 iterations, the emulator provides a runtime improvement of about 10^4 with respect to Delft3D-FLOW. An additional validation of the HPIC framework would require a comparison between the trajectory of the operating objective (and associated release decisions) optimized with the emulator and the original simulation model. However, the computational requirements of Delft3D-FLOW do not allow for such numerical analysis. Indirectly, this represents a practical demonstration of the potential of the proposed HPIC framework, as with a high-fidelity simulation model any real-time optimal decisions would be simply impossible to be designed.

5.2.1. Water Quality

Figure 8 shows the salinity generated with the two real-time operation strategies across all layers of the control point. The salinity in the upper 3 m of the water column varies between 0.05 and 6–7 ppt with both strategies, while a significant difference is observed at greater depths. When accounting for water quantity objectives only (top), the salinity at the bottom of the reservoir has an average value of 28.31, 28.67, and 28.15 ppt (Table 4), with the layers below a depth of 3.5 m being affected by prolonged periods of high salinity. On the other hand, the adoption of the dynamic emulator allows the MPC scheme to account for the groundwater seepage and to operate the bottom pipes accordingly, resulting in an average 10% improvement of the mean salinity value at the control point. Furthermore, the diffusion and mixing processes are limited to the layers below a depth of 4.5 m, thus preserving the quality of the surface layers. The HPIC framework also improves the minimum value of the salinity in the deepest layer, with a decrease of 23, 30, and 59% on April–December 2009, January–December 2010 and January–December 2011, respectively, for the three evaluation periods (Table 4). Unsurprisingly, the maximum salinity does not vary across the management and hydrometeorological scenarios: during dry periods, it is not possible to discharge water from the bottom pipes, so the groundwater seepage rapidly increases the amount of salt water in the bottom part of the reservoir.

5.2.2. Water Quantity

The reduction in the salinity values at the control point has a positive effect on the salinity at the drinking water intake and, on its turn, on the water yield. During the period April–December 2009, January–

Table 4. Value of the Different Objectives Obtained Accounting for Water Quantity Only and Water Quantity and Quality Over the Period April–December 2009, January–December 2010, and January–December 2011

Objective	Unit of Measurement	April–December 2009		January–December 2010		January–December 2011	
		Quant.	Quant. + Qual.	Quant.	Quant. + Qual.	Quant.	Quant. + Qual.
Mean salinity	ppt	28.31	25.58	28.67	26.10	28.15	25.41
Min. salinity	ppt	13.38	10.24	12.84	8.93	5.31	2.17
Max. salinity	ppt	31.41	31.41	31.17	31.38	31.12	31.65
Wat. supply deficit	Mm^3/month	9.78	9.34	9.34	8.86	8.74	8.21
Flood control	h/month	6.33	5.35	6.50	5.17	10.92	10.00
Energy usage	Mm^3/month	1.95	1.58	1.80	1.65	3.04	2.80

December 2010, and January–December 2011, the monthly deficit of water supply decreases by about 450,000, 470,000, and 530,000 m^3, respectively (Table 4). This implies a decrease in the average annual deficit of about 5.80 Mm^3—almost twice the active storage of Marina Reservoir (about 4% of the total annual demand). Figure 9 compares the two management strategies over a 2 day period (26–27 July 2009) during which the salinity at the drinking water intake is close (or larger) to (than) the critical threshold of 0.25 ppt. The salinity obtained by accounting for water quantity objectives only (blue line, (a)) is often larger than 0.25 ppt, causing a provisional arrest of the drinking water intake (blue bars, (b)), while the emulation-based strategy achieves lower salinity values (red line, (a)), with a positive impact on the water intake (red bars, (b)). Such improvement is due to the operation of the bottom pipes (d), which are activated during low-tide events (c) and allow discharging water with high salinity. A comparison between the salinity values along the south-north cross section obtained by both operation strategies at the beginning of the 2 day period is given in Figure 10. It can be noticed that, when the bottom pipes are not operated (left), the salinity diffusion process is quite pronounced not only in the area in proximity of the barrage but also in the north-eastern part of the reservoir, where the drinking water intake is located. On the other hand, the salinity diffusion problem is limited by the operation strategy that accounts for water quantity and quality targets (right). A video of the salinity behavior during the 2 day period is provided in the supporting information.

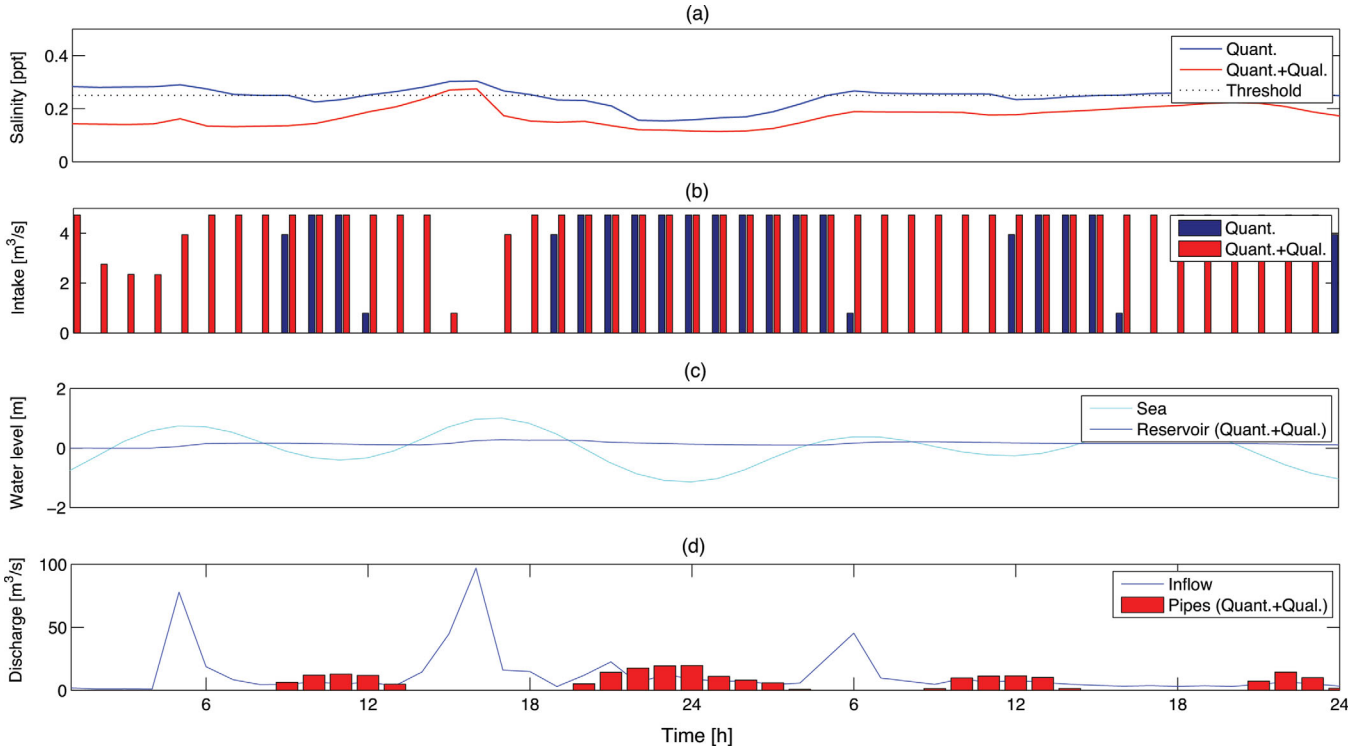

Figure 9. Salinity values at the drinking water intake during the period 26–27 July 2009, obtained by accounting for (a) both quality and quantity targets and (b) corresponding water intake. (c) Marina Reservoir and sea water level and (d) discharge from the bottom pipes regulated with the emulation-based strategy.

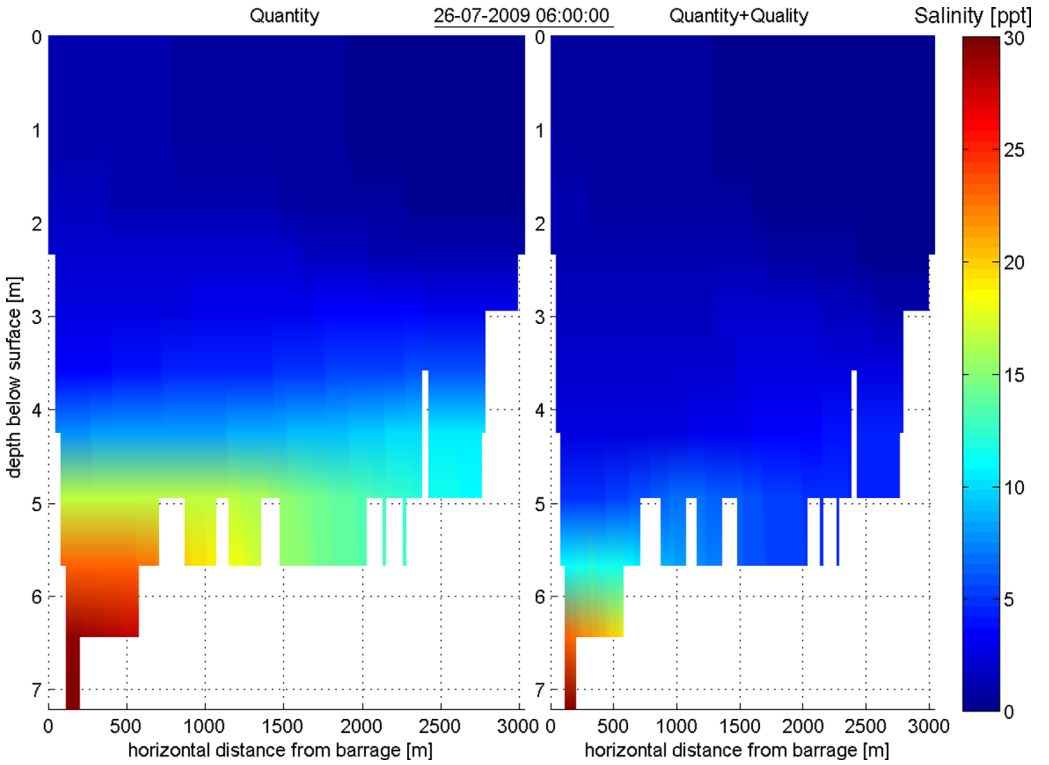

Figure 10. Salinity along the south-north cross section on 26 July 2009, at 6.00 A.M. obtained by accounting for (left) water quantity and (right) both quality and quantity targets.

The variation in the flood control and energy usage objectives is influenced by both management strategy and hydrometeorological boundaries. The HPIC framework entails a repeated use of the bottom pipes; this strategy tends to lower the water level in the reservoir, thus slightly decreasing the flood risk as well as the pumps usage. In year 2010, for example, the HPIC framework reduces these two operating objectives by about 20 and 8% with respect to the baseline strategy. A larger variation is instead due to the inflow process. The larger amount of rainfall and inflow events in year 2011 substantially increases the pumps usage and the deviation from the safety water level.

6. Conclusions

This study introduces a novel HPIC framework for the real-time operation of urban water reservoirs affected by water quality problems. The framework combines model reduction of 3-D hydroecological simulation models with a MPC scheme. Model reduction reduces the computational requirements of 3-D simulation models and thus makes their integration with real-time operation techniques effective; MPC guarantees prompt reaction against the fast hydrometeorological processes characterizing urban catchments. The model reduction relies on a semiautomatic procedural approach (DEMo), which consists of variable aggregation, selection, and calibration-validation of the dynamic emulator.

The application of the DEMo procedure to Delft3D-FLOW—a 3-D hydrodynamic model used to simulate nonsteady flow and transport phenomena in Marina Reservoir—shows that (1) the Hierarchical Time Series Clustering algorithm is successful in reducing the number of state variables by finding, for each specific variable, a few macroareas characterized by a similar pattern over time and (2) the Iterative Input variable Selection algorithm further reduces the complexity of the original model by selecting only the relevant and nonredundant variables to the output of interest. The dynamic emulator has three important features, that is, (1) accuracy in reproducing the output of the original model (R^2 of 0.970 on the validation period), (2) high complexity reduction (runtime improvement of about 10^4 with respect to Delft3D-FLOW), and (3) physical interpretability. This is an important aspect that differentiates the DEMo approach from other model reduction procedures (e.g., Principal Orthogonal Decomposition), which commonly condition the emulator

on a projection of the entire state vector. Yet the output-oriented model reduction approach may become unsuitable or computationally intensive for other modeling problems requiring knowledge of the entire state vector—for example, if multiple points within the reservoir domain were to be modeled.

The simulation results over the period April 2009 to December 2011 show that the HPIC framework decreases by about 37% the minimum value of the salinity at the control point and limits the diffusion process in the upper layers (with respect to a standard operating strategy aimed at water quantity targets only). This is obtained by discharging water with high salinity content through the operation of the bottom pipes during low-tide events. Although being occasionally limited by the presence of short dry periods, the improved real-time strategy has a positive effect on the deficit of drinking water supply, whose average monthly value decreases by about 483,000 m^3 thanks to the reduction of the salinity values at the drinking water intake. Such results are obtained not by resorting to any structural measure, but by simply exploiting the potential of High-Performance Integrated Control.

A limit of the present study is the validation of the entire HPIC framework. While the dynamic emulator is validated by (1) comparing the trajectory of the output variables against those produced by Delft3D-FLOW and (2) solving the control problem in real time, a thorough validation of the framework can only be performed by comparing the trajectory of the operating objective optimized with the emulator and Delft3D-FLOW. Yet the computational requirements of such high-fidelity simulation model do not allow for this analysis, and parallelization seems to be the only viable way to perform a more accurate emulator model diagnostic. This is a subject of ongoing research. Two additional methodological aspects deserve further investigation. The first concerns an exhaustive comparison against POD-based model reduction techniques, whereas the second concerns the reduction of more complex 3-D simulation models that reproduce not only flow and transport phenomena but also other processes such as the dynamics of phosphates, nitrates, or chlorophyll *a*.

Appendix A: Algorithms for Model Reduction

A1. Hierarchical Time Series Clustering

Similarly to static data clustering, time series clustering is based on algorithms that form clusters given a set of time series data—such as samples of the spatially distributed state variables generated via simulation with a 3-D hydrodynamic model. Most of the algorithms currently available modify the existing algorithms for clustering static data in two different ways: *raw-data-based* algorithms replace the distance/similarity measure for static data with an appropriate one for time series, while *feature-based* algorithms first convert a raw time series data into a feature vector of lower dimension and then apply a conventional clustering algorithm (see the review by *Liao* [2005]). The Hierarchical Time Series Clustering algorithm adopted in this study belongs to the first category [*Magni et al*., 2008]. At the first iteration, the algorithm places each time series in its own cluster, and it then merges these atomic clusters until a single cluster is obtained. The distance between each cluster is measured with the Euclidean metric, while clusters are merged according to Ward's criterion, which minimizes the total within-cluster variance and thus supports the creation of compact and well-separated clusters.

A2. Iterative Input Variable Selection and Extra-Trees

The Iterative Input variable Selection (IIS) algorithm combines a ranking-based measure of inputs significance with a stepwise, forward-selection approach [*Galelli and Castelletti*, 2013a]. The algorithm ranks each input by estimating its contribution, in terms of variance reduction, to the identification of an underlying model of the output. The first p variables in the ranking are then evaluated against the output by identifying p Single Input – Single Output (SISO) models, with the best performing input being added to the set of selected variables. At the first iteration, the ranking algorithm is run on a data set composed of all candidate inputs and corresponding output values, while for the subsequent iterations the original output values are replaced by the residuals of the underlying model identified in the previous step. This ensures that redundant inputs are not selected. The set of selected inputs is built incrementally by reiterating this process until the accuracy of the model built upon the selected set does not significantly improve—according to a prespecified tolerance ε—with the addition of a further variable. The model accuracy is assessed by calculating the average coefficient of determination R^2 across the k validation folds of a k-fold cross-validation process. The algorithm parameters p, k, and ε are set according to the guidelines provided by *Galelli and Castelletti*

[2013a]. The number p and k of SISO models and folds are both set to 5, while the algorithm tolerance ε is equal to 0.001. This means that the IIS algorithm is stopped when the selection of a further variable leads to an increase of R^2 lower than 0.001.

The two building blocks of the IIS algorithm are the ranking and the model building algorithms. Extra-Trees are used for both tasks, since their particular structure can be exploited to infer the relative importance of the different input variables. The Extra-Trees building algorithm creates an ensemble of M regression trees, each representing a different cascade of what-if conditions (nodes) aimed at predicting numerical values of the output variable [*Geurts et al*., 2006]. The tree-building process relies on the partitioning of the input variables space into mutually exclusive regions, progressively narrowing the size of the regions. Eventually, when the number of instances in a region becomes smaller than a specific user-defined value, the partitioning of that region stops and a leaf is created. Nodes are created with the following rule: K different inputs (cut-directions) are randomly selected and, for each one, a random value (cut-point) is chosen; a score is then associated with each cut-direction, with the one maximizing the score adopted to split the node. When the number of instances within the node is smaller than a user-defined number N_{min}, the algorithm stops partitioning a node and a leaf is created. Given a specific realization of input variables, the output predicted by a single tree is obtained by first following a specific path according to the nodes created during the building process and then averaging the values contained in the leaf. The prediction of the ensemble is obtained by averaging the predictions produced by all trees. This regression method can characterize strongly nonlinear relationships, is computationally efficient, and does not require any assumption on the statistical properties of the data set at hand. Furthermore, being a nonparametric method, Extra-Trees do not require any parameter optimization, whereas they provide good performance over a broad range of hyper-parameters. These features are particularly useful in an iterative ranking and regression process. As for the Extra-Trees setting, default values for the hyper-parameters M, K, and n_{min} are set according to *Geurts et al*. [2006] and *Galelli and Castelletti* [2013b]. The number M of trees in an ensemble is 500, the number K of alternative cut-directions is equal to the number of candidate inputs and n_{min}, the minimum cardinality for splitting a node, is 5.

Acknowledgments
The data used in this study were provided by Singapore's Public Utilities Board in the context of the Singapore-Delft Water Alliance Multi-Objective Multiple-Reservoir Management research program (R-303-001-005-272). As such, it is not possible to share these data with third parties. Findings, opinions, and conclusions expressed herein are those of the authors and do not represent the views or opinions of PUB. The first author is currently supported by the SRG ESD 2013 061 Start-up Research project. The authors are grateful to Hans Eikaas, Kok Meng Tan, and Daniel Twigt for their technical support during the early stages of this research.

References

Antenucci, J., K. Tan, H. Eikaas, and J. Imberger (2013), The importance of transport processes and spatial gradients on in situ estimates of lake metabolism, *Hydrobiologia*, *700*(1), 9–21.
Antoulas, A. (2005), An overview of approximation methods for large-scale dynamical systems, *Annu. Rev. Control*, *29*(2), 181–190.
Asher, M., B. Croke, A. Jakeman, and L. Peeters (2015), A review of surrogate models and their application to groundwater modeling, *Water Resour. Res.*, *51*, 5957–5973, doi:10.1002/2015WR016967.
Bach, P. M., W. Rauch, P. S. Mikkelsen, D. T. McCarthy, and A. Deletic (2014), A critical review of integrated urban water modelling—Urban drainage and beyond, *Environ. Modell. Software*, *54*, 88–107.
Barbosa, A., J. Fernandes, and L. David (2012), Key issues for sustainable urban stormwater management, *Water Res.*, *46*(20), 6787–6798.
Beck, M. (2005), Vulnerability of water quality in intensively developing urban watersheds, *Environ. Modell. Software*, *20*(4), 381–400.
Blanning, R. (1975), The construction and implementation of metamodels, *Simulation*, *24*(6), 177–184.
Boyce, S. E., and W. W.-G. Yeh (2014), Parameter-independent model reduction of transient groundwater flow models: Application to inverse problems, *Adv. Water Resour.*, *69*, 168–180.
Bray, B. S., and W. W.-G. Yeh (2008), Improving seawater barrier operation with simulation optimization in southern California, *J. Water Resour. Plann. Manage.*, *134*(2), 171–180.
Brown, R., M. Farrelly (2009), Delivering sustainable urban water management: A review of the hurdles we face, *Water Sci. Technol.*, *59*(5), 839–846.
Butler, D., and M. Schütze (2005), Integrating simulation models with a view to optimal control of urban wastewater systems, *Environ. Modell. Software*, *20*(4), 415–426.
Camacho, E. F., and C. B. Alba (2013), *Model Predictive Control*, Springer, London, U. K.
Cardoso, M., and L. J. Durlofsky (2010), Linearized reduced-order models for subsurface flow simulation, *J. Comput. Phys.*, *229*(3), 681–700.
Castelletti, A., F. Pianosi, and R. Soncini-Sessa (2008), Water reservoir control under economics, social and environmental constraints, *Automatica*, *44*(6), 1595–1607.
Castelletti, A., F. Pianosi, R. Soncini-Sessa, and J. Antenucci (2010), A multiobjective response surface approach for improved water quality planning in lakes and reservoirs, *Water Resour. Res.*, *46*, W06502, doi:10.1029/2009WR008389.
Castelletti, A., S. Galelli, M. Ratto, R. Soncini-Sessa, and P. Young (2012a), A general framework for dynamic emulation modelling in environmental problems, *Environ. Modell. Software*, *34*, 5–18.
Castelletti, A., S. Galelli, M. Restelli, and R. Soncini-Sessa (2012b), Data-driven dynamic emulation modelling for the optimal management of environmental systems, *Environ. Modell. Software*, *34*, 30–43.
Davies, D., and D. Bouldin (1979), A cluster separation measure, *IEEE Trans. Pattern Anal. Mach. Intel.*, *1*(2), 224–227.
Deltares (2014), *Delft3D-FLOW, User Manual 3.15.34158*, Delft, Netherlands.
Fletcher, T., H. Andrieu, and P. Hamel (2013), Understanding, management and modelling of urban hydrology and its consequences for receiving waters: A state of the art, *Adv. Water Resour.*, *51*, 261–279.
Forssell, U., and L. Ljung (2000), Some results on optimal experiment design, *Automatica*, *36*(5), 749–756.

Galelli, S., and A. Castelletti (2013a), Tree-based iterative input variable selection for hydrological modeling, *Water Resour. Res.*, *49*, 4295–4310, doi:10.1002/wrcr.20339.
Galelli, S., and A. Castelletti (2013b), Assessing the predictive capability of randomized tree-based ensembles in streamflow modelling, *Hydrol. Earth Syst. Sci.*, *17*, 2669–2684.
Galelli, S., A. Anand, A. Goedbloed, D. Schwanenberg, J. Pinho, and J. Vieira (2012), Multi-objective multiple-reservoir management. WP4: Multi-objective operational management of reservoirs, *Int. Res. Rep. SDWA-MR-12-04*, Singapore-Delft Water Alliance, Natl. Univ. of Singapore, Singapore.
Galelli, S., A. Goedbloed, D. Schwanenberg, and D. van Overloop (2014a), Optimal real-time operation of multi-purpose urban reservoirs: A case study in Singapore, *J. Water Resour. Plann. Manage.*, *140*(3), 511–523.
Galelli, S., G. B. Humphrey, H. R. Maier, A. Castelletti, G. C. Dandy, and M. S. Gibbs (2014b), An evaluation framework for input variable selection algorithms for environmental data-driven models, *Environ. Modell. Software*, *62*, 33–51.
Geurts, P., D. Ernst, and L. Wehenkel (2006), Extremely randomized trees, *Mach. Learning*, *63*(1), 3–42.
Guyon, I., and A. Elisseeff (2003), An introduction to variable and feature selection, *J. Mach. Learning Res.*, *3*, 1157–1182.
Hartigan, J., and M. Wong (1979), Algorithm AS 136: A *k*-means clustering algorithm, *J. R. Stat. Soc. Ser. C Appl. Stat.*, *28*(1), 100–108.
Joseph-Duran, B., C. Ocampo-Martinez, and G. Cembrano (2014), Hybrid modeling and receding horizon control of sewer networks, *Water Resour. Res.*, *50*(11), 8497–8514.
Kalteh, A. M., P. Hjorth, and R. Berndtsson (2008), Review of the self-organizing map (SOM) approach in water resources: Analysis, modelling and application, *Environ. Modell. Software*, *23*(7), 835–845.
Kristiana, R., J. Antenucci, and J. Imberger (2011), Sustainability assessment of the impact of the Marina Bay development on Singapore: Application of the index of sustainable functionality, *Int. J. Environ. Sustainable Dev.*, *10*(1), 1–35.
Liao, T. (2005), Clustering of time series data—A survey, *Pattern Recognition*, *38*(11), 1857–1874.
Lin, B., and D. McLaughlin (2014a), Efficient characterization of uncertain model parameters with a reduced-order ensemble Kalman Filter, *SIAM J. Sci. Comput.*, *36*(2), 198–224.
Lin, B., and D. McLaughlin (2014b), Real-time ensemble control with reduced-order modeling, *SIAM J. Sci. Comput.*, *36*(4), 749–775.
Liu, X., Q. Zhou, J. Birkholzer, and W. A. Illman (2013), Geostatistical reduced-order models in underdetermined inverse problems, *Water Resour. Res.*, *49*, 6587–6600, doi:10.1002/wrcr.20489.
Luyun, R., K. Momii, and K. Nakagawa (2011), Effects of recharge wells and flow barriers on seawater intrusion, *Ground Water*, *49*(2), 239–249.
MacWilliams, F., and N. Sloane (1976), Pseudo-random sequences and arrays, *Proc. IEEE*, *64*(12), 1715–1729.
Magni, P., F. Ferrazzi, L. Sacchi, and R. Bellazzi (2008), Timeclust: A clustering tool for gene expression time series, *Bioinformatics Appl. Note*, *24*(3), 430–432.
Marlow, D. R., M. Moglia, S. Cook, and D. J. Beale (2013), Towards sustainable urban water management: A critical reassessment, *Water Res.*, *47*(20), 7150–7161.
McDonald, R., P. Green, D. Balck, B. Fekete, C. Revenga, M. Todd, and M. Montgomery (2011), Urban growth, climate change, and freshwater availability, *Proc. Natl. Acad. Sci. U. S. A.*, *108*(15), 6312–6317.
McPhee, J., and W. W.-G. Yeh (2008), Groundwater management using model reduction via empirical orthogonal functions, *J. Water Resour. Plann. Manage.*, *134*(2), 161–170.
Mugunthan, P., and C. A. Shoemaker (2006), Assessing the impacts of parameter uncertainty for computationally expensive groundwater models, *Water Resour. Res.*, *42*, W10428, doi:10.1029/2005WR004640.
NEA (2012), *The Weather and Climate of Singapore*, Singapore.
Pasetto, D., A. Guadagnini, and M. Putti (2011), POD-based Monte Carlo approach for the solution of regional scale groundwater flow driven by randomly distributed recharge, *Adv. Water Resour.*, *34*(11), 1450–1463.
Razavi, S., B. A. Tolson, and D. H. Burn (2012), Review of surrogate modeling in water resources, *Water Resour. Res.*, *48*, W07401, doi:10.1029/2011WR011527.
Roweis, S. T., and L. K. Saul (2000), Nonlinear dimensionality reduction by locally linear embedding, *Science*, *290*(5500), 2323–2326.
Santner, T., B. Williams, and W. Notz (2003), *The Design and Analysis of Computer Experiments*, Springer, N. Y.
Siade, A. J., M. Putti, and W. W.-G. Yeh (2010), Snapshot selection for groundwater model reduction using proper orthogonal decomposition, *Water Resour. Res.*, *46*, W08539, doi:10.1029/2009WR008792.
Soncini-Sessa, R., A. Castelletti, and E. Weber (2007), *Integrated and Participatory Water Resources Management. Theory*, Elsevier, Amsterdam.
Tortajada, C. (2006), Water management in Singapore, *Water Resour. Dev.*, *22*(2), 227–240.
United Nations Population Fund (2011), State of the world population 2011: People and possibilities—In a world of 7 billions, *Tech. Rep. 2011 UNPF Rep.*, N. Y.
Ushijima, T. T., and W. W.-G. Yeh (2013), Experimental design for estimating unknown groundwater pumping using genetic algorithm and reduced order model, *Water Resour. Res.*, *49*, 6688–6699, doi:10.1002/wrcr.20513.
van Doren, J. F., R. Markovinović, and J.-D. Jansen (2006), Reduced-order optimal control of water flooding using proper orthogonal decomposition, *Comput. Geosci.*, *10*(1), 137–158.
Vermeulen, P., A. Heemink, and C. Te Stroet (2004), Reduced models for linear groundwater flow models using empirical orthogonal functions, *Adv. Water Resour.*, *27*(1), 57–69.
Vermeulen, P., C. Te Stroet, and A. Heemink (2006), Model inversion of transient nonlinear groundwater flow models using model reduction, *Water Resour. Res.*, *42*, W09417, doi:10.1029/2005WR004536.
Waltz, R., J. Morales, J. Nocedal, and D. Orban (2006), An interior algorithm for nonlinear optimization that combines line search and trust region steps, *Math. Program.*, *107*(3), 391–408.
Werner, A. D., M. Bakker, V. E. Post, A. Vandenbohede, C. Lu, B. Ataie-Ashtiani, C. T. Simmons, and D. A. Barry (2013), Seawater intrusion processes, investigation and management: Recent advances and future challenges, *Adv. Water Resour.*, *51*, 3–26.
Willcox, K., and J. Peraire (2002), Balanced model reduction via the proper orthogonal decomposition, *AIAA J.*, *40*(11), 2323–2330.
Wu, B., Y. Zheng, Y. Tian, X. Wu, Y. Yao, F. Han, J. Liu, and C. Zheng (2014), Systematic assessment of the uncertainty in integrated surface water-groundwater modeling based on the probabilistic collocation method, *Water Resour. Res.*, *50*, 5848–5865, doi:10.1002/2014WR015366.
Wu, B., Y. Zheng, X. Wu, Y. Tian, F. Han, J. Liu, and C. Zheng (2015), Optimizing water resources management in large river basins with integrated surface water-groundwater modeling: A surrogate-based approach, *Water Resour. Res.*, *51*, 2153–2173, doi:10.1002/2014WR016653.

Xu, M., P. Van Overloop, N. Van De Giesen, and G. Stelling (2010), Real-time control of combined surface water quantity and quality: Polder flushing, *Water Sci. Technol.*, *61*(4), 869–878.

Xu, M., P. Van Overloop, and N. Van De Giesen (2013), Model reduction in model predictive control of combined water quantity and quality in open channels, *Environ. Modell. Software*, *42*, 72–87.

Yan, S., and B. Minsker (2010), Applying dynamic surrogate models in noisy genetic algorithms to optimize groundwater remediation designs, *J. Water Resour. Plann. Manage.*, *137*(3), 284–292.

Young, P. C. (2011), *Recursive Estimation and Time-Series Analysis: An Introduction for the Student and Practitioner*, Springer, Berlin.

Zijl, F., and D. Twigt (2007), Singapore Marina Reservoir study. Hydrodynamic modelling, *Res. Rep. Z.4265.10/20/30*, Deltares, Delft, Netherlands.