Control Engineering Practice 81 (2018) 199–206

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

A modeling and distributed MPC approach for water distribution networks

Felix Berkel *, Sebastian Caba, Jonas Bleich, Steven Liu

Institute of Control Systems, Department of Electrical and Computer Engineering, University of Kaiserslautern, Erwin-Schroedinger-Str. 12, 67663 Kaiserslautern, Germany

ARTICLE INFO

Keywords:
Model predictive control
Drinking water networks
Distributed control
Event-triggered communication
Economic model predictive control

ABSTRACT

This paper deals with the modeling and control of water distribution networks consisting of multiple pressure zones that are interconnected by fixed-speed pumps. A simplified control-oriented model with linear dynamics and non-convex constraints which considers the pressures and losses in the zones is derived. Based on the model, a periodic distributed model predictive scheme which is tailored to the non-convex constraints and minimizes an economic objective is introduced. The scheme uses event-triggered communication between the subsystems to reduce the communicational load. The proposed model and controller are evaluated using a model of the water distribution network of the city of Kaiserslautern, Germany.

1. Introduction

Water distribution networks (WDNs) transport water from supply systems to the consumers. The systems are usually large-scale, complex, constrained and highly nonlinear, for instance, due to losses in the pipes and pump characteristics. Moreover, the systems have a non-continuous character as the pumps are often switched on/off since the water demand can vary strongly over the day (Brdys & Ulanicki, 1994). Considering several days, the water demand does not change significantly from day to day, i.e., the WDNs have a periodic character.

The major requirement for the operational control of the WDN is that the water demand in the system is covered at all times as water is vital for humans. The supply must also be guaranteed for safety reasons, e.g., in the case of fire. Further requirements are low operational costs.

Due to the complexity and the large-scale character of the plant, WDNs are classically controlled hierarchically. A higher-level controller minimizes an economic cost function and provides references which are tracked by a lower-level controller, see Scattolini (2009). Model predictive control (MPC) is often applied at a higher-level due to the possibility of incorporating future predictions of the water demand. The explicit consideration of the constraints allows operations close to the boundaries (Rawlings & Mayne, 2009). Recently, there is a trend to use economic MPC (EMPC) where a single-layer controller directly minimizes an economic cost function, see, e.g., Angeli, Amrit, and Rawlings (2012) and Rawlings, Angeli, and Bates (2012). EMPC for periodic systems is studied in Zanon, Gros, and Diehl (2013). The application of EMPC to WDNs has been investigated, e.g., in Grosso, Ocampo-Martínez, Puig, Limon, and Pereira (2014) and Limon, Pereira, De La Peña, Alamo, and Grosso (2014).

Due to the high dimensionality and the complexity of WDNs, distributed MPC (DMPC) has gained great interest in this area. Here the plant is decomposed into several subsystems and each subsystem has a local controller. The controllers communicate with each other to coordinate their decision making. A good overview can be found in Christofides, Scattolini, Muñoz del la Peña, and Liu (2013) and Negenborn and Maestre (2014). In Ocampo-Martinez, Barcelli, Puig, and Bemporad (2012) a hierarchical MPC scheme with decentralized MPC controllers at the lower-level for the Barcelona WDN is investigated. The authors of Leirens, Zamora, Negenborn, and Schutter (2010) apply the alternating direction of multiplier method (ADMM) to solve the control problem in a distributed fashion. In Grosso, Ocampo-Martinez, and Puig (2017) the economic feasible cooperation-based DMPC of Lee and Angeli (2014) is applied to WDNs.

The set-up considered in this paper is motivated by the WDN of the city of Kaiserslautern, Germany, which is located in a mountainous region. The WDN is divided into several pressure zones which are interconnected by fixed-speed pumps. Although variable-speed pumps where the flow can be adjusted independently from the pressure are spreading more and more, in many WDNs still fixed-speed pumps or pumps with variable but gradual speed stages are installed. Here the flow cannot be set independently from the pressure which requires the explicit consideration of the different pressures in the system. Those are neglected in the controller models of the publications mentioned above. Thus, a novel control-oriented model of the WDN is derived considering the special topology, the pressures, the presence of fixed-speed pumps and the losses in the system. The resulting model has linear dynamics but is subject to non-convex constraints. Moreover, the pressure zones (which correspond to the subsystems) are coupled in the dynamics over the inputs, are subject to common constraints and have a cascaded structure. A DMPC control scheme minimizing an economic objective and running the simplified model inside is designed. The

* Corresponding author.
E-mail addresses: berkel@eit.uni-kl.de (F. Berkel), caba@eit.uni-kl.de (S. Caba), jonas.bleich@gmx.de (J. Bleich), sliu@eit.uni-kl.de (S. Liu).

https://doi.org/10.1016/j.conengprac.2018.09.017
Received 20 March 2018; Received in revised form 11 September 2018; Accepted 16 September 2018
Available online 27 September 2018
0967-0661/

DMPC scheme is tailored to the non-convexity as well as to the special structure in couplings and constraints. To reduce the communication load, the algorithm uses event-triggered communication between the subsystems. This means that the controllers of each subsystem only need to send information to other subsystems when a predefined threshold is violated. Feasibility and performance guarantees for the closed-loop system can be provided. The resulting optimization problem in each controller can be recast as a mixed-integer quadratic problem (MIQP) for which efficient solvers exist. The algorithm is derived in general form and can thus be applied to other applications with input coupling in the dynamics. The main motivation for the application of a DMPC is that it is more robust than a centralized controller in the following sense: While the overall system cannot be controlled anymore if the centralized controller fails, with a distributed controller at least some subsystems could still be operated. Moreover, the online optimization problems of the DMPC are less complex compared to the ones of the centralized controller.

Note that DMPC with event-triggered communication for input-coupled subsystems is also investigated in Gross and Stursberg (2016) which is based on the iterative method in Stewart, Venkat, Rawlings, Wright, and Pannocchia (2010) where each subsystem applies a convex combination of the actual and a previously computed input. However, the method in Gross and Stursberg (2016) assumes that the MPC optimization problem is convex which is not the case for the control-oriented model of the WDN. Therefore, no convex combination can be applied. The concept of event-triggered communication in DMPC is also studied in the previous work (Berkel & Liu, 2018) but for subsystems coupled in the dynamics over the state and subject to convex constraints.

The paper is structured as follows: Section 2 introduces the notation and reviews the modeling of the required components in WDNs. In Section 3 the control-oriented model is derived and in Section 4 the control problem is formulated. Moreover, the DMPC scheme is introduced and discussed. A simulation example is given in Section 5. Section 6 concludes the paper.

2. Preliminaries

2.1. Notation

$\mathcal{I}_{n_1:n_2}$ is the set of integers from n_1 to n_2. $|\mathcal{S}|$ is the cardinality of a set $\mathcal{S}$. $\boldsymbol{D} = \operatorname{diag}\left[\boldsymbol{D}_{11}, \dots, \boldsymbol{D}_{nn}\right]$ denotes a block-diagonal matrix with matrices $\boldsymbol{D}_{11}, \dots, \boldsymbol{D}_{nn}$ on its diagonal. The vector $\boldsymbol{v} = \operatorname{col}_{i\in\mathcal{I}_{n_1:n_2}}\left(\boldsymbol{v}_i\right)$ is the vector obtained from stacking the vectors $\boldsymbol{v}_{n_1}, \dots, \boldsymbol{v}_{n_2}$. For two integers n and $m \neq 0$, $[n]_m$ is the modulo operation.

2.2. Modeling of components in a WDN

Next, the general modeling of components in a WDN which are required for later investigations are reviewed. If not stated otherwise, this section follows the modeling of Brdys and Ulanicki (1994, chapter 2).

A WDN can be described by a graph consisting of nodes and branches. The branches can be pipes, valves or pump stations and link the nodes with each other. Based on the flow continuity law for all nodes i

$$\sum_{j\in\mathcal{N}_i} q_{ij} = d_{\mathrm{W}i} \tag{1}$$

holds where the set $\mathcal{N}_i$ consists of the indexes of the neighbor nodes of node i, i.e., the nodes where there is a branch to. The flows in the branches are q_{ij} and $d_{\mathrm{W}i}$ is the demand in this node. The hydraulic head h_i (which corresponds to the pressure and is also called head in the sequel) in the node i is given by

$$h_i = E_i + p_i, \tag{2}$$

i.e., the sum of elevation E_i and the water pressure head p_i.

Losses in a pipe are given in terms of a head drop by the Hazen–Williams formula

$$\Delta h_{\mathrm{p}ij} = R_{ij} q_{ij} |q_{ij}|^{0.852} \tag{3}$$

where R_{ij} is the pipe resistance. This empirical relationship can be computed based on the pipe length, diameter and roughness coefficient if available.

The head difference between two nodes connected with a branch consisting of a pump station with n_{ij}^{p} identical variable speed pumps operated at the same pump speed ω_{ij} is given by the characteristic

$$\Delta h_{\mathrm{s}ij} = \begin{cases} \left[\frac{c_{2ij}}{\delta_{ij}}\right] q_{ij}^2 + \left[\frac{c_{1ij}}{\delta_{ij}}\right] \omega_{ij} q_{ij} + c_{0ij}\omega_{ij}^2, & \delta_{ij} \neq 0 \\ h_j - h_i, & \delta_{ij} = 0 \end{cases} \tag{4}$$

with $q_{ij} \geq 0$, the number of running pumps δ_{ij} and the pump parameters c_{2ij}, c_{1ij} and c_{0ij} which are given by the pump manufacturer. In the case $\delta_{ij} = 0$, i.e., no pump is running, the flow $q_{ij} = 0$ as check valves prevent a flow from the higher potential to the lower one. Therefore, the nodes are decoupled and the head difference is directly given by $\Delta h_{ij} = h_j - h_i$. When at least one pump is running, the head increase is given by the upper relation in (4) which is quadratic in the flow q_{ij}. To adjust the flow to time-varying demands, either the speed ω_{ij} or the number of pumps running can be changed. Although the speed could be changed continuously, in this paper it is assumed that a small number of fixed speed levels are used which is mostly the case in practical applications. This means that $\omega_{ij} \in \mathcal{W}_{ij}$ where $\mathcal{W}_{ij}$ is the set of discrete pump speeds. If the changes in the demands are large, several pumps are installed in parallel in a pump station. The number of pumps running in parallel δ_{ij} belongs to the set $\mathcal{I}_{0:n_{ij}^{\mathrm{p}}}$ where n_{ij}^{p} is the number of pumps in the station. The combinations of running pumps and speeds $(\delta_{ij}, \omega_{ij})$ are called pump configurations in the sequel. To simplify the notation, the configurations are indexed with $k \in \mathcal{I}_{1:n_{ij}^{\mathrm{con}}}$ where the number of configurations in a pump station is called n_{ij}^{con}. By definition $k = 1$ corresponds to the case where $\delta_{ij} = 0$. Note that for each $k \in \mathcal{I}_{2:n_{ij}^{\mathrm{con}}}$, the coefficients in (4) are constants and so (4) can be recast as

$$\Delta h_{\mathrm{s}ij} = \begin{cases} \tilde{c}_{2ij}^k q_{ij}^2 + \tilde{c}_{1ij}^k q_{ij} + \tilde{c}_{0ij}^k, & k > 1 \\ h_j - h_i, & k = 1. \end{cases} \tag{5}$$

To put it in a nutshell, each pump configuration is modeled as a single fixed-speed pump or in other words each pump station with multiple pumps and fixed-speeds is modeled as multiple pumps with single fixed-speed. Note that only one of the configurations can be active at the same time.

A precise pump power approximation is given in Caba, Lepper, and Liu (2016) by

$$P_{ij} = b_{0ij}^k + b_{1ij}^k q_{ij} + b_{2ij}^k q_{ij}^2 + b_{3ij}^k q_{ij}^3 \tag{6}$$

with the pump parameters $b_{0ij}^k, b_{1ij}^k, b_{2ij}^k, b_{3ij}^k$ which are dependent on the pump configuration and can be obtained from the pump manufacturer.

Another important component of a WDN is a reservoir where water can be stored. Analogous to (2), the reservoir head

$$h_{\mathrm{r}i} = E_{\mathrm{r}i} + x_{\mathrm{r}i} \tag{7}$$

is defined by its elevation $E_{\mathrm{r}j}$ and water level $x_{\mathrm{r}j}$. The reservoir dynamic depends on the flow difference between ingoing flow $q_{\mathrm{in}i}$ and outgoing flow $q_{\mathrm{out}i}$

$$\dot{h}_{\mathrm{r}i}(t) = \frac{1}{S_{\mathrm{r}i}}\left(q_{\mathrm{in}i}(t) - q_{\mathrm{out}i}(t)\right) \tag{8}$$

with the constant cross-sectional area $S_{\mathrm{r}i}$ and time $t \in \mathbb{R}_{\geq 0}$.

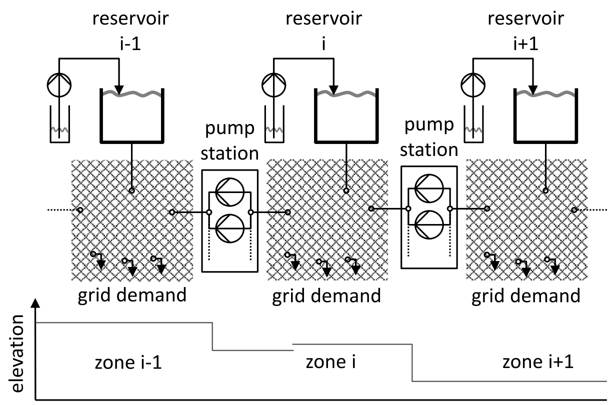

Fig. 1. Sketch of considered water distribution networks.

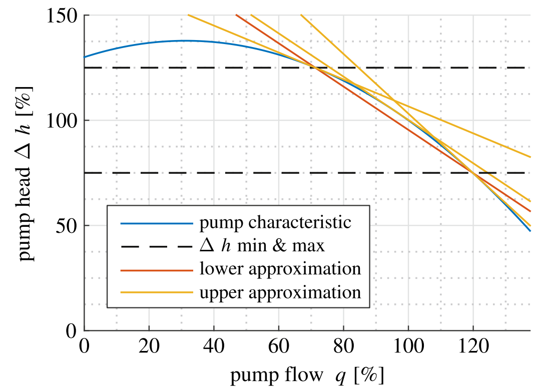

Fig. 2. Polytopic approximation of a pump characteristic.

3. Control-oriented modeling

Combining the elements according to the given network topology, the behavior of the complete WDN can be modeled. However, such models are usually not feasible for controller design and simpler models adapted to the properties of the considered WDN are required (Brdys & Ulanicki, 1994). In this paper such a control-oriented model is derived for WDNs with multiple cascaded pressure zones as sketched in Fig. 1. The structure is motivated by the structure of the WDN in Kaiserslautern where due to the geographic conditions the WDN is divided into M pressure zones each corresponding to an area with similar elevation. Each zone is aggregated to a node with a single reservoir and demand. The zones are linked via pump stations to transport water to zones with higher elevation.

Discretizing (8) and applying (1), the reservoir head $h_{\mathrm{r}i}$ with $i \in \mathcal{I}_{1:M}$ can be described by the discrete-time model

$$h_{\mathrm{r}i,t+1} = h_{\mathrm{r}i,t} + s_{\mathrm{r}i}(q_{\mathrm{s}i,t} + q_{i+1i,t} - q_{ii-1,t} - d_{\mathrm{W}i,t}) \tag{9}$$

where $t \in \mathbb{N}$ are the sampling instances from now on, $s_{ri} = \frac{T_{\mathrm{s}}}{S_{ri}}$, T_{s} is the sampling time, $q_{\mathrm{s}i}$ and $d_{\mathrm{W}i}$ are the supply flow and the demand in the zone, respectively. q_{i+1i} is the flow from zone $i+1$ to i. For zone 1 and M, $q_{10} = 0$ and $q_{M+1M} = 0$, respectively.

To account for the losses in the pipes of the zones, they are lumped into the links between the pressure zones. Due to the almost quadratic form of (3), the approximation

$$\Delta h_{\mathrm{p}ij} = \tilde{R}_{ij} q_{ij}^2 \tag{10}$$

yields a small error and $\tilde{R}_{ij}$ can be estimated via measurements. The losses are then included into the pump characteristic (5) which leads to

$$\Delta h_{\mathrm{s}ij} = \begin{cases} \left(\tilde{c}_{2ij}^k - \tilde{R}_{ij}\right) q_{ij}^2 + \tilde{c}_{1ij}^k q_{ij} + \tilde{c}_{0ij}^k, & k > 1 \\ h_j - h_i, & k = 1. \end{cases} \tag{11}$$

Integrating (11) into an optimization problem for determining a control law would lead to a nonlinear program. As such problems can be hard to solve, an approximation is introduced. Thereby, the pump characteristic for each pump configuration is approximated by a polytope around the nominal operation point. The procedure for determining the approximation is described in the following and is shown in Fig. 2. In a first step, the range $\left(h_i, h_j\right) \in \mathcal{H}_{ij} \subseteq \mathbb{R}^2$ in which $\Delta h_{ij} = h_j - h_i$ can lie is determined. This range is bounded as the water levels in the reservoirs are bounded. Then, a straight line is determined through the intersection points of $\mathcal{H}_{ij}$ with the pump characteristic curve (lower approximation). A predefined number of lines which are tangentially to the pump characteristic curve build the upper approximation. The intersection of the half-spaces spanned by this lines defines the polytope $\tilde{\mathcal{P}}_{ij}^k = \left\{ \boldsymbol{x} \in \mathbb{R}^3 | \boldsymbol{H}_{ij}^k \boldsymbol{x} \leq \boldsymbol{f}_{ij}^k \right\}$ for a configuration $k \in \mathcal{I}_{2:n_{ij}^{\mathrm{con}}}$. As mentioned before, when the pumps in the stations are switched off, check valves prevent a flow from the higher potential to the lower one. Therefore the nodes are decoupled, $q_{ij} = 0$ and the head drop is directly given by $\Delta h_{ij} = h_j - h_i$. Thus heads and flow are in the set $\tilde{\mathcal{P}}_{ij}^1 = \left\{ x_1, x_2, x_3 \in \mathbb{R} | x_2 - x_1 \in \mathcal{H}_{ij} \wedge x_3 = 0 \right\}$.

Considering all configurations in a pump station, one can define the union of the polytopes $\mathcal{P}_{ij} = \bigcup_{k \in \mathcal{I}_{1:n_{ij}^{\mathrm{con}}}} \tilde{\mathcal{P}}_{ij}^k$. To ensure that $\left(h_i, h_j, q_{ij}\right)$ satisfy the pump characteristic approximately, the constraint

$$\left(h_i, h_j, q_{ij}\right) \in \mathcal{P}_{ij} \tag{12}$$

is included into the controller optimization problem as shown later. Note that the constraint is non-convex but can be integrated in the optimization as described in Section 5.

The power relation (6) can be approximated in the operation area by a piecewise affine function, i.e.,

$$P_{ij}^k(q_{ij}) = \tilde{b}_{1ij}^k q_{ij} + \tilde{b}_{0ij}^k \text{ for } q_{ij}^{\min k} \leq q_{ij} \leq q_{ij}^{\max k} \tag{13}$$

where k indexes again the configuration, $q_{ij}^{\min k}$ and $q_{ij}^{\max k}$ are minimal and maximum flow for a configuration. For the case that all pumps are switched off, $P_{ij}^1(q_{ij}) = 0$.

4. Distributed control strategy

Consider the system which is composed of the subsystems $i \in \mathcal{I}_{1:M}$ and is of the form

$$\begin{aligned} &\boldsymbol{x}_{1,t+1} = \boldsymbol{A}_{11}\boldsymbol{x}_{1,t} + \boldsymbol{B}_{11}\boldsymbol{u}_{1,t} + \boldsymbol{d}_{1,t} \\ &\text{and for all } i \in \mathcal{I}_{2:M}: \\ &\boldsymbol{x}_{i,t+1} = \boldsymbol{A}_{ii}\boldsymbol{x}_{i,t} + \boldsymbol{B}_{ii}\boldsymbol{u}_{i,t} + \boldsymbol{B}_{ii-1}\boldsymbol{u}_{i-1,t} + \boldsymbol{d}_{i,t} \end{aligned} \tag{14}$$

with $t \in \mathbb{N}$. The states $\boldsymbol{x}_i \in \mathbb{R}^{n_i}$ and inputs $\boldsymbol{u}_i \in \mathbb{R}^{m_i}$ of the subsystems are subject to the constraints $\boldsymbol{x}_i \in \mathcal{X}_i \subseteq \mathbb{R}^{n_i}$ and $\boldsymbol{u}_i \in \mathcal{U}_i \subseteq \mathbb{R}^{m_i}$, respectively, as well as to the coupled constraints $\left(\boldsymbol{x}_i, \boldsymbol{x}_{i+1}, \boldsymbol{u}_i\right) \in \mathcal{X}_i^{\mathrm{C}} \in \mathbb{R}^{2n_i+m_i}$. The sets $\mathcal{X}_i$, $\mathcal{X}_i^{\mathrm{C}}$ and $\mathcal{U}_i$ are compact sets. $\boldsymbol{d}_i \in \mathbb{R}^{n_i}$ is a known exogenous input for which predicted values are available. System (14) has a cascaded structure and the subsystems are input-coupled. In the sequel subsystem $i-1$ and $i+1$ are called the predecessor and successor of subsystem i, respectively.

The WDN described by (9) can be recast in the form (14). Thereby, a subsystem corresponds to a pressure zone and $\boldsymbol{x}_i = h_{\mathrm{r}i}$, $\boldsymbol{u}_i = \begin{bmatrix} q_{\mathrm{s}i} & q_{i+1i} \end{bmatrix}^{\mathrm{T}}$, $\boldsymbol{d}_i = -s_{\mathrm{r}i} d_{\mathrm{W}i}$, $\boldsymbol{A}_{ii} = 1$, $\boldsymbol{B}_{ii} = s_{\mathrm{r}i} \begin{bmatrix} 1 & 1 \end{bmatrix}$ and $\boldsymbol{B}_{ii-1} = s_{\mathrm{r}i} \begin{bmatrix} 0 & -1 \end{bmatrix}$. It is assumed that $\boldsymbol{d}_i$ is periodic with period $P \in \mathbb{N}$, i.e., $\boldsymbol{d}_{i,t} = \boldsymbol{d}_{i,t+P}$ for all $t \in \mathbb{N}$. For a water system this assumption approximately holds with a period of one day as the water demands $d_{\mathrm{W}i}$ do not change significantly from day to day. Moreover, the subsystems are coupled by the head–flow constraint (12), i.e., $\mathcal{X}_i^{\mathrm{C}} = \mathcal{P}_{ii+1}$.

The goal of this section is to design a DMPC for the dynamics (14) where each subsystem is controlled by a local controller trying to minimize an economical objective cooperatively while satisfying the constraints. Therefore, the controllers use augmented models and an iterative algorithm that is tailored to the cascaded structure and the coupling constraints. To keep the communication effort low, the controllers exchange predicted state and input sequences in an event-triggered fashion.

Noteworthy, the following derivations are done in a general way so that the proposed method can also be applied to other systems with this structure, e.g., irrigation canals as considered in Negenborn, van Overloop, Keviczky, and De Schutter (2009) or supply chains in Maestre, De La Pena, Camacho, and Alamo (2011)

4.1. Controller models

Each controller $\forall i \in \mathcal{I}_{1:M}$ uses the following augmented model comprising the dynamics of its and the successive subsystem which are both directly influenced by $\boldsymbol{u}_i$:

$$\boldsymbol{x}_{\mathrm{a}i,t+1} = \boldsymbol{A}_{\mathrm{a}i}\boldsymbol{x}_{\mathrm{a}i,t} + \boldsymbol{B}_{\mathrm{a}i}\boldsymbol{u}_{i,t} + \boldsymbol{w}_{i,t} + \boldsymbol{d}_{\mathrm{a}i,t}. \tag{15}$$

The augmented state and exogenous input are $\boldsymbol{x}_{\mathrm{a}i} = \begin{bmatrix}\boldsymbol{x}_i^{\mathrm{T}} & \boldsymbol{x}_{i+1}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ and $\boldsymbol{d}_{\mathrm{a}i} = \begin{bmatrix}\boldsymbol{d}_i^{\mathrm{T}} & \boldsymbol{d}_{i+1}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$, respectively, the augmented system and input matrices are $\boldsymbol{A}_{\mathrm{a}i} = \mathrm{diag}\left(\boldsymbol{A}_{ii}, \boldsymbol{A}_{i+1i+1}\right)$, $\boldsymbol{B}_{\mathrm{a}i} = \begin{bmatrix}\boldsymbol{B}_{ii}^{\mathrm{T}} & \boldsymbol{B}_{i+1i}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ for all $i \in \mathcal{I}_{1:M-1}$. For subsystem M $\boldsymbol{x}_{\mathrm{a}M} = \boldsymbol{x}_M$, $\boldsymbol{d}_{\mathrm{a}M} = \boldsymbol{d}_M$, $\boldsymbol{A}_{\mathrm{a}M} = \boldsymbol{A}_{MM}$, $\boldsymbol{B}_{\mathrm{a}M} = \boldsymbol{B}_{MM}$ holds. The couplings to the predecessor and successor systems are

$$\boldsymbol{w}_i = \begin{cases}\boldsymbol{B}_{\mathrm{a}1}^{+}\boldsymbol{u}_2, & i = 1\\ \boldsymbol{B}_{MM-1}\boldsymbol{u}_{M-1}, & i = M\\ \boldsymbol{B}_{\mathrm{a}i}^{-}\boldsymbol{u}_{i-1} + \boldsymbol{B}_{\mathrm{a}i}^{+}\boldsymbol{u}_{i+1}, & \forall i \in \mathcal{I}_{2:M-1}\end{cases} \tag{16}$$

with $\boldsymbol{B}_{\mathrm{a}i}^{+} = \begin{bmatrix}\boldsymbol{0}^{\mathrm{T}} & \boldsymbol{B}_{i+1i+1}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$ and $\boldsymbol{B}_{\mathrm{a}i}^{-} = \begin{bmatrix}\boldsymbol{B}_{ii-1}^{\mathrm{T}} & \boldsymbol{0}^{\mathrm{T}}\end{bmatrix}^{\mathrm{T}}$.

The motivation for choosing the augmented models is to design cooperative controllers which consider the influence of their decision on other subsystems. In the case of the WDN, the state of the successor zone is included since zone i can pump water from subsystem $i+1$ and thus directly influences the water level in zone $i+1$.

The constraints for the augmented controller model are

$$\boldsymbol{x}_{\mathrm{a}i} \in \mathcal{X}_{\mathrm{a}i} = \mathcal{X}_i \times \mathcal{X}_{i+1}, \quad \left(\boldsymbol{x}_{\mathrm{a}i}, \boldsymbol{u}_i\right) \in \mathcal{X}_i^{\mathrm{C}},$$

$\forall i \in \mathcal{I}_{1:M-1}$ and $\mathcal{X}_{\mathrm{a}M} = \mathcal{X}_M$, $\mathcal{X}_M^{\mathrm{C}} = \emptyset$ for subsystem M as well as $\boldsymbol{u}_i \in \mathcal{U}_i$ $\forall i \in \mathcal{I}_{1:M}$. The sets $\mathcal{X}_i^{\mathrm{C}}$, $\mathcal{X}_{\mathrm{a}i}$ and $\mathcal{U}_i$ are combined in $\left(\boldsymbol{x}_{\mathrm{a}i}, \boldsymbol{u}_i\right) \in \mathcal{Z}_{\mathrm{a}i}$ for all $i \in \mathcal{I}_{1:M}$ for further discussions.

Moreover, the global model is defined as

$$\boldsymbol{x}_{t+1} = \boldsymbol{A}\boldsymbol{x}_t + \boldsymbol{B}\boldsymbol{u}_t + \boldsymbol{d}_t$$

where $\boldsymbol{x} = \mathrm{col}_{i\in\mathcal{I}_{1:M}}\left(\boldsymbol{x}_i\right)$, $\boldsymbol{u} = \mathrm{col}_{i\in\mathcal{I}_{1:M}}\left(\boldsymbol{u}_i\right)$, $\boldsymbol{d} = \mathrm{col}_{i\in\mathcal{I}_{1:M}}\left(\boldsymbol{d}_i\right)$, the global system matrix $\boldsymbol{A} = \mathrm{diag}_{i\in\mathcal{I}_{1:M}}\left(\boldsymbol{A}_{ii}\right)$ and the input matrix $\boldsymbol{B} \in \mathbb{R}^{n\times m}$ consists of the blocks $\boldsymbol{B}_{ij}$. The global system is subject to the constraint $(\boldsymbol{x}, \boldsymbol{u}) \in \mathcal{Z}$ which is composed of the constraints $\left(\boldsymbol{x}_{\mathrm{a}i}, \boldsymbol{u}_i\right) \in \mathcal{Z}_{\mathrm{a}i}$ for $i \in \mathcal{I}_{1:M}$.

4.2. Objective function and optimal trajectory

The cost of every subsystem i is $V_{N,i,t}\left(\boldsymbol{X}_{i,t}, \boldsymbol{U}_{i,t}\right) = \sum_{r=t}^{t+N-1} l_{i,r}\left(\boldsymbol{x}_{i,r}, \boldsymbol{u}_{i,r}\right)$ where $l_i : \mathbb{R}^{n_i} \times \mathbb{R}^{m_i} \to \mathbb{R}$ are the local stage costs. They are assumed to be periodic with period P, i.e. $l_{i,t} = l_{i,t+P}$ for all $t \in \mathbb{N}$. N is the prediction horizon. The global stage cost is $l_r(\boldsymbol{x}, \boldsymbol{u}) = \sum_{i=1}^{M} l_{i,r}\left(\boldsymbol{x}_i, \boldsymbol{u}_i\right)$. More information is given in Section 5.

The controller makes use of an optimal periodic trajectory which can be obtained by solving the optimization problem

$$\underset{\bar{\boldsymbol{X}},\bar{\boldsymbol{U}}}{\text{minimize}} \quad \sum_{r=0}^{P-1} l_r\left(\bar{\boldsymbol{x}}_r, \bar{\boldsymbol{u}}_r\right) \tag{17a}$$

$$\text{subject to} \quad \bar{\boldsymbol{x}}_P = \bar{\boldsymbol{x}}_0 \tag{17b}$$

$$\text{for all } r \in \mathcal{I}_{0:P-1}:$$

$$\bar{\boldsymbol{x}}_{r+1} = \boldsymbol{A}\bar{\boldsymbol{x}}_r + \boldsymbol{B}\bar{\boldsymbol{u}}_r + \boldsymbol{d}_r \tag{17c}$$

$$(\bar{\boldsymbol{x}}_r, \bar{\boldsymbol{u}}_r) \in \mathcal{Z}. \tag{17d}$$

The optimal sequences obtained from the solution of the optimization problem are denoted $\bar{\boldsymbol{X}}^* = [\bar{\boldsymbol{x}}_0^{*,\mathrm{T}}, \dots, \bar{\boldsymbol{x}}_{P-1}^{*,\mathrm{T}}]^{\mathrm{T}}$ and $\bar{\boldsymbol{U}}^* = [\bar{\boldsymbol{u}}_0^{*,\mathrm{T}}, \dots, \bar{\boldsymbol{u}}_{P-1}^{*,\mathrm{T}}]^{\mathrm{T}}$. Noteworthy, the initial state is free in the optimization problem and the sequences $\bar{\boldsymbol{X}}^*$ and $\bar{\boldsymbol{U}}^*$ are periodic due to the constraint (17b). The sequences are determined offline and are not necessarily unique.

4.3. Algorithm

At the core of the DMPC algorithm, the following optimization problem is solved in every subsystem $i \in \mathcal{I}_{1:M}$:

$$\underset{\boldsymbol{Z}_{i,t}}{\text{minimize}} \quad V_{\mathrm{a}i,t}\left(\boldsymbol{X}_{\mathrm{a}i,t}, \boldsymbol{U}_{\mathrm{a}i,t}\right) \tag{18a}$$

$$\text{subject to} \quad \boldsymbol{x}_{\mathrm{a}i,t} = \boldsymbol{x}_{\mathrm{a}i,t}^{\mathrm{m}} \tag{18b}$$

$$\boldsymbol{x}_{\mathrm{a}i,t+N} = \bar{\boldsymbol{x}}_{\mathrm{a}i,[t]_N}^{*} \tag{18c}$$

$$\text{for all } r \in \mathcal{I}_{t:t+N-1}:$$

$$\boldsymbol{x}_{\mathrm{a}i,r+1} = \boldsymbol{A}_{\mathrm{a}i}\boldsymbol{x}_{\mathrm{a}i,r} + \boldsymbol{B}_{\mathrm{a}i}\boldsymbol{u}_{i,r} + \boldsymbol{w}_{i,r} + \boldsymbol{d}_{\mathrm{a}i,r} \tag{18d}$$

$$\boldsymbol{w}_{i,r} = \begin{cases}\boldsymbol{B}_{\mathrm{a}1}^{+}\boldsymbol{u}_{2,r}, & i=1\\ \boldsymbol{B}_{MM-1}\boldsymbol{u}_{M-1,r}, & i=M\\ \boldsymbol{B}_{\mathrm{a}i}^{-}\boldsymbol{u}_{i-1,r} + \boldsymbol{B}_{\mathrm{a}i}^{+}\boldsymbol{u}_{i+1,r}, & \text{else}\end{cases} \tag{18e}$$

$$\text{for all } j \in \mathcal{I}_{\max(i-1,1):\min(i+1,M)} \setminus \{i\}:$$

$$\boldsymbol{u}_{j,r} = \hat{\boldsymbol{u}}_{j,r}^{k} \tag{18f}$$

$$\text{for all } j \in \mathcal{I}_{\max(i-1,1):\min(i+1,M)}:$$

$$\left(\boldsymbol{x}_{\mathrm{a}j,r}, \boldsymbol{u}_{j,r}\right) \in \mathcal{Z}_{\mathrm{a}j} \tag{18g}$$

$$\text{for all } j \in \mathcal{I}_{\max(i-1,1):\min(i+2,M)} \setminus \{i, i+1\}:$$

$$\boldsymbol{x}_{j,r} = \hat{\boldsymbol{x}}_{j,r}^{k}. \tag{18h}$$

Starting from the measurement $\boldsymbol{x}_{\mathrm{a}i,t}^{\mathrm{m}}$ in (18b), each subsystem predicts the future augmented states using the system model (15) in (18d). The constraint (18c) ensures that the terminal state lies on the periodic optimal trajectory in order to ensure recursive feasibility of the control law. The prediction horizon N is chosen identical to the period P for reasons of simplicity. The coupling to other subsystems is included in $\boldsymbol{w}_{i,r}$ in (18e). The inputs from the neighbor subsystems are taken from the last iteration in constraint (18f) where $\hat{\boldsymbol{u}}$ refer to inputs from successor and predecessor subsystems of subsystem i. Moreover, the subsystem has to consider the constraints which include water levels it influences directly. This is done in constraint (18g). The water levels from subsystems involved in these constraints are included into the optimization in (18h).

The cost that is minimized by the controller is

$$V_{\mathrm{a}i,t}\left(\boldsymbol{X}_{\mathrm{a}i}, \boldsymbol{U}_{\mathrm{a}i}\right) = \sum_{j\in\{i,i+1\}} V_{N,j,t}\left(\boldsymbol{X}_j, \boldsymbol{U}_j\right) \tag{19}$$

for all $i \in \mathcal{I}_{1:M-1}$ and $V_{\mathrm{a}M,t}\left(\boldsymbol{X}_M, \boldsymbol{U}_M\right) = V_{N,M,t}\left(\boldsymbol{X}_M, \boldsymbol{U}_M\right)$ for subsystem M. $\boldsymbol{X}_{\mathrm{a}i}$ and $\boldsymbol{U}_{\mathrm{a}i}$ are the predicted augmented state and input sequences. $\boldsymbol{X}_i$ and $\boldsymbol{U}_i$ are the local predicted state and input sequences. $\boldsymbol{Z}_i$ is the optimization variable which comprises all state and input sequences in the optimization problem. The cost includes the local cost as well as the cost of subsystem $i+1$ which is directly influenced by $\boldsymbol{U}_i$. Thus, the controllers work cooperatively.

Next, the steps of the iterative DMPC are discussed in Algorithm 1. $k \in \mathcal{I}_{1:k_{\max}}$ is the iteration counter and $k_{\max}$ is the maximum number of iterations. At the beginning each controller solves the DMPC problem (18) locally having state and input sequences $\hat{\boldsymbol{X}}$ and $\hat{\boldsymbol{U}}$ for the required subsystems from the last iteration or initialization. Then, the difference ΔV_i of the optimal cost to one of the previous iteration is computed.

Note that for subsystem M, $\hat{\boldsymbol{U}}^k_{i+1,t} = \mathbf{0}$ for all $k \in \mathcal{I}_{1:k_{\max}}$ and $t \in \mathbb{N}$. If the difference is smaller than some predefined threshold $\gamma_i \in \mathbb{R}_{<0}$, an event is generated. This implies that the binary trigger variable δ_i is set to 1, otherwise it is 0. From the systems which triggered an event, the system with the biggest cost decrease is determined in step 4 and called system i^*. For this either global communication is require or it is determined iteratively by neighbor-to-neighbor communication. $\boldsymbol{U}_{i^*}$ is then sent to the neighboring subsystems, i.e., to the subsystems i^*+1 and i^*-1. Furthermore, the state sequence $\boldsymbol{X}^{k+1}_{i^*,t}$ is sent to subsystems which require it in constraint (18g), i.e., to the subsystems $j \in \mathcal{I}_{\max(1,i^*-2):\min(i^*+1,M)}$ in step 5. Note that the communication of input and state sequences is only required when an event is triggered. Thereby, the communication effort is kept low. The solutions then are included in the corresponding systems and the solutions of the remaining systems are discarded in step 6. The procedure is repeated until the maximum number of iterations is reached or no event is triggered, i.e., $\mathcal{D} = \emptyset$ (step 7).

Algorithm 1 DMPC algorithm

1: **Measure** $\boldsymbol{x}^{\mathrm{m}}_{i,t}$ and send to neighbor subsystems
2: **for** $k = 1, \dots, k_{\max}$
3: **for** $i = 1, \dots, M$ in parallel

Solve (18) to obtain
$\boldsymbol{Z}^{k+1}_{i,t} = \mathrm{argmin}_{\boldsymbol{Z}_{i,t}} V_{\mathrm{a}i,t}\left(\boldsymbol{X}_{\mathrm{a}i,t}, [\boldsymbol{U}_{i,t}, \hat{\boldsymbol{U}}^k_{i+1,t}]\right)$
Extract $\boldsymbol{X}^{k+1}_{\mathrm{a}i,t}$, $\boldsymbol{X}^{k+1}_{i,t}$ and $\boldsymbol{U}^{k+1}_{i,t}$ from $\boldsymbol{Z}^{k+1}_{i,t}$
Compute $\Delta V_i = V_{\mathrm{a}i,t}\left(\boldsymbol{X}^{k+1}_{\mathrm{a}i,t}, [\boldsymbol{U}^{k+1}_{i,t}, \hat{\boldsymbol{U}}^k_{i+1,t}]\right)$
$-V_{\mathrm{a}i,t}\left(\boldsymbol{X}^k_{\mathrm{a}i,t}, [\hat{\boldsymbol{U}}^k_{i,t}, \hat{\boldsymbol{U}}^k_{i+1,t}]\right)$
Evaluate event-trigger:
if $\Delta V_i < \gamma_i$ **then** $\delta_i = 1$ **else** $\delta_i = 0$

endfor
4: **Find**: $i^* = \mathrm{argmin}_{i \in \mathcal{D}} \Delta V_i$ with $\mathcal{D} = \left\{ i \in \mathcal{I}_{1:M} | \delta_i = 1 \right\}$
5: **Send** state and input sequence from subsystem i^*
6: **for** $i = 1, \dots, M$ in parallel

if $i = i^*$ **then** $\hat{\boldsymbol{U}}^{k+1}_{i^*,t} = \boldsymbol{U}^{k+1}_{i^*,t}$, $\hat{\boldsymbol{X}}^{k+1}_{i^*,t} = \boldsymbol{X}^{k+1}_{i^*,t}$
else set $\hat{\boldsymbol{U}}^{k+1}_{i,t} = \hat{\boldsymbol{U}}^k_{i,t}$, $\hat{\boldsymbol{X}}^{k+1}_{i,t} = \boldsymbol{X}^k_{i,t}$

endfor
7: **if** $\mathcal{D} = \emptyset$ **then** go to step 8
endfor
8: **for** $i = 1, \dots, M$ in parallel
Set $\hat{\boldsymbol{U}}^1_{i,t+1} = \left[\hat{\boldsymbol{u}}^{k+1,\mathrm{T}}_{i,t+1}, \dots, \hat{\boldsymbol{u}}^{k+1,\mathrm{T}}_{i,t+N-1}, \bar{\boldsymbol{u}}^{*,\mathrm{T}}_{i,[t]_N}\right]^{\mathrm{T}}$,
Set $\hat{\boldsymbol{X}}^1_{i,t+1} = \left[\hat{\boldsymbol{x}}^{k+1,\mathrm{T}}_{i,t+1}, \dots, \hat{\boldsymbol{x}}^{k+1,\mathrm{T}}_{i,t+N}, \bar{\boldsymbol{x}}^{*,\mathrm{T}}_{i,[t]_N+1}\right]^{\mathrm{T}}$
endfor
9: **Set** $\boldsymbol{u}_t = \mathrm{col}_{i \in \mathcal{I}_{1:M}}\left([\boldsymbol{I}\ \mathbf{0}\ \dots\ \mathbf{0}]\, \hat{\boldsymbol{U}}^{k+1}_{i,t}\right)$

After that, the new initial state and input sequences for the next time step are generated by shifting the current state and input sequence and extending it with the corresponding state or input of the optimal sequences in step 8. This step can be executed without communication if the optimal trajectory is known in all subsystems. In the last step the first input is applied to the system.

Note that in each iteration the algorithm has feasible iterates and the costs are non-increasing over the iterations. A feasible solution for the next time step is ensured due to the shifting and extension using the optimal trajectory.

For time $t = 0$ an initially feasible sequence is required. The procedure for finding feasible initial sequences is application dependent. For the considered WDN it is described in Section 5. The set of states where a feasible input sequence can be found is called $\mathcal{X}_{N,t}$. Next the theoretical properties are discussed.

Theorem 4.1. *Consider system (14) and assume that $\boldsymbol{x}_0 \in \mathcal{X}_{N,0}$ as well as that there are no disturbances or modeling errors, then the control law obtained from Algorithm 1 is recursively feasible for all $t \in \mathbb{N}$ and the average performance of the closed-loop fulfills*

$$\limsup_{T \to +\infty} \frac{\sum_{t=0}^{T} l_t\left(\boldsymbol{x}_t, \boldsymbol{u}_t\right)}{T+1} \le \frac{\sum_{t=0}^{P-1} l_t\left(\bar{\boldsymbol{x}}^*_t, \bar{\boldsymbol{u}}^*_t\right)}{P}. \tag{20}$$

The proof can be found in the Appendix. Eq. (20) says that the asymptotic average performance of the closed-loop system is better than the performance of the optimal periodic trajectories determined by (17).

Remark 4.2. In the proof of Theorem 4.1 it is assumed that the measurements at time t match the predicted value for $t+1$ at time t, i.e., $\boldsymbol{x}^{\mathrm{m}}_{i,t+1} = \boldsymbol{x}^{k+1}_{i,t+1}$. Due to non-predicted disturbances or modeling errors, this might not hold in a practical application and thus a robust controller must be designed for guaranteeing recursive feasibility. However, the design of a robust MPC controller for the considered set-up is a challenging task as it requires the computation of (robustly) positive invariant sets. Due to the integer nature of the problem this is not straightforward. For more information on robust MPC refer to (Rawlings & Mayne, 2009, chapter 3).

Remark 4.3. Note that compared to a centralized controller, a distributed one is usually more robust to failures in the system. If the centralized controller fails, the overall plant cannot be controlled anymore. For the considered DMPC, however, it might in some cases be possible to control at least some of the subsystems in case of a failure in one of the controllers. In which cases this is possible dependents on the application. An example for the considered set-up is discussed in Section 5. For failure situations the optimal periodic trajectory changes. They can be computed offline before operation and stored for the online usage.

5. Simulation

5.1. Simulation set-up

The proposed DMPC scheme is tested in a simulation of a model of the WDN of Kaiserslautern where the complete system is aggregated in $M = 3$ pressure zones/subsystems. The first zone has a water storage with water level h_1, an aggregated demand $d_{\mathrm{W}1}$ and an supply flow $q_{\mathrm{s}1}$ where water is pumped from deep wells. Moreover, it is connected to the second zone from which water can be pumped to zone one over a pump station with $n^{\mathrm{p}}_{21} = 2$ identical fixed-speed pumps running at $|\mathcal{W}_{21}| = 1$ speed. The pumps are modeled as described in Sections 2 and 3. The number of configurations is $n^{\mathrm{con}}_{21} = 3$, as either no, one, or both pumps are running. Note that since both pumps are identical it does not matter to the optimization which is the one that is used. Zone two has a water storage with water level h_2, an aggregated demand $d_{\mathrm{W}2}$. It takes the complete supply from the third system, i.e., there are no deep wells in this system which means that $q_{\mathrm{s}2,t} = 0$ for all $t \in \mathbb{N}$. The third zone has a water reservoir with level h_3 and is a pure supply zone with supply flow $q_{\mathrm{s}3}$. Furthermore, it is supplied by a spring which is considered as a negative demand in $d_{\mathrm{W}3}$ as there are no costs for pumping the water to the surface. From zone three one can pump water to zone two over a pump station with $n^{\mathrm{p}}_{32} = 1$ pump which can operate at $|\mathcal{W}_{32}| = 2$ different fixed speeds. The number of pump configurations is $n^{\mathrm{con}}_{32} = 3$ as the pump is either switched off or is running with one of the speeds. The detailed system data cannot be provided due to a nondisclosure agreement with the WDN operator.

Interestingly, the first zone could be operated independently as the deep wells could cover the demand in this zone. However, as the first system has a higher elevation the cost for pumping the water in the deep wells is higher than in zone three why it might be beneficial to also take water supplied by zone three.

In a first step, a detailed nonlinear model of the plant has been derived following the methodology of Section 2 and Brdys and Ulanicki

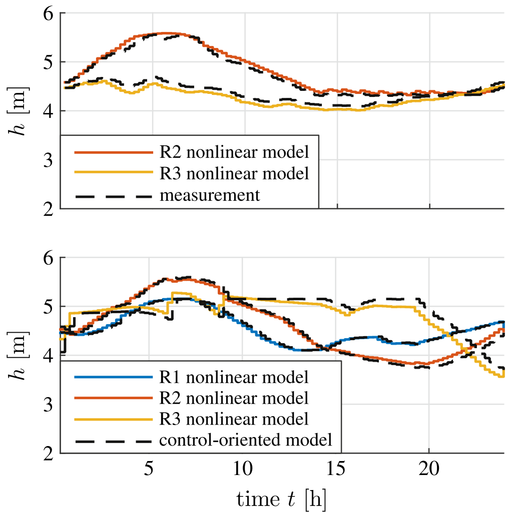

Fig. 3. The upper figure shows the comparison of water levels in reservoirs from measurements and simulation using with the nonlinear model. The lower figure shows the comparison of the water levels in the reservoirs using the nonlinear and the control-oriented model.

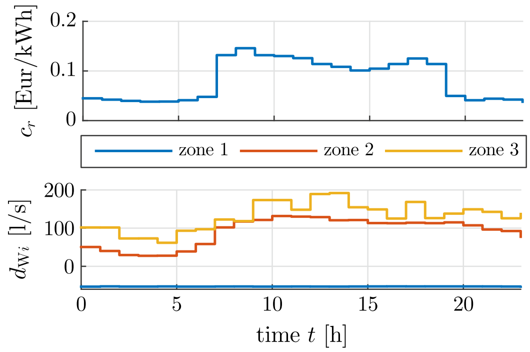

Fig. 4. Price for electrical power c_r and water demands d_{Wi} over one day.

Table 1
Simulation data.

Explanation	Name	Value
Sample time	T_s	1 h
Prediction horizon	N	24
max. iterations DMPC	k_{max}	10

(1994, chapter 2). All required parameters have been taken from data sheets or identified from measurement data. The upper part of Fig. 3 depicts the water levels of reservoir two and three obtained from measurements and from the derived nonlinear model. Thereby, the power-on times and the speed of the pumps as well as the water demand has been taken from measurements and applied to the nonlinear model. One can see that the nonlinear model approximates the behavior of the plant accurately. For reservoir one, no measurements are available.

In a second step, the nonlinear model has been simplified to the control-oriented model described in Section 3. The lower part of Fig. 3 shows the water levels in the reservoirs when applying the inputs obtained from an optimization problem subject to the control-oriented model on the detailed nonlinear model in an open-loop fashion. The small deviation shows that the control-oriented model is a good approximation for the nonlinear model and thus for the behavior of the plant.

Next, the cost function is discussed. The electricity cost is determined by multiplying the time-varying price c_r for energy with the electrical power consumed by the pump over the sampling time. The electricity price and water demand in each zone $d_{\mathrm{W}i}$ are shown in Fig. 4 for one day. Both are assumed to be periodic with period P, i.e., $c_t = c_{t+P}$ and $d_{\mathrm{W}i,t} = d_{\mathrm{W}i,t+P}$ for all $i \in \mathcal{I}_{1:M}$ and $t \in \mathbb{N}$. For the pumps connecting pressure zones, the power is approximated according to (13).

The flow q_{si} aggregates the supply flows of multiple pumps in zone i pumping water from deep wells. A lower-level controller schedules the pumps in the deep wells to provide the required flow q_{si}. It has to consider legal and technical limitations but is not further discussed here. The power consumption dependent on the flow q_{si} has been determined from measurement data and can be approximated by

$$P_{si}(q_{si}) = a_{2i} q_{si}^2 + a_{1i} q_{si} + a_{0i}. \tag{21}$$

Thus, the total cost for electricity can be recast in the stage cost as

$$l_{i,r}^{\mathrm{u}}\left(\boldsymbol{u}_{i,r}\right) = \boldsymbol{u}_{i,r}^{\mathrm{T}} \boldsymbol{R}_{i,r} \boldsymbol{u}_{i,r} + \boldsymbol{r}_{i,r}^{\mathrm{T}} \boldsymbol{u}_{i,r} + s_{i,r} \tag{22}$$

where $\boldsymbol{R}_{i,r}$, $\boldsymbol{r}_{i,r}$, $s_{i,r}$ are weights combining the electricity price c_r and the pump power characteristics (13) and (21). They are periodic with period P (due to the electricity price), i.e., $\boldsymbol{R}_{i,t} = \boldsymbol{R}_{i,t+P}$, $\boldsymbol{r}_{i,t} = \boldsymbol{r}_{i,t+P}$ and $s_{i,t} = s_{i,t+P}$ for all $t \in \mathbb{N}$. The total stage cost is given by

$$l_{i,r}\left(\boldsymbol{x}_{i,r}, \boldsymbol{u}_{i,r}\right) = l_{i,r}^{\mathrm{u}}\left(\boldsymbol{u}_{i,r}\right) + l_{i,r}^{\mathrm{x}}\left(\boldsymbol{x}_{i,r}\right) \tag{23}$$

$$l_{i,r}^{\mathrm{x}}\left(\boldsymbol{x}_{i,r}\right) = \left(\boldsymbol{x}_{i,r} - \boldsymbol{x}_{i,\mathrm{ref}}\right)^{\mathrm{T}} \boldsymbol{Q}_{i,r} \left(\boldsymbol{x}_{i,r} - \boldsymbol{x}_{i,\mathrm{ref}}\right) \tag{24}$$

where $\boldsymbol{x}_{\mathrm{ref},i}$ is an artificial reference for the state and the cost $l_{i,r}^{\mathrm{x}}\left(\boldsymbol{x}_{i,r}\right)$ penalizes the deviation of the actual state from it. The term can be used so that the water level in the reservoir does not become too low due to safety requirements in cases of blackouts in the electrical power system or fire. $\boldsymbol{Q}_{i,r}$ are positive definite weighting matrices.

Together with the cost (23), optimization problem (18) can be recast as an MIQP for which efficient solvers such as CPLEX or Gurobi exist. The integer variables stem from the switching of the pumps, i.e., correspond to the possible configurations in the pump stations.

As mentioned before, for the initialization of the distributed algorithm at time $t = 0$, initially feasible input trajectories must be determined. To prevent a centralized initialization, the WDN is divided into self-sustaining areas. For this purpose, the pump station between subsystem one and two is shut off. Thus, subsystem one solves a decentralized optimization problem where the connection between the first and the second subsystem is set to zero, i.e. with $q_{21} = 0$. Additionally, subsystem two and three solve jointly an optimization problem with $q_{21} = 0$. The obtained input trajectories are used to start the algorithm. This approach can also be applied for larger systems.

As pointed out in Remark 4.3, in case of a failure of one of the controllers in the subsystems, it might be possible to keep up the control of the other subsystems. In the considered WDN, in case of a failure in zone 1, one can still control zone 2 and 3 and vice versa as those zones can be operated independently. Similar to the reasons in the initialization approach, this can also be applied to larger systems.

The simulation environment is MATLAB 2014b and the solver CPLEX 12.6 which uses a branch and cut algorithm to solve the MIQPs. All computations are carried out using an Intel Core i5-660 processor with 3.33 GHz and 8 GB RAM under Windows 7 Professional. Some controller data can be found in Table 1.

5.2. Simulation results

The simulation results of the DMPC are compared to an omniscient centralized MPC (CMPC). The controller data for centralized and distributed control are identical. The state and input trajectory are plotted over four periods in Fig. 5. In the simulation the optimization problem is feasible at all time (recursively feasibility) and it can be observed that the constraints are respected and the trajectories of both controllers converge to the optimal trajectory computed by solving problem (17) offline. The controllers exploit the storage capacity of the reservoirs by filling them in the morning time when the energy price is low and emptying them when the price is high (cf. Fig. 4). Furthermore, water

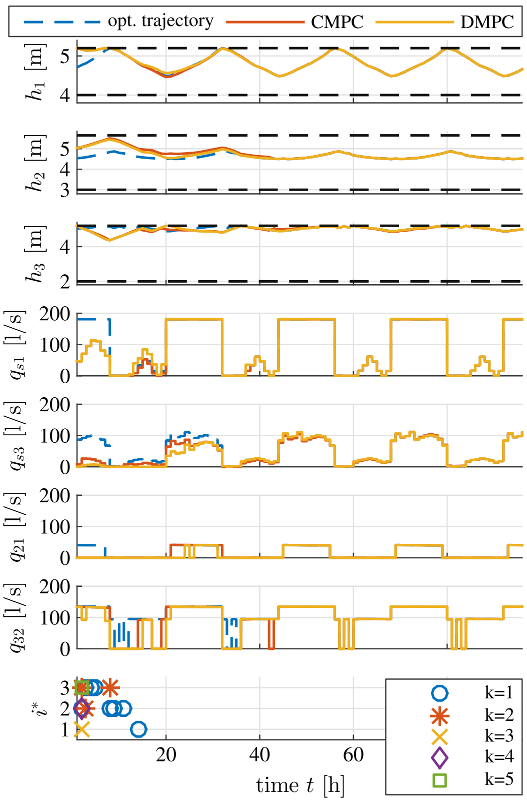

Fig. 5. Simulation results over four periods.

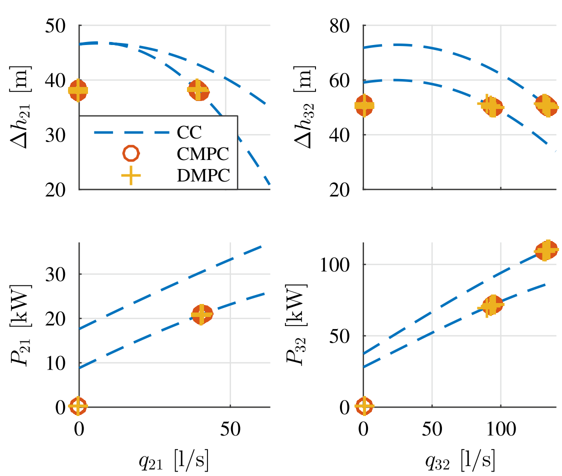

Fig. 6. Pump characteristic curves and operating points in simulation.

is pumped from zone three over zone two to zone one as the cost for the water supply in zone three is lower as in zone one due to the lower elevation.

Considering Fig. 6 one can see that the pumps are operated close to their flow–head characteristic curve (CC) and the approximation is valid. The power (6) is also approximated satisfactorily by (13). Moreover, the pump connecting subsystems two and three is operated in both fixed speeds.

The average times required to solve the optimization problems are $t_{\text{DMPC}} = 0.2753$ s and $t_{\text{CMPC}} = 0.4003$ s. This shows that the controller is real-time applicable and that for the considered scenario it is in average faster to solve several smaller problems in parallel than one large centralized one. The closed-loop costs $J = \sum_{t=0}^{T_{\text{sim}}} l_t\left(\boldsymbol{x}_t, \boldsymbol{u}_t\right)$ are $J_{\text{CMPC}} = 955$ and $J_{\text{DMPC}} = 963$ showing that the DMPC is performing slightly worse due to the distributed decision making and the incomplete knowledge in the subsystems. In Fig. 5 one can also see which subsystem updates its input trajectory in which iteration. While there are iterations at the beginning, there are no more iterations required when the DMPC trajectories approach the optimal trajectories.

6. Conclusions

Motivated by the WDN of Kaiserslautern, in the paper, first a control-oriented model for WDNs with cascaded multiple pressure zones and fixed-speed pumps connecting them is presented. The characteristic curves of the pumps are approximated by an union of polytopes which can easily be integrated into the controller optimization problems which are solved with standard software. A recursively feasible DMPC algorithm was designed where there is a focus on the reduction of the communication effort using event-triggered communication. For the considered simulation scenario the state and input trajectories converge to the optimal periodic trajectory determined offline.

Acknowledgment

The research work presented here has been supported by the German Ministry for Economic Affairs and Energy (funding no. 03ET1278C).

Appendix. Proof of Theorem 4.1

First, recursive feasibility is considered. It is shown that from feasible sequences $\hat{\boldsymbol{U}}_{i,t}^1$ and $\hat{\boldsymbol{X}}_{i,t}^1$ $\forall i \in \mathcal{I}_{1:M}$ at time t, the sequences $\hat{\boldsymbol{U}}_{i,t}^k$ $\forall i \in \mathcal{I}_{1:M}$ remain feasible over the iterations $k \in \mathcal{I}_{1:k_{\max}}$. Then, it is shown how the sequences $\hat{\boldsymbol{U}}_{i,t+1}^1$ and $\hat{\boldsymbol{X}}_{i,t+1}^1$ can be constructed for the time $t+1$.

As by assumption $\hat{\boldsymbol{U}}_{i,t}^1$ $\forall i \in \mathcal{I}_{1:M}$ are feasible solutions to the optimization problem (18), there exists at least one solution such that it is feasible and thus the sequences $\boldsymbol{U}_{i^*,t}^2$, $\hat{\boldsymbol{U}}_{j,t}^1$ and $\boldsymbol{X}_{i^*,t}^2$, $\hat{\boldsymbol{X}}_{j,t}^1$ $j \in \mathcal{I}_{1:M} \setminus \{i^*\}$ are feasible. Through the assigning step 6, $\hat{\boldsymbol{U}}_{i,t}^2$ and $\hat{\boldsymbol{X}}_{i,t}^2$ $\forall i \in \mathcal{I}_{1:M}$ are feasible. By induction it follows that $\hat{\boldsymbol{U}}_{i,t}^k$ and $\hat{\boldsymbol{X}}_{i,t}^k$ $\forall i \in \mathcal{I}_{1:M}$ are feasible for $k \in \mathcal{I}_{1:k_{\max}}$.

Next, it is shown that the sequences $\hat{\boldsymbol{U}}_{i,t+1}^1$, $\hat{\boldsymbol{X}}_{i,t+1}^1$ generated in step 8 are feasible for the optimization problem (18). Since $\hat{\boldsymbol{U}}_{i,t}^k$ and $\hat{\boldsymbol{X}}_{i,t}^k$ $\forall i \in \mathcal{I}_{1:M}$ are feasible as well as $\boldsymbol{x}_{i,t+1}^{\mathrm{m}} = \boldsymbol{x}_{i,t+1}^{k+1}$, $\boldsymbol{u}_{i,t+r}^{k+1}$ and $\boldsymbol{x}_{i,t+r}^{k+1}$ satisfy the constraints for $r \in \mathcal{I}_{1:N-1}$. As $\boldsymbol{x}_{i,t+N}^{k+1} = \boldsymbol{x}_{i,[t]_N}^*$, it follows from the optimal trajectory that $\boldsymbol{x}_{i,[t]_N}^*, \boldsymbol{u}_{i,[t]_N}^*$ and $\boldsymbol{x}_{i,[t]_N+1}^*$ satisfy the constraints. Thus, $\hat{\boldsymbol{U}}_{i,t+1}^1$ and $\hat{\boldsymbol{X}}_{i,t+1}^1$ are feasible solutions for (18) and recursive feasibility holds.

Now, the performance relation (20) is considered. Let $k_{\mathrm{b},t}$ be the iteration where the algorithm terminates for time t. Evaluating the cost $V_N\left(\bar{\boldsymbol{X}}, \bar{\boldsymbol{U}}\right)$, it follows that

$$V_{N,t+1}\left(\bar{\boldsymbol{X}}_{t+1}^1, \bar{\boldsymbol{U}}_{t+1}^1\right) - V_{N,t}\left(\bar{\boldsymbol{X}}_t^{k_{\mathrm{b},t}+1}, \bar{\boldsymbol{U}}_t^{k_{\mathrm{b},t}+1}\right)$$
$$\leq -l_t\left(\boldsymbol{x}_t, \boldsymbol{u}_t\right) + l_t\left(\bar{\boldsymbol{x}}_{[t]_N}^*, \bar{\boldsymbol{u}}_{[t]_N}^*\right).$$

By optimality it follows that $V_{N,t+1}\left(\bar{\boldsymbol{X}}_{t+1}^{k_{\mathrm{b},t}+1}, \bar{\boldsymbol{U}}_{t+1}^{k_{\mathrm{b},t}+1}\right) \leq V_{N,t+1}\left(\bar{\boldsymbol{X}}_{t+1}^1, \bar{\boldsymbol{U}}_{t+1}^1\right)$. Following the steps of the proof in Grosso et al. (2017, Theorem A.1), leads to (20). This completes the proof. □

References

Angeli, D., Amrit, R., & Rawlings, J. B. (2012). On average performance and stability of economic model predictive control. *IEEE Transactions on Automatic Control*, *57*(7), 1615–1626.

Berkel, F., & Liu, S. Non-iterative distributed model predictive control with event-triggered communication. In *Proceedings of the American control conference* (pp. 2344–2349), Milwaukee, USA.

Brdys, M., & Ulanicki, B. (1994). *Operational control of water systems: structures, algorithms, and applications*. Prentice Hall.

Caba, S., Lepper, M., & Liu, S. Nonlinear controller and observer design for centrifugal pumps. In *Proceedings of the IEEE conference on control applications* (pp. 569–574), Buenos Aires, Argentinia.

Christofides, P. D., Scattolini, R., Muñoz del la Peña, D., & Liu, J. (2013). Distributed model predictive control: A tutorial review and future research directions. *Computers and Chemical Engineering*, *51*, 21–41.

Gross, D., & Stursberg, O. (2016). A cooperative distributed MPC algorithm with event-basd communication and parallel optimization. *IEEE Transactions on Control of Network Systems*, *3*(3), 275–285.

Grosso, J. M., Ocampo-Martinez, C., & Puig, V. (2017). A distributed predictive control approach for periodic flow-based networks: application to drinking water systems. *International Journal of Systems Science*, *48*(14), 3106–3117.

Grosso, J., Ocampo-Martínez, C., Puig, V., Limon, D., & Pereira, M. (2014). Economic MPC for the management of drinking water networks. In *Proceedings of the European control conference* (pp. 790–795), Strasbourg, France.

Lee, J., & Angeli, D. (2014). Cooperative economic model predictive control for linear systems with convex objectives. *European Journal of Control*, *20*(3), 141–151.

Leirens, S., Zamora, C., Negenborn, R. R., & Schutter, B. D. Coordination in urban water supply networks using distributed model predictive control. In *Proceedings of the American control conference* (pp. 3957–3962), Baltimore, Maryland.

Limon, D., Pereira, M., De La Peña, D. M., Alamo, T., & Grosso, J. (2014). Single-layer economic model predictive control for periodic operation. *Journal of Process Control*, *24*(8), 1207–1224.

Maestre, J., De La Pena, D. M., Camacho, E., & Alamo, T. (2011). Distributed model predictive control based on agent negotiation. *Journal of Process Control*, *21*(5), 685–697.

Negenborn, R., & Maestre, J. (2014). Distributed model predictive control - an overview and roadmap of future research opportunities. *IEEE Control Systems Magazine*, *34*(4), 87–97.

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

Ocampo-Martinez, C., Barcelli, D., Puig, V., & Bemporad, A. (2012). Hierarchical and decentralised model predictive control of drinking water networks: Application to barcelona case study. *IET Control Theory & Applications*, *6*(1), 62–71.

Rawlings, J. B., Angeli, D., & Bates, C. N. (2012). Fundamentals of economic model predictive control. In *Proceedings of the 51st conference on decision and control* (pp. 3851–3861), Maui, USA.

Rawlings, J. B., & Mayne, D. Q. (2009). Model predictive control - theory and design. Nob Hill Publishing.

Scattolini, R. (2009). Architectures for distributed and hierarchical model predictive control - a review. *Journal of Process Control*, *19*(5), 723–731.

Stewart, B. T., Venkat, A. N., Rawlings, J. B., Wright, S. J., & Pannocchia, G. (2010). Cooperative distributed model predictive control. *Systems & Control Letters*, *59*(8), 460–469.

Zanon, M., Gros, S., & Diehl, M. A lyapunov function for periodic economic optimizing model predictive control. In *Proceedings of the 52nd annual conference ondecision and control* (pp. 5107–5112), Florence, Italy.