Journal of Process Control 84 (2019) 70–88

Contents lists available at ScienceDirect

Journal of Process Control

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

Enhancing cooperative distributed model predictive control for the water distribution networks pressure optimization

Yuan Zhang, Yi Zheng, Shaoyuan Li*

Department of Automation, Shanghai Jiao Tong University, and Key Laboratory of System Control and Information Processing, Ministry of Education of China, Shanghai, 200240, PR China

ARTICLE INFO

Article history:
Received 3 October 2018
Revised 28 September 2019
Accepted 29 September 2019
Available online 22 October 2019

Keywords:
Distributed model predictive control
Large-scale water distribution networks
Pressure optimization
Modeling and partitioning

ABSTRACT

In water distribution networks (WDNs), the supply pressure to users should be homogeneous and close to the lower bound so as to reduce leakages and energy consumption while meeting users' water demand. The large-scale features and incomplete structure information of modern WDNs increase the difficulty in modeling and pressure optimization. In this paper, an enhancing cooperative distributed model predictive control (EC-DMPC) strategy is proposed to solve this problem with enhanced operating performance. Firstly, an efficient modeling method and a partitioning method based on operation data are proposed to decompose the WDNs and construct the subsystem models. These methods are based on *k*-Shape clustering and canonical correlation analysis with considering the mechanism of WDNs. In order to increase the flexibility of coordination while guaranteeing the global optimization performance, a neighbourhood optimization coordination strategy is adopted, where each subsystem considers the performance of itself and that of strongly affected subsystems. Moreover, a robust invariant set and a robust invariant set control law are utilized to handle the disturbance caused by the ignored weak couplings. Part of Shanghai's WDNs is used as the study case to demonstrate the application of the proposed methods. A comparative experiment study is developed. Results have shown the efficiency of the proposed methods. By the coordination of the EC-DMPC strategy, when the water demand of users changes, the users' pressure can be maintained homogeneous, and the average pressure decreases, which is beneficial in the reduction of leakages and energy consumption. The distributed strategy could also improve the flexibility and scalability for large-scale WDNs.

1. Introduction

Water distribution networks (WDNs) are the systems that purify water and deliver clean water to the users through the cooperation of multiple components, such as pump station, valve, reservoir, pipe, etc. Therefore, WDNs are critical urban infrastructures and parts of foundations of the city's normal operation. In the operation of WDNs, the supply pressure to users is the key indicator as it affects the primary function of WDNs, which is providing enough water to users. For reasonably designed WDNs, users' demand can be guaranteed by providing supply pressure above a lower bound [1]. However, if the supply pressure is much higher than the lower bound, excessive pressure can lead to more leakages and energy consumption [2]. Thus, no matter how the water demand changes, the water supply pressure to users should be kept homogeneous and close to the lower bound.

Many studies have been conducted on the pressure optimization of WDNs. Dai et al. [3] propose a strategy where the pump station's model is included in the mechanistic model used in the pressure optimization problem. Then the considered optimization problem becomes a large-scale mixed-integer nonlinear programming problem. An approach which decomposes the original problem into smaller-scale problems is proposed, then the calculation efficiency is improved. Ulanicki et al. [4] use a leakage model in pressure control. An on-line control strategy of predictive and feedback control is proposed to schedule and control the pressure reducing valves within an area. In [3,4], only the steady-state model is considered, and the optimization is conducted with the change of water demand. A predictive zone control method is proposed by Liu and Li [5] to maintain the pressure of WDNs within a zone. With this method, the frequency of actuators' adjustment can be reduced when the water demand changes. Wang et al. [6–9] consider the management criterions as objective function while the pressure-flow model and the lower pressure bounds are taken

* Corresponding author.
E-mail address: syli@sjtu.edu.cn (S. Li).

https://doi.org/10.1016/j.jprocont.2019.09.009
0959-1524/

Nomenclature

$\bar{\boldsymbol{B}}_1$, $\bar{\boldsymbol{D}}_1$	coefficient matrices in nominal system model
$\bar{\boldsymbol{x}}$, $\bar{\boldsymbol{u}}$, $\bar{\boldsymbol{y}}$	variable vectors in nominal system model
$\boldsymbol{A}$, $\boldsymbol{B}_i$, $\boldsymbol{D}_i$	coefficient matrices in the original system model
$\boldsymbol{a}_i^*$, $\boldsymbol{b}_i^*$, $\boldsymbol{a}_i^O$, $\boldsymbol{b}_i^O$	coefficient vectors in CCA
$\boldsymbol{K}$, $\bar{\boldsymbol{K}}$	robust invariant set control law matrix and terminal invariant set control law matrix
$\boldsymbol{M}_i$	matrix composed by ordinal vectors
$\boldsymbol{u}_{min}$, $\boldsymbol{u}_{max}$	lower and upper bound of pump station's outlet pressure (KPa)
$\boldsymbol{W}_i$, γ_i	weighting matrices and iteration weighting term in subsystem i
$\boldsymbol{w}_{1,i}$, $\boldsymbol{w}_{2,i}$	influence of weakly coupling upstream neighbours of subsystem i
$\boldsymbol{x}_{min}$, $\boldsymbol{x}_{max}$, $\boldsymbol{x}_r$	lower, upper bound and reference value of reservoir's level (m)
$\boldsymbol{y}_{min}$	lower bound of users' pressure (KPa)
$\mathcal{N}_{i(strong/weak)}$, $\mathcal{H}_i$	sets of strong or weak coupling upstream neighbours and sets of strong coupling downstream neighbours of system i
$\mathcal{X}$, $\mathcal{U}$, $\mathcal{Y}$, $\mathcal{W}_{1,i}$, $\mathcal{W}_{2,i}$	constraint sets
ϕ, $\mathcal{X}_f$	robust invariant set and terminal invariant set
$C_{i,j}$, T_i	strong coupling between i and j and threshold used in partitioning method
d_i, $\boldsymbol{d}$	users' water demand of the ith water demand forecast area (m^3/h) (i.e. known disturbance) and $\boldsymbol{d} = [d_1, \ldots, d_{n_d}]^T$
k	number of subsystems
n_i	number of different variabels
p, N	number of iteration and optimization horizon
S_i	subsystem i
u_i, $\mathbb{U}$, $\boldsymbol{u}$	outlet pressure of the ith pump station (KPa) (i.e. input variable), set of input variables and $\boldsymbol{u} = [u_1, \ldots, u_{n_u}]^T$
v_i, $\mathbb{V}$	inlet pressure of the ith pump station (KPa) and the set of inlet pressure of pump station
$V_{iN,L}$, V_{iN}, V_N, V_f	local optmization function, neighbourhood optimization function of subsystem i, global optimization function and terminal cost function
x_i, $\mathbb{X}$, $\boldsymbol{x}$	level of the ith reservoir (m) (i.e. state variable), set of state variables and $\boldsymbol{x} = [x_1, \ldots, x_{n_x}]^T$.
y_i, $\mathbb{Y}$, $\boldsymbol{y}$	pressure of the i-th user node (KPa) (i.e. output variable), set of output variables and $\boldsymbol{y} = [y_1, \ldots, y_{n_y}]^T$

as constraints, where the lower pressure bounds can be satisfied; however, the excessive pressure is not guaranteed to be reduced.

Nowadays, with population growth and the resulting urban expansion, the size and complexity of WDNs are continually growing. Large-scale WDNs are fully interconnected through pipes. Centralized control is impractical and unsuitable for the control of large-scale systems like WDNs, which need high flexibility and high fault tolerance. Then some non-centralized strategies are proposed to handle the issues caused by the large-scale characteristics after a reasonable division of the WDNs. A distributed model predictive control (DMPC) strategy using a parallel coordination scheme is applied to the pressure and flow optimization problem of WDNs by Leirens et al. [10]. Taking the same objective as in [6–9], Fambrini et al. [11] propose a decentralized MPC scheme. Ocampo-Martinez et al. [12,13] propose a hierarchical and decentralized MPC scheme and a decentralized MPC scheme.

The decentralized and distributed MPC strategies are widely adopted in the optimization of WDNs because they not only inherit MPC's ability to accommodate constraints explicitly but also possess the advantages of fault tolerance, less computation and being flexible to system structure [14]. This makes them also widely used in other large-scale systems, such as the transportation system [15], metallurgical process [16] and microgrid system [17,18]. The pressure optimization performance in WDNs is critical because it is related to the degree of reducing leakages and energy consumption. Hence, the DMPC strategy, which can achieve better global performance than the decentralized MPC, becomes more appropriate. In DMPC coordination strategies, according to the type of cost function optimized in each subsystem-based MPC, the simplest and most adopted one is that each local controller only considers its own subsystem's performance, where the global performance can not be guaranteed, but with high flexibility. Another commonly used coordination strategy is that each subsystem-based MPC optimizes the cost of the overall system [19], where the global performance is improved, but the flexibility is reduced, and the communication load increases. In order to guarantee performance improvement with appropriate communication burden among subsystem, an extended scheme based on a so-called "neighbourhood optimization" is proposed by Zhang and Li [20], where the optimization objective of each subsystem-based MPC considers not only the performance of the local subsystem but also those of its neighbours. Taking into account flexibility and the optimization performance, we adopt the neighbourhood optimization strategy in the pressure optimization of large-scale WDNs in this paper.

The type of WDNs considered in this paper is a common network in most cities in China. It is different from the WDNs described in the existing literature both in the structure and the control mode of the actuator. In network structure, the reservoir is located in some of the pump stations, and it acts as a buffer when water demand changes. Hence, it blocks the coupling between upstream and downstream pump stations, which increases the freedom and improves the effect of optimization. However, the reservoir in other WDNs is used as a storage facility [11] and even directly provides supply pressure to users. As for the control mode of the actuator (pump station), the local controller located in actuator follows the pressure set-point instead of the flow set-point and adjusts the flow automatically. Based on the previous two reasons, in addition to the steady-state model used in [3,4], reservoir's model should also be included in the pressure optimization problem. The control-oriented model used in [12,13] includes the model of the reservoir, but it takes the flow of actuator as input and does not include the pressure-flow relationship. Hence, it is not an appropriate model, and the model of this type of WDNs used in pressure optimization problem is still to be constructed.

Large-scale characteristics of WDNs usually mean that the construction of WDNs lasts for a long time. Limited by the management capability or replacement of operating agencies, obtaining the parameter information of each component is time-consuming or even impossible, and the topology information of WDNs may not be crystal clear. Therefore, for large-scale WDNs, it is challeng-

ing to obtain a mechanistic model based on the topology information and the parameter information of each component. Simultaneously, the pressure-flow relationship in mechanistic model is nonlinear, which increases the computation burden dramatically when solving the pressure optimization problem. Thus, in order to obtain an appropriate model for pressure optimization, a modeling and partitioning method using operation data is proposed in this paper. The coupling strength among subsystems is also determined.

In the proposed DMPC scheme, to further increase flexibility and scalability, only strong couplings between the subsystems are preserved. The ignored weak couplings between the subsystems are treated as the bounded disturbance as in [21]. A robust MPC approach proposed by Mayne et al. [22] is used to handle the bounded disturbance. With the existence of a suitable decentralized auxiliary control law, the stability of the closed-loop control system is guaranteed.

Overall, the main contribution of this paper consists of establishing the model for the most common type of WDNs in China, which combines a novel, efficient and operation data based partitioning method. With the obtained subsystem models, an enhancing cooperative distributed model predictive control (EC-DMPC) strategy which combines the neighbourhood optimization strategy, the robust invariant set and the corresponding control law is applied to the pressure optimization of large-scale WDNs. This method could guarantee closed-loop stability and keep the supply pressure to users homogeneous in the premise of meeting users' water demand.

This paper is organized as follows: Section 2 describes the WDNs studied in this paper. Section 3 introduces the modeling process, which includes the operational data based partitioning method. In Section 4, the optimization problem is introduced, and the EC-DMPC strategy is demonstrated. The properties of the EC-DMPC algorithm are also analyzed. The experiment results derived from the application of the modeling, partitioning and the EC-DMPC strategy are discussed in Section 5. Finally, conclusions are drawn in Section 6.

2. Description of WDNs

The structure and operation of WDNs are varied according to the number of water sources, location, size and construction time of the city. As illustrated in Fig. 1, the main components of the WDNs considered in this paper include water treatment plant, pump station (including pump station and pump station with reservoir) and pipe. Raw water is purified in water treatment plants and then delivered to users through the network consisting of pipes and pump stations. Pump stations exist both in the outlet of water treatment plants and in the pipes. They play the role of pressurization. When the supply pressure of pump stations is sufficient, the users' pressure can be met, and clean water can be distributed to all users. There are reservoirs in some of the pump stations. The reservoir acts as a buffer and blocks the coupling between upstream and downstream pump stations, which increases the freedom of optimization and makes it an essential component for the pressure optimization there. Users are coupled with multiple actuators through pipes, so different combinations of actuators could be selected to meet users' water demand. For ease of monitoring and management, sensors are also distributed in WDNs to detect operation data, such as pressure and flow.

The operation structure of WDNs is shown in Fig. 2. The control structure is similar to most other WDNs and can be divided into two layers [12]. The upper layer (supervisory layer) is in charge of the global control of the network and establishes the set-point of the local controller (L-C) at the lower layer. The supervisory layer controller is of advanced type and could operate in a centralized or non-centralized manner, which is the DMPC in this paper. The

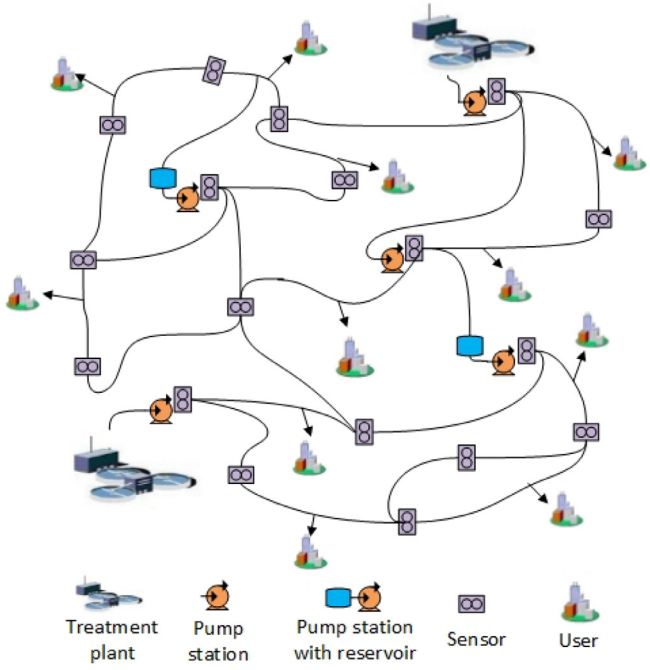


Fig. 1. Schematic diagram of the WDNs.

local controller usually controls each actuator directly to track the set-point. There is also a Supervisory Control And Data Acquisition (SCADA) system in the supervisory layer nowadays, which obtains data from sensors, such as pressure sensors in pump stations and level sensors in reservoirs. The monitored and recorded data in the SCADA system facilitate the pressure optimization of WDNs, and they can also be employed as an index to test the effects of pressure optimization.

3. Modeling and partitioning of WDNs

For large-scale WDNs, the information used in the mechanistic model is very difficult to obtain, and the mechanistic model is very complicated. Moreover, the existing control-oriented model [12] represents the relationship between the reservoir's level and pump station's flow but does not contain the nonlinear relationship between flow and pressure. Hence, a control-oriented model for the pressure optimization problem used in DMPC is built. The structure and representation of the model are given based on the analysis of WDNs' mechanism first. A clustering method *k*-shape and the canonical correlation analysis (CCA) are the main methods used in the proposed partitioning scheme. Subsystem models and the coupling strength among subsystems are determined by the partitioning of WDNs. Coefficients of the model are identified using operation data.

3.1. Model structure and model representation

Based on hydraulics, there are two balances in WDNs, namely mass balance and energy balance. The level change, inflow and outflow of the reservoir meet the mass balance [12], which is expressed as the current level of the reservoir is obtained from the previous level by accumulating the inflow and outflow in the sampling interval, i.e. it is an integration relationship. Users' water demand is the flow required by users, and together with other flows constitutes another mass balance. The energy balance exists in pipes and is expressed as the pressure difference of the pipe provides energy for the flow in the pipe. The flow-pressure relationship in the pipe is a static nonlinear relationship [21]. As de-

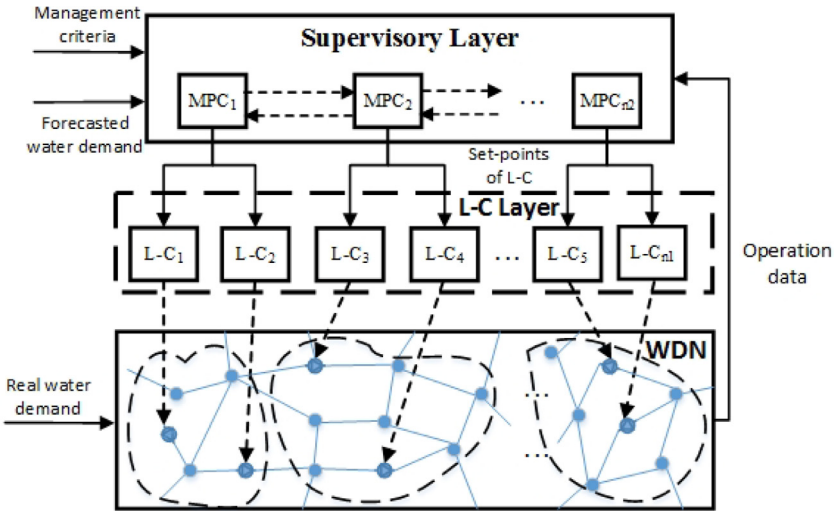


Fig. 2. The operation structure of the WDNs.

scribed in Section 2, reservoirs are located in some of the pump stations. Simultaneously, pump stations with reservoirs are connected with other pump stations and users through interlaced pipe network, so the model of reservoir's level has an integrator and complicated nonlinear contributions of pump stations' outlet pressures and users' water demand. Users and pump stations are also connected through interlaced pipe network, so there is a nonlinear relationship among users' pressures, users' water demand and other pressures in the network. Hence, the mechanistic model for a large-scale WDN, which should include all the components' information and topology information could be more complicated, and then the mechanistic model is discarded.

In this paper, pump stations are actuators, and the outlet pressures of pump stations can be manipulated, so the outlet pressures of pump stations are selected as input variables and constitute the set $\mathbb{U} = \{u_1, \ldots, u_{n_u}\}$, where u_i is the outlet pressure of the ith pump station, and n_u is the number of pump stations. As users' pressures are variables to be optimized, so they are selected as output variables and collected in set $\mathbb{Y} = \{y_1, \ldots, y_{n_y}\}$, where y_i is the pressure of the ith user node, and n_y is the number of user nodes. The levels of reservoirs are time-varying and considered as state variables, which are included in set $\mathbb{X} = \{x_1, \ldots, x_{n_x}\}$, where x_i is the level of the ith reservoir, and n_x is the number of reservoirs. Besides, users' water demand in the ith water demand forecast area d_i is considered as known disturbance and usually can be forecasted by multiple methods, such as methods in [22–24]. n_d is the number of water demand forecast areas. Furthermore, based on the analysis on the mechanism of WDNs, the system model consists of the model of reservoirs' levels and the model of users' pressures, namely a state equation and an output equation. The state equation has an integrator plus a nonlinear part because the current level of the reservoir is obtained from the previous level by accumulating the nonlinear influence of pump stations' outlet pressures and users' water demand in the sampling interval. The output equation has nonlinear contributions of inputs and the known disturbance because users' pressures are influenced by pump stations' outlet pressures and users' water demand through the interlaced pipe network.

In the constructed control-oriented model, the nonlinear part is replaced by a linear part to reduce the model's complexity. The model only characterizes the numerical relationship among variables. Overall, for ease of control and optimization, the model representation is selected as

$$\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}_1\boldsymbol{u}(t) + \boldsymbol{B}_2\boldsymbol{d}(t), \tag{1a}$$

$$\boldsymbol{y}(t) = \boldsymbol{D}_1\boldsymbol{u}(t) + \boldsymbol{D}_2\boldsymbol{d}(t), \tag{1b}$$

where $\boldsymbol{x} = [x_1, \ldots, x_{n_x}]^T$, $\boldsymbol{u} = [u_1, \ldots, u_{n_u}]^T$, $\boldsymbol{y} = [y_1, \ldots, y_{n_y}]^T$ and $\boldsymbol{d} = [d_1, \ldots, d_{n_d}]^T$. Based on the mass balance of reservoirs, $\boldsymbol{A}$ is a unit diagonal matrix. $\boldsymbol{B}_1$, $\boldsymbol{B}_2$, $\boldsymbol{D}_1$ and $\boldsymbol{D}_2$ are coefficient matrices with appropriate dimensions. These coefficient matrices have no clear physical meaning, but can represent the couplings between variables. They are obtained by identification.

Here, partial least squares (PLS) regression is selected as the identification method because it can overcome the ill-conditioned problem and large variance caused by the ordinary least squares regression for models with input collinearity like WDNs, and is widely used in process modeling and monitoring to deal with a large number of variables with collinearity [25]. In the identification procedure, the operation data of all the input, output and state variables in the same period with the same sampling interval are gathered first. Then the variables in each subsystem model are determined with the partitioning method described later. Coefficients in each subsystem model are identified using PLS with the corresponding variables' operation data at last. The operation data of each variable used in the coefficient identification is the part after eliminating the data which have significant deviate with the pressure's lower bound so that the model could represent the WDNs' operation with acceptable precision.

Remark 1. The constructed model can reflect the actual operation situation of WDNs and is a proper model for optimization because the operation data used in identification contain the complete information of WDNs including leakages, which are inevitable in the normal operation of WDNs. The composition and structure of WDNs do not change unless there is an accident, which means the characteristics of each component in WDNs are consistent, and then the leakages in WDNs are not changed. Hence, the actual operation data used in modeling already contain information on leakages.

3.2. Partitioning method review

Many partitioning methods have been proposed for the decentralized or distributed control of WDNs, such as method proposed by Ocampo-Martinez et al. [13] and methods proposed by Di Nardo et al. [26–28]. Most of these methods are based on graph-theory, so the topology of WDNs is the piece of information that must be known. The resulting coupling among subsystems reflects the

coupling on physical topology and does not reflect the strength of the influence on the control. Considering that the operation data are easy to obtain in modern large-scale WDNs, and they contain the coupling information between variables, partitioning subsystems based on operation data may be a feasible choice. Partitioning method based on operation data has been studied and applied in the decomposition of chemical process by Li et al. [29], where output variables are assigned to subsystems based on the knowledge of the system, and then input variables are assigned to different subsystems based on the coupling relationship with the output variables in each subsystem. Unlike the chemical process, there is no sequential process flow in the operation of WDNs, and the boundaries of each subsystem are challenging to be identified, so, in this paper, a method is proposed to decompose the output variables. Besides, in order to make the strength of the coupling be reasonably quantified, an improved indicator is proposed.

The proposed partitioning method divides the WDN into k subsystems, and the coupling strength among subsystems is also determined. For subsystem S_i, $i \in \mathbb{I}_{1:k}$, define $\mathcal{N}_{i(strong)}$ as the set of strong-coupling upstream neighbours, which have a strong influence on S_i. Define $\mathcal{N}_{i(weak)}$ as the set of weak-coupling upstream neighbours, which have a weak influence on S_i. Then the model for S_i is obtained and written as

$$\boldsymbol{x}_i(t+1) = \boldsymbol{A}_{ii}\boldsymbol{x}_i(t) + \boldsymbol{B}_{1,ii}\boldsymbol{u}_i(t) + \boldsymbol{B}_{2,ii}\boldsymbol{d}_i(t) + \sum_{j\in\mathcal{N}_{i(strong)}} \boldsymbol{B}_{1,ij}\boldsymbol{u}_j(t) + \sum_{h\in\mathcal{N}_{i(weak)}} \boldsymbol{B}_{1,ih}\boldsymbol{u}_h(t), \tag{2a}$$

$$\boldsymbol{y}_i(t) = \boldsymbol{D}_{1,ii}\boldsymbol{u}_i(t) + \boldsymbol{D}_{2,ii}\boldsymbol{d}_i(t) + \sum_{j\in\mathcal{N}_{i(strong)}} \boldsymbol{D}_{1,ij}\boldsymbol{u}_j(t) + \sum_{h\in\mathcal{N}_{i(weak)}} \boldsymbol{D}_{1,ih}\boldsymbol{u}_h(t), \tag{2b}$$

where

$$\mathcal{N}_{i(weak)} \cup \mathcal{N}_{i(strong)} = \mathcal{N}_i.$$

In this model, $\boldsymbol{A}_{ii}$ is a unit diagonal matrix with appropriate dimensions based on the mechanistic model of reservoirs.

The definition of strong coupling between subsystems will be elaborated more details in a later subsection. A clustering method called k-shape is adopted in the partitioning of output variables. CCA result is used to quantify the coupling strength among subsystems. These two methods are briefly introduced next.

3.2.1. k-Shape

For each variable, the obtained operation data are time-series sequences. The process of allocating variables into subsystems according to the coupling is the process of clustering time-series sequences according to their shapes. There are many methods to cluster time-series sequences [30], and all of these methods need to specify the number of clusters in advance. A clustering method called k-Shape proposed by Paparrizos and Gravano [31] is chosen because the distance measurement method used is invariant to scaling and shifting, and it is robust. With the previously specified number of clusters k, this algorithm assigns a set of variables into k clusters $\mathcal{C}_i$, $\forall i \in \mathbb{I}_{1:k}$, and a variable can only be partitioned into one cluster.

3.2.2. Canonical correlation analysis

Canonical correlation analysis (CCA) is a statistical analysis method used to analyze the correlation between two groups of variables. The CCA algorithm maximizes the correlation between a linear combination of two groups of variables. Specifically, for a group of variables (such as state, input and output variables) with a total number of g, operation data of each variable are collected with the same sampling interval, and then a time-series sequence of length m is obtained for each variable. All the data are normalized to form an m-by-g matrix $\boldsymbol{X}$. An m-by-q matrix $\boldsymbol{Y}$ is formed by the normalized data of q variables in another group. Then the linear combinations of the column vectors of the matrix $\boldsymbol{X}$ and $\boldsymbol{Y}$ are

$$\boldsymbol{\alpha} = \boldsymbol{X}\boldsymbol{a}, \boldsymbol{\beta} = \boldsymbol{Y}\boldsymbol{b},$$

respectively, where $\boldsymbol{a}$ and $\boldsymbol{b}$ are combination coefficient vectors. The objective of CCA is

$$\max_{\boldsymbol{a},\boldsymbol{b}} J = \rho(\boldsymbol{\alpha}, \boldsymbol{\beta}),$$

where $\rho(\boldsymbol{\alpha}, \boldsymbol{\beta})$ is the correlation coefficient of $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$. By solving the above optimization problem, the optimal coefficient vectors $\boldsymbol{a}^*$ and $\boldsymbol{b}^*$ are obtained. The value of the ith element in $\boldsymbol{a}^*$ represents the extent to which the ith variable in $\boldsymbol{X}$ affects another group of variables. Same holds for the elements in $\boldsymbol{b}^*$.

3.3. Partitioning algorithm

Considering the application of the obtained partition to distributed control strategy, an ideal partition result is that the output variables in $\mathbb{Y}$, state variables in $\mathbb{X}$ and input variables in $\mathbb{U}$ are allocated to subsystems without overlapping, and strong couplings between subsystems are determined. The obtained subsystems also should have the similar number of output variables, means that covers areas of similar size. The operation data used in the proposed partitioning method include not only time-series sequences of variables in $\mathbb{U}$ and $\mathbb{Y}$ but also the time-series sequences of the inlet pressure of pump stations with reservoirs. So define the inlet pressure of the pump station with the ith reservoir as v_i and $\mathbb{V} = \{v_1, \ldots, v_{n_x}\}$. The WDN is marked as $S(\mathbb{U}, \mathbb{X}, \mathbb{Y})$, and the partitioning problem can be established as follows:

Problem 1. Given a WDN system $S(\mathbb{U}, \mathbb{X}, \mathbb{Y})$, $\mathbb{U}$ denotes the variable set of pump station's outlet pressure, $\mathbb{X}$ denotes the variable set of reservoir's level, and $\mathbb{Y}$ denotes the variable set of users' pressure. $\mathbb{U}_i$, $\mathbb{X}_i$ and $\mathbb{Y}_i$ denote the sets of variables in subsystem S_i, and $\#\mathbb{Y}_i$ represents the number of variables in $\mathbb{Y}_i$. $C_{\mathbb{X}_i,u_j}$ denotes the strong coupling between $\mathbb{X}_i$ and u_j, $C_{\mathbb{Y}_i,u_j}$ denotes the strong coupling between $\mathbb{Y}_i$ and u_j, $i \in \mathbb{I}_{1:k}$, $j \in \mathbb{I}_{1:n_u}$ and $u_j \notin \mathbb{U}_i$. Find k, $k \in \mathbb{I}_{\geq 1}$ subsystems $S_1(\mathbb{U}_1, \mathbb{X}_1, \mathbb{Y}_1)$, $S_2(\mathbb{U}_2, \mathbb{X}_2, \mathbb{Y}_2)$, $\ldots$, $S_k(\mathbb{U}_k, \mathbb{X}_k, \mathbb{Y}_k)$ of $S(\mathbb{U}, \mathbb{X}, \mathbb{Y})$ such that

(1) $\bigcup_{i=1}^{k} \mathbb{U}_i = \mathbb{U}$, $\bigcup_{i=1}^{k} \mathbb{X}_i = \mathbb{X}$ and $\bigcup_{i=1}^{k} \mathbb{Y}_i = \mathbb{Y}$,
(2) $\mathbb{U}_i \cap \mathbb{U}_j = \emptyset$, $\mathbb{X}_i \cap \mathbb{X}_j = \emptyset$, $\mathbb{Y}_i \cap \mathbb{Y}_j = \emptyset$, $\forall i \in \mathbb{I}_{1:k}$, $\forall j \in \mathbb{I}_{1:k}$ and $i \neq j$,
(3) $\mathbb{U}_i \neq \emptyset$, $\mathbb{Y}_i \neq \emptyset$, $\forall i \in \mathbb{I}_{1:k}$,
(4) $\#\mathbb{Y}_1 \approx \#\mathbb{Y}_2 \approx \ldots \approx \#\mathbb{Y}_k$,
(5) $C_{\mathbb{X}_i,u_j}$ and $C_{\mathbb{Y}_i,u_j}$, $i \in \mathbb{I}_{1:k}$, $j \in \mathbb{I}_{1:n_u}$ and $u_j \notin \mathbb{U}_i$, are determined.

In Problem 1, the strong coupling between $\mathbb{Y}_i$ and u_j is defined as follows:

Definition 1. (Strong coupling between $\mathbb{Y}_i$ and u_j, i.e. $C_{\mathbb{Y}_i,u_j}$) For two groups of variables $\mathbb{Y}_i$ and $\mathbb{U}$, $\forall i \in \mathbb{I}_{1:k}$, $\boldsymbol{a}^*_{\mathbb{Y}_i}$ and $\boldsymbol{b}^*_{\mathbb{Y}_i}$ are the solved optimal coefficient vectors in CCA, respectively. Take absolute value of elements in vector $\boldsymbol{b}^*_{\mathbb{Y}_i}$, and get the ordinal value in descending order of the absolute value. The obtained ordinal value is recorded in $\boldsymbol{b}^O_{\mathbb{Y}_i}$. If the jth element in $\boldsymbol{b}^O_{\mathbb{Y}_i}$ is smaller than a predetermined threshold $T_{\mathbb{Y}}$, then there is a strong coupling between $\mathbb{Y}_i$ and u_j.

The definition of strong coupling between $\mathbb{X}_i$ and u_j is similar to Definition 1. It is also defined that if there is a strong coupling

between $\mathbb{Y}_i$ and $u_j \in \mathbb{U}_m$, $i, m \in \mathbb{I}_{1:k}$, $i \neq m$, or there is a strong coupling between $\mathbb{X}_i$ and $u_j \in \mathbb{U}_m$, $i, m \in \mathbb{I}_{1:k}$, $i \neq m$, then the subsystems S_i and S_m are strongly coupled subsystems.

The proposed partitioning method solves Problem 1 with considering the specific characteristics of WDNs. The three main steps and the corresponding considered characteristics are explained as follows:

Step 1: Determine the number of subsystems k and $\mathbb{Y}_i$, $\forall i \in \mathbb{I}_{1:k}$. In the control of WDNs, part of users' pressures are usually controlled by a part of the pump stations, so the users' pressures are clustered according to the coupling relationship with pump stations. The time-series data of users' pressure nodes, which are coupled to the same part of pump stations, present a similar shape. Therefore, we first determine the number of subsystems k by the shape of the time-series data of the variables in $\mathbb{Y}$, and then $\mathbb{Y}_i$, $\forall i \in \mathbb{I}_{1:k}$ are decided. The method used in this step is based on k-shape and noted as Users' Pressure Partitioning (UPP), which will be described later.

Step 2: Determine $\mathbb{U}_i$, $\mathbb{X}_i$ and $C_{\mathbb{Y}_i,u_j}$, $\forall i \in \mathbb{I}_{1:k}$, $j \in \mathbb{I}_{1:n_u}$ and $u_j \notin \mathbb{U}_i$. With the obtained $\mathbb{Y}_i$, $\forall i \in \mathbb{I}_{1:k}$, for all $j \in \mathbb{I}_{1:n_u}$, if u_j has the strongest coupling (the ordinal value is minimum) with $\mathbb{Y}_i$, then u_j is assigned into $\mathbb{U}_i$. Besides, if there is still a strong coupling between $\mathbb{Y}_i$ and u_j, then $C_{\mathbb{Y}_i,u_j}$ is recorded. As the reservoir is located in the pump station, so the variable in $\mathbb{X}$ is assigned into the same subsystem with its corresponding input variable in $\mathbb{U}$. With the determination of $\mathbb{U}_i$, $\forall i \in \mathbb{I}_{1:k}$, $\mathbb{X}_i$ are also determined.

Step 3: Determine $C_{\mathbb{X}_i,u_j}$, $i \in \mathbb{I}_{1:k}$, $j \in \mathbb{I}_{1:n_u}$ and $u_j \notin \mathbb{U}_i$. As the reservoir is located in the pump station, the reservoir's level is affected by the inlet and outlet pressure of the pump station. So it is obvious that there is a strong coupling between the reservoir's level and the pump station's outlet pressure. The inlet pressure is related to the upstream pump stations, so the other couplings between $\mathbb{X}_i$ and u_j could be determined based on the couplings between $\mathbb{V}_i$ and u_j. $C_{\mathbb{X}_i,u_j}$ is recorded if the coupling between $\mathbb{V}_i$ and u_j is strong.

The algorithms of Step 2 and 3 are similar. In Step 2, the coupling strength between u_j and different $\mathbb{Y}_i$, $\forall i \in \mathbb{I}_{1:k}$ needs to be compared because the input variables in $\mathbb{U}$ should be assigned to the subsystem with the strongest coupling. According to the definition of CCA, the greater the absolute value of the optimal coefficient vector's element, the stronger the coupling between the variables. However, it is not suitable for direct comparison between different coefficient vectors. Therefore, unlike the method of setting a threshold for $\boldsymbol{b}^*_{\mathbb{Y}_i}$ directly in [29], the modified method in this paper first processes the elements in $\boldsymbol{b}^*_{\mathbb{Y}_i}$ as in Definition 1: (1) taking the absolute value of the elements in $\boldsymbol{b}^*_{\mathbb{Y}_i}$, (2) obtaining the ordinal value based on absolute value in descending order, (3) recording the ordinal value in vector $\boldsymbol{b}^O_{\mathbb{Y}_i}$. Ordinal value makes the comparison between different coefficient vectors more intuitive and reasonable. Specifically, the process of the algorithm in Step 2 is: the ordinal vectors $\boldsymbol{b}^O_{\mathbb{Y}_i}$, $\forall i \in \mathbb{I}_{1:k}$ constituent a n_u-by-k matrix $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$, then the input variable u_j, $\forall j \in \mathbb{I}_{1:n_u}$, corresponds to the jth row of $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$; if the ith element is the minimum in the jth row of $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$, then u_j is assigned into $\mathbb{U}_i$; if the ith element in the jth row of $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$ is no more than a threshold $T_{\mathbb{Y}}$, but not the minimum in this row, then $C_{\mathbb{Y}_i,u_j}$ is recorded. The algorithm in Step 3 is similar to the algorithm in Step 2. In Step 3, the ordinal value vectors are marked as $\boldsymbol{b}^O_{\mathbb{V}_i}$. The ordinal matrix is marked as $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{V}}}$, and the threshold is $T_{\mathbb{V}}$. The thresholds $T_{\mathbb{Y}}$ and $T_{\mathbb{V}}$ are positive integers greater than 1, which are determined by the requirement of flexibility. The main procedure of the partitioning method and the specific algorithms used in Step 2 and 3 are summarized in Algorithm 1.

Algorithm 1 The partitioning method.

Input: Threshold $T_{\mathbb{Y}}$ and $T_{\mathbb{V}}$, $\boldsymbol{M}_{\mathbb{Y}}$ is a n_y-by-m matrix, $\boldsymbol{M}_{\mathbb{U}}$ is a n_u-by-m matrix and $\boldsymbol{M}_{\mathbb{V}}$ is a n_x-by-m matrix. The rows of matrices $\boldsymbol{M}_{\mathbb{Y}}$, $\boldsymbol{M}_{\mathbb{U}}$ and $\boldsymbol{M}_{\mathbb{V}}$ are normalized time-series data of variables in $\mathbb{Y}$, $\mathbb{U}$ and $\mathbb{V}$, respectively. The maximum number of subsystems $n_{\max}$.

Output: $S_i(\mathbb{U}_i, \mathbb{X}_i, \mathbb{Y}_i)$, $C_{\mathbb{Y}_i,u_j}$ and $C_{\mathbb{X}_i,u_j}$, $\forall i \in \mathbb{I}_{1:k}$, $j \in \mathbb{I}_{1:n_u}$ and $u_j \notin \mathbb{U}_i$.

Step 1
$\{k, \mathbb{Y}_1, \ldots, \mathbb{Y}_k\} \leftarrow UPP(\boldsymbol{M}_{\mathbb{Y}}, n_{\max})$;
$\mathbb{U}_i, \mathbb{X}_i, \mathbb{V}_i \leftarrow \emptyset$, $\forall i \in \mathbb{I}_{1:k}$;
Step 2
$\{\boldsymbol{a}^*_{\mathbb{Y}_i}, \boldsymbol{b}^*_{\mathbb{Y}_i}\} \leftarrow CCA(\boldsymbol{M}^T_{\mathbb{Y}_i}, \boldsymbol{M}^T_{\mathbb{U}})$, $\forall i \in \mathbb{I}_{1:k}$;
$\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}} \leftarrow$ ordinal value of absolute value of $\boldsymbol{b}^*_{\mathbb{Y}_i}$, $\forall i \in \mathbb{I}_{1:k}$;
if $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}(j,i) = \min \boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}(j)$, $\forall i \in \mathbb{I}_{1:k}$ and $\forall j \in \mathbb{I}_{1:n_u}$ **then**
  $\mathbb{U}_i \leftarrow \mathbb{U}_i + \mathbb{U}(j)$;
end if
if $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}(j,i) \leq T_{\mathbb{Y}}$ and $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}(j,i) \neq \min \boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}(j)$, $\forall i \in \mathbb{I}_{1:k}$ and $\forall j \in \mathbb{I}_{1:n_u}$ **then**
  $C_{\mathbb{Y}_i,u_j} \leftarrow$ a strong coupling between $\mathbb{Y}_i$ and u_j;
end if
$\mathbb{X}_i \leftarrow$ partition result of $\mathbb{U}_i$, $\forall i \in \mathbb{I}_{1:k}$;
Step 3
$\mathbb{V}_i \leftarrow$ partition result of $\mathbb{U}_i$, $\forall i \in \mathbb{I}_{1:k}$;
$\{\boldsymbol{a}^*_{\mathbb{V}_i}, \boldsymbol{b}^*_{\mathbb{V}_i}\} \leftarrow$CCA$(\boldsymbol{M}^T_{\mathbb{V}_i}, \boldsymbol{M}^T_{\mathbb{U}})$, $\forall i \in \mathbb{I}_{1:k}$;
$\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{V}}} \leftarrow$ ordinal value of absolute value of $\boldsymbol{b}^*_{\mathbb{V}_i}$, $\forall i \in \mathbb{I}_{1:k}$;
if $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{V}}}(j,i) \leq T_{\mathbb{V}}$, $\forall i \in \mathbb{I}_{1:k}$ and $\forall j \in \mathbb{I}_{1:n_u}$ **then**
  $C_{\mathbb{X}_i,u_j} \leftarrow$ a strong coupling between $\mathbb{X}_i$ and u_j;
end if

In Step 3, if the number of variables in $\mathbb{X}$ is relatively small, $C_{\mathbb{X}_i,u_j}$ can be determined using the CCA result between v_i, $\forall i \in \mathbb{I}_{1:n_x}$ and $\mathbb{U}$. By making such changes, the coupling between x_i, $\forall i \in \mathbb{I}_{1:n_x}$ and u_j, $\forall j \in \mathbb{I}_{1:n_u}$ is obtained. With this result, the coupling between $\mathbb{X}_i$ and u_j is clearer, and then the modeling accuracy could be improved.

The problem solved by the UPP method in Step 1 is part of Problem 1. This problem can be established as follows:

Problem 2. Given the time-series data of variables in $\mathbb{Y}$, determine the number of subsystems k and $\mathbb{Y}_1, \mathbb{Y}_2, \ldots, \mathbb{Y}_k$ such that

(1) $\bigcup_{i=1}^{k} \mathbb{Y}_i = \mathbb{Y}$,
(2) $\mathbb{Y}_i \cap \mathbb{Y}_j = \emptyset$, $\forall i \in \mathbb{I}_{1:k}$, $\forall j \in \mathbb{I}_{1:k}$ and $i \neq j$,
(3) $\mathbb{Y}_i \neq \emptyset$, $\forall i \in \mathbb{I}_{1:k}$,
(4) $\#\mathbb{Y}_1 \approx \#\mathbb{Y}_2 \approx \ldots \approx \#\mathbb{Y}_k$.

Two issues need to be addressed: the first one is the number of subsystems k, and the second is the partition of $\mathbb{Y}$. The number of subsystem k is determined by considering the balance of the number of output variables in each subsystem and the clustering result. The concept of balanced subsystem references the statement by Ocampo-Martinez et al. [13]. That is, if the number of output variables in each subsystem is similar, these subsystems are called balanced subsystems. In this concept, the silhouettes coefficient is selected as the criterion to measure the quality of clustering [32]. The silhouette value is a measure of how similar a variable is to its own cluster (cohesion) compared to other clusters (separation). After being clustered into k clusters, for each variable y_i, let e_i be the average distance between y_i and all other variables within the same cluster. The average dissimilarity of variable y_i to cluster $\mathcal{C}_j$ is the average of the distance from y_i to all variables in $\mathcal{C}_j$. Let f_i be the minimum average distance of y_i to all variables in any other

cluster, of which y_i is not a member. Then the silhouettes coefficient is defined as:

$$sh_i = \frac{f_i - e_i}{\max\{e_i, f_i\}}, \tag{3}$$

where, $-1 \le sh_i \le 1$. An sh_i close to 1 means that the data is appropriately clustered, and an sh_i close to -1 means that the data would be more appropriate if it was clustered in its neighbouring cluster. Therefore, when all the sh_i are greater than 0, the clustering result is recorded as an acceptable result. With the acceptable clustering result, the variance of the number of output variables in each cluster is used to measure the subsystems' balance. The smaller the variance, the more balanced the subsystems are. Based on the preceding two rules, the complete algorithm of UPP method is summarized in Algorithm 2.

Algorithm 2 $\{k, \mathbb{Y}_1, \ldots, \mathbb{Y}_k\} = UPP(\boldsymbol{M}_{\mathbb{Y}}, n_{\max})$.

Input: $\boldsymbol{M}_{\mathbb{Y}}$ is a n_y-by-m matrix. The rows of matrix $\boldsymbol{M}_{\mathbb{Y}}$ are normalized time-series data of variables in $\mathbb{Y}$. Maximum number of subsystems $n_{\max}$.

Output: The number of subsystems k and $\{\mathbb{Y}_1, \ldots, \mathbb{Y}_k\}$.

$\mathbb{K} \leftarrow \emptyset$;
for $i = 2$ to $n_{\max}$ **do**
  $\boldsymbol{r}_i \leftarrow k$-Shape$(\boldsymbol{M}_{\mathbb{Y}}, i)$; ▷ $\boldsymbol{r}_i$ is a n_y-by-1 vector which records the clustering result for every variable in $\mathbb{Y}$
  Calculate $sh_{i,j}, \forall j \in \mathbb{I}_{1:n_y}$;
  if all the $sh_{i,j} > 0$, for $j \in \mathbb{I}_{1:n_y}$ **then**
    $\mathbb{K} \leftarrow \mathbb{K} \oplus \{i\}$;
  end if
end for ▷ Find the acceptable clustering results and record in $\mathbb{K}$
if $\#\mathbb{K} = 1$ **then**
  $k \leftarrow \mathbb{K}$;
else
  $Var_i \leftarrow$ the variance of the number of variables when the number of clusters is $\mathbb{K}(i), \forall i \in \mathbb{I}_{1:\#\mathbb{K}}$;
  if $Var_i = \min_{j=1}^{\#\mathbb{K}} Var_j$ **then**
    $k \leftarrow \mathbb{K}(i)$;
  end if
end if
$\{\mathbb{Y}_1, \ldots, \mathbb{Y}_k\} \leftarrow$ the k-Shape clustering result when the number of clusters is k;

Overall, the main novelties of the partitioning algorithm presented in this paper are allocating the output variables by the algorithm based on k-Shape with considering the balance of subsystems. The method to determine the strength of coupling is also improved. The partitioning method can be implemented automatically. As the coupling between output variables is not considered in the partitioning, the model of each subsystem can be identified separately by using corresponding input and output variables' operation data.

4. Pressure optimization problem and enhancing cooperative distributed model predictive control

With the obtained partitioning result and the model of WDNs, an EC-DMPC strategy is applied to the pressure optimization problem of large-scale WDNs. The EC-DMPC strategy combines a robustness-based scheme and the neighbourhood optimization coordination strategy. The information of weakly coupled neighbours is considered as a disturbance in the subsystem model, which is handled by the robust invariant set in local MPC controller. In this section, the pressure optimization problem is described at first. The details of the EC-DMPC algorithm are introduced secondly, and the closed-loop stability of the algorithm is analyzed at last.

4.1. Pressure optimization problem

In the operation of WDNs, the goal of pressure optimization is to keep users' pressure homogeneous and close to the lower bound to reduce leakage and energy consumption while meeting users' water demand. Define $y_{\min}$ as the lower bound for all elements in $\boldsymbol{y}$. The ideal optimization effects for an area are shown in Fig. 3, where the average pressure is reduced, and the pressure for all the subsystems is more homogeneous after optimization. For a large-scale WDN which has been divided into k subsystems, the constraint for $\boldsymbol{y}_i, \forall i \in \mathbb{I}_{1:k}$ is written as

$$\boldsymbol{y}_i(t) \in \mathcal{Y}_i, \mathcal{Y}_i = \{\boldsymbol{y}_i | \boldsymbol{y}_i \ge \boldsymbol{y}_{i,\min}\}, \forall t, \forall i \in \mathbb{I}_{1:k}, \tag{4}$$

where $\boldsymbol{y}_i$ is the vector composed by the variables in $\mathbb{Y}_i$, and $\boldsymbol{y}_{i,\min}$ is the vector with appropriate dimensions.

The structure of WDNs also limits the optimization of pressure. Fortunately, with reservoirs in pump stations, the effects of pressure optimization could be improved. So the physical limits for the reservoir's level $\boldsymbol{x}_i, \forall i \in \mathbb{I}_{1:k}$ should also be satisfied, which is

$$\boldsymbol{x}_i(t) \in \mathcal{X}_i, \mathcal{X}_i = \{\boldsymbol{x}_i | \boldsymbol{x}_{i,\min} \le \boldsymbol{x}_i \le \boldsymbol{x}_{i,\max}\}, \forall t, \forall i \in \mathbb{I}_{1:k}. \tag{5}$$

As there are physical limits for the pump station's outlet pressure $\boldsymbol{u}_i$, the constraint for the pump station's outlet pressure

$$\boldsymbol{u}_i(t) \in \mathcal{U}_i, \mathcal{U}_i = \{\boldsymbol{u}_i | \boldsymbol{u}_{i,\min} \le \boldsymbol{u}_i \le \boldsymbol{u}_{i,\max}\}, \forall t, \forall i \in \mathbb{I}_{1:k}, \tag{6}$$

is also considered.

Overall, when the optimization horizon is N, the pressure optimization problem could be written as

$$\min_{\boldsymbol{u}} \sum_{i=1}^{k} \sum_{l=0}^{N-1} \|\boldsymbol{y}_i(t+l|t) - \boldsymbol{y}_{i,\min}\|^2_{\boldsymbol{W}_{\boldsymbol{y}_i}}, \tag{7}$$

subjects to (1), (4)–(6). Where $\boldsymbol{W}_{\boldsymbol{y}_i}$ is the weight matrix.

For the overall system, $\boldsymbol{x}$, $\boldsymbol{u}$ and $\boldsymbol{y}$ fit constraints $\boldsymbol{x}(t) \in \mathcal{X}$, $\boldsymbol{u}(t) \in \mathcal{U}$ and $\boldsymbol{y}(t) \in \mathcal{Y}$, where $\mathcal{X} = (\times \mathcal{X}_i)$, $\mathcal{U} = (\times \mathcal{U}_i)$, $\mathcal{Y} = (\times \mathcal{Y}_i)$, $\forall i \in \mathbb{I}_{1:k}$ and $\times$ denotes Cartesian product. Because all actuators can be controlled independently, the constraints are not coupled after partitioning. However, in the optimization of the whole system, the constraints are coupled through the subsystem models.

4.2. The algorithm of EC-DMPC

For the statement of individual MPC sub-problems solved in the EC-DMPC strategy, we rely on the robust MPC algorithm presented in [33] for constrained linear systems with bounded disturbance. Neglect the weakly coupling neighbours' influence $\boldsymbol{w}_{1,i}(t) = \sum_{h \in \mathcal{N}_{i(weak)}} \boldsymbol{B}_{1,ih}\boldsymbol{u}_h$, $\boldsymbol{w}_{2,i}(t) = \sum_{h \in \mathcal{N}_{i(weak)}} \boldsymbol{D}_{1,ih}\boldsymbol{u}_h$, and define the ith subsystem nominal model associated with Eq. (2) as

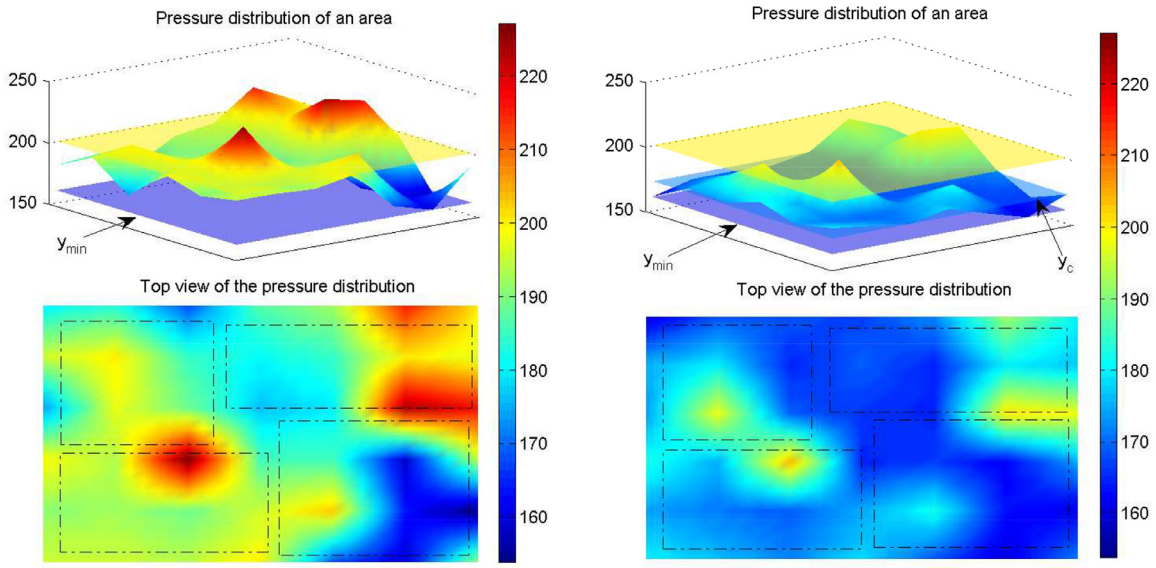
$$\bar{\boldsymbol{x}}_i(t+1) = \boldsymbol{A}_{ii}\bar{\boldsymbol{x}}_i(t) + \boldsymbol{B}_{1,ii}\bar{\boldsymbol{u}}_i(t) + \boldsymbol{B}_{2,ii}\boldsymbol{d}_i(t) + \sum_{j \in \mathcal{N}_{i(strong)}} \boldsymbol{B}_{1,ij}\bar{\boldsymbol{u}}_j(t), \tag{8a}$$

$$\bar{\boldsymbol{y}}_i(t) = \boldsymbol{D}_{1,ii}\bar{\boldsymbol{u}}_i(t) + \boldsymbol{D}_{2,ii}\boldsymbol{d}_i(t) + \sum_{j \in \mathcal{N}_{i(strong)}} \boldsymbol{D}_{1,ij}\bar{\boldsymbol{u}}_j(t), \tag{8b}$$

where $\bar{\boldsymbol{x}}_i$, $\bar{\boldsymbol{u}}_i$and $\bar{\boldsymbol{y}}_i$ represent the state, input and output of the nominal model. The weakly coupling neighbours' influence $\boldsymbol{w}_{1,i} \in \mathcal{W}_{1,i}$ and $\boldsymbol{w}_{2,i} \in \mathcal{W}_{2,i}$, where $\mathcal{W}_{1,i} = (\oplus \boldsymbol{B}_{1,ih}\mathcal{U}_h)$, $\mathcal{W}_{2,i} = (\oplus \boldsymbol{D}_{1,ih}\mathcal{U}_h)$, $\forall h \in \mathcal{N}_{i(weak)}$, $\oplus$ denotes Minkowski set addition. With the nominal subsystem model, the nominal overall system model with new coupling matrices $\bar{\boldsymbol{B}}_1$ and $\bar{\boldsymbol{D}}_1$ is, therefore:

$$\bar{\boldsymbol{x}}(t+1) = \boldsymbol{A}\bar{\boldsymbol{x}}(t) + \bar{\boldsymbol{B}}_1\bar{\boldsymbol{u}}(t) + \boldsymbol{B}_2\boldsymbol{d}(t), \tag{9a}$$

$$\bar{\boldsymbol{y}}(t) = \bar{\boldsymbol{D}}_1\bar{\boldsymbol{u}}(t) + \boldsymbol{D}_2\boldsymbol{d}(t). \tag{9b}$$



(a) Pressure distribution of an area before optimization. (b) Pressure distribution of an area after optimization.

Fig. 3. Diagram of preassure optimization goal.

where coefficient matrix $\boldsymbol{A}$ is still the unit diagonal matrix. The new coefficient matrices $\bar{\boldsymbol{B}}_1$ and $\bar{\boldsymbol{D}}_1$ ignore the elements which represent the weak coupling compared to $\boldsymbol{B}_1$ and $\boldsymbol{D}_1$. These ignored elements are set 0. The input variables of the original model $\boldsymbol{u}$ and the input variables of the nominal model $\bar{\boldsymbol{u}}$ represent the same physical components, but the values of them are different. $\boldsymbol{u}$ not only includes the part from $\bar{\boldsymbol{u}}$, but also includes another part to eliminate the effect of disturbance based on the robust invariant set control law. Define $\boldsymbol{u} = \bar{\boldsymbol{u}} + \boldsymbol{K}(\boldsymbol{x} - \bar{\boldsymbol{x}})$, where $\boldsymbol{K}$ is the robust invariant set control gain. The robust invariant set control law used here is defined as follows.

Definition 2 (Robust invariant set control law)**.** Given $\boldsymbol{e} = (\boldsymbol{x} - \bar{\boldsymbol{x}})$ which represents dynamics of the error between the original plant and the nominal model:

$$\boldsymbol{e}(t+1) = \boldsymbol{A}_{\mathcal{K}}\boldsymbol{e}(t) + \boldsymbol{w}_1(t), \tag{10}$$

with $\boldsymbol{A}_{\mathcal{K}} = (\boldsymbol{A} + \bar{\boldsymbol{B}}_1\boldsymbol{K})$. A set ϕ is called a robust invariant set for system (10), if $\boldsymbol{A}_{\mathcal{K}}\phi \oplus \mathcal{W}_1 \subseteq \phi$. And the control is called robust invariant set control law.

The definition of robust invariant set illustrates that for system $\boldsymbol{x}(t+1) = \boldsymbol{A}\boldsymbol{x}(t) + \bar{\boldsymbol{B}}_1\boldsymbol{u}(t) + \boldsymbol{B}_2\boldsymbol{d}(t) + \boldsymbol{w}_1(t)$, if ϕ and the robust invariant set control gain $\boldsymbol{K}$ exist, then for $\boldsymbol{e}(0) = \boldsymbol{x}(0) - \bar{\boldsymbol{x}}(0)$, at arbitrary time t, the deviation between the trajectories of the original system $\boldsymbol{x}(t)$ and the trajectories of the nominal system $\bar{\boldsymbol{x}}(t)$ can be controlled in ϕ, namely, $\boldsymbol{x}(t) - \bar{\boldsymbol{x}}(t) \in \phi$. Assumptions for robust invariant set and the corresponding control law are summarized as follows.

Assumption 1. There exists a block-diagonal matrix $\boldsymbol{K} = blockdiag\{\boldsymbol{K}_1, \boldsymbol{K}_2, \ldots, \boldsymbol{K}_k\}$ such that

1. $\boldsymbol{A} + \bar{\boldsymbol{B}}_1\boldsymbol{K}$ is Schur.
2. ϕ_i is an admissible robust invariant set for S_i's deviation $(\boldsymbol{x}_i - \bar{\boldsymbol{x}}_i)$ subject to constraints $(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_i) \in \mathcal{F}_i$, $\mathcal{F}_i : (\mathcal{X}_i \times \mathcal{U}_i) \ominus (\phi_i \times \boldsymbol{K}_i\phi_i)$ and $\bar{\boldsymbol{y}}_i \in \mathcal{Y}_i \ominus \mathcal{W}_{2,i} \ominus \sum_{j\in\{i\}\cup\mathcal{N}_{i(strong)}} \boldsymbol{D}_{1,ij}\boldsymbol{K}_i\phi_i$, $\forall i \in \mathbb{I}_{1:k}$.
3. $\boldsymbol{u}_{i,e} = \boldsymbol{K}_i\boldsymbol{e}_i$ is the corresponding robust invariant set control law for S_i, $\forall i \in \mathbb{I}_{1:k}$.
4. the original system is asymptotically stable.

For subsystem S_i, $\forall i \in \mathbb{I}_{1:k}$, Assumption 1 ensures that the state estimated by the nominal model (8) is close to the real system's trajectory before system being stable, and the original system's state is controlled in $\mathcal{X}_i$. Specifically, on the basis of Definition 2 and Assumption 1, the dynamics of deviation $\boldsymbol{e}_i = \boldsymbol{x}_i - \bar{\boldsymbol{x}}_i$ is written as

$$\boldsymbol{e}_i(t+1) = \boldsymbol{A}_{ii}\boldsymbol{e}_i(t) + \boldsymbol{B}_{1,ii}\boldsymbol{u}_{i,e}(t) + \sum_{j\in\mathcal{N}_{i(strong)}} \boldsymbol{B}_{1,ij}\boldsymbol{u}_{j,e}(t) + \boldsymbol{w}_{1,i}(t), \tag{11}$$

where $\boldsymbol{u}_{i,e} = \boldsymbol{K}_i\boldsymbol{e}_i$ and $\boldsymbol{u}_{j,e} = \boldsymbol{K}_j\boldsymbol{e}_j$, $\forall j \in \mathcal{N}_{i(strong)}$. When $(\bar{\boldsymbol{x}}_i(t), \bar{\boldsymbol{u}}_i(t)) \in \mathcal{F}_i$ and $(\bar{\boldsymbol{x}}_j(t), \bar{\boldsymbol{u}}_j(t)) \in \mathcal{F}_j$, $\forall j \in \mathcal{N}_{i(strong)}$, the state and input of the original system satisfy $(\boldsymbol{x}_i(t), \boldsymbol{u}_i(t)) \in \mathcal{X}_i \times \mathcal{U}_i$, where $\boldsymbol{u}_i(t) = \bar{\boldsymbol{u}}_i(t) + \boldsymbol{K}_i(\boldsymbol{x}_i(t) - \bar{\boldsymbol{x}}_i(t))$. Thus, with the help of robust invariant set, the optimization of the original system can be obtained by solving a new optimization problem based on the nominal model and tighter constraints.

In order to increase the flexibility of coordination while maintaining global performance, neighbourhood optimization is adopted in the proposed EC-DMPC strategy. The objective function of each DMPC considers the performance of itself and its downstream neighbours. Specifically, in the controller for subsystem i, the objective function includes the objective of the ith subsystem and systems in $\mathcal{H}_i$, where $\mathcal{H}_i = \{h | i \in \mathcal{N}_{h(strong)}, \forall h \in \mathbb{I}_{1:k}, h \neq i\}$, i.e. $\mathcal{H}_i$ is the set of subsystems which are strongly affected by S_i. Therefore, the objective function is defined as

$$\begin{aligned} V_{iN}(\boldsymbol{x}_i, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}) &= V_{iN,L}(\boldsymbol{x}_i, \boldsymbol{x}_{i,r}, \boldsymbol{y}_{i,\min}; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}) \\ &+ \sum_{h\in\mathcal{H}_i} V_{hN,L}(\boldsymbol{x}_h, \boldsymbol{x}_{h,r}, \boldsymbol{y}_{h,\min}; \bar{\boldsymbol{x}}_h, \bar{\boldsymbol{u}}_{h,0:N-1}), \end{aligned} \tag{12}$$

with

$$\begin{aligned} &V_{iN,L}(\boldsymbol{x}_i, \boldsymbol{x}_{i,r}, \boldsymbol{y}_{i,\min}; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}) \\ &= \sum_{l=0}^{N-1} \left(\|\bar{\boldsymbol{x}}_i(t+l|t) - \boldsymbol{x}_{i,r}\|^2_{\boldsymbol{W}_{\boldsymbol{x}_i}} + \|\bar{\boldsymbol{y}}_i(t+l|t) - \boldsymbol{y}_{i,\min}\|^2_{\boldsymbol{W}_{\boldsymbol{y}_i}} \right) \\ &+ \|\bar{\boldsymbol{x}}_i(t+N|t) - \boldsymbol{x}_{i,r}\|^2_{\boldsymbol{P}_i}, \end{aligned} \tag{13}$$

where reservoir's level $\bar{\boldsymbol{x}}_i$ is also added in the objective as a quadratic expression because it still needs to be stabilized to a

reference level $\boldsymbol{x}_{i,r}$ when $\boldsymbol{u}_i$ is not changed. This ensures the normal operation of the buffer function of the reservoir. $\boldsymbol{W}_{\boldsymbol{x}_i}$, as a $n_{x,i} \times n_{x,i}$ matrix, is the weight term. $n_{x,i}$ is the number of reservoirs in subsystem S_i. The weight term $\boldsymbol{W}_{\boldsymbol{x}_i}$ should be set as a smaller value in the application as the main goal is to optimize pressure. In the objective $V_{i,N}(\boldsymbol{x}_i, \boldsymbol{x}_r, \boldsymbol{y}_{min}; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1})$, $\boldsymbol{x}_i$, $\boldsymbol{x}_r$ and $\boldsymbol{y}_{min}$ are the given initial state, reference value and lower pressure bound value, $\bar{\boldsymbol{x}}_i$ is the initial state of the nominal system, $\bar{\boldsymbol{u}}_{i,0:N-1}$ are the input predictions from 0 to $N-1$ sample time ahead the current sample time, namely, $\bar{\boldsymbol{u}}_{i,0:N-1}(t) = \{\bar{\boldsymbol{u}}_i(t|t), \ldots, \bar{\boldsymbol{u}}_i(t+N-1|t)\}$. $\|\bar{\boldsymbol{x}}_i(t+N|t) - \boldsymbol{x}_{i,r}\|^2_{\boldsymbol{P}_i}$ is the terminal cost function $V_{i,f}$ and $V_f = \|\bar{\boldsymbol{x}}(t+N|t) - \boldsymbol{x}_r\|^2_{\boldsymbol{P}}$, where $\boldsymbol{P} = blockdiag\{\boldsymbol{P}_1, \ldots, \boldsymbol{P}_k\}$.

To ensure the closed-loop stability of the system, the terminal invariant set $\mathcal{X}_{i,f} = \{\bar{\boldsymbol{x}}_i | \|\bar{\boldsymbol{x}}_i(t+N|t) - \boldsymbol{x}_{i,r}\|^2_{\boldsymbol{P}_i} \le \alpha_i\}$ is also added as constraints in the optimization problem. Assumptions used to prove the closed-loop system's stability are given as follows. Concrete theorem and the analysis of stability are given in next subsection.

Assumption 2. There exists a matrix $\bar{\boldsymbol{K}} = blockdiag(\bar{\boldsymbol{K}}_1, \bar{\boldsymbol{K}}_2, \ldots, \bar{\boldsymbol{K}}_k)$ and a matrix $\boldsymbol{P} = blockdiag(\boldsymbol{P}_1, \boldsymbol{P}_2, \ldots, \boldsymbol{P}_k)$ such that

1. $\boldsymbol{A} + \bar{\boldsymbol{B}}_1 \bar{\boldsymbol{K}}$ is Schur.
2. $\mathcal{X}_{i,f} \subset \mathcal{X}_i \ominus \phi_i$ is an invariant set for $\bar{\boldsymbol{x}}_i(t)$, $\forall i \in \mathbb{I}_{1:k}$.
3. $\bar{\boldsymbol{u}}_i = \bar{\boldsymbol{K}}_i(\bar{\boldsymbol{x}}_i - \boldsymbol{x}_{i,r}) + \bar{\boldsymbol{u}}_{i,r} \in \mathcal{U}_i \ominus \boldsymbol{K}_i\phi_i$ is the corresponding terminal control law for any $\bar{\boldsymbol{x}}_i \in \mathcal{X}_{i,f}$, where $\bar{\boldsymbol{u}}_{i,r}$ corresponds to the input when the state $\bar{\boldsymbol{x}}_i$ stabilizes to $\boldsymbol{x}_{i,r}$, $\forall i \in \mathbb{I}_{1:k}$.
4. $\bar{\boldsymbol{y}}_i \in \mathcal{Y}_i \ominus \mathcal{W}_{2,i} \ominus \sum_{j\in\{i\}\cup\mathcal{N}_{i(strong)}} \boldsymbol{D}_{1,ij}\boldsymbol{K}_j\phi_j$, $\forall i \in \mathbb{I}_{1:k}$.
5. $V_f(\bar{\boldsymbol{x}}(t+1)) + l(\bar{\boldsymbol{x}}(t), \bar{\boldsymbol{u}}(t)) \le V_f(\bar{\boldsymbol{x}}(t))$, $\forall \bar{\boldsymbol{x}} \in \mathcal{X}_f$, where $\mathcal{X}_f = (\times\mathcal{X}_{i,f})$, $\forall i \in \mathbb{I}_{1:k}$. $l(\bar{\boldsymbol{x}}(t), \bar{\boldsymbol{u}}(t)) = ||\bar{\boldsymbol{x}}(t) - \boldsymbol{x}_r||_{\boldsymbol{W}_{\boldsymbol{x}}} + ||\bar{\boldsymbol{y}}(t) - \boldsymbol{y}_{min}||_{\boldsymbol{W}_{\boldsymbol{y}}}$, $\boldsymbol{W}_{\boldsymbol{x}} = blockdiag\{\boldsymbol{W}_{\boldsymbol{x}_1}, \ldots, \boldsymbol{W}_{\boldsymbol{x}_k}\}$ and $\boldsymbol{W}_{\boldsymbol{y}} = blockdiag\{\boldsymbol{W}_{\boldsymbol{y}_1}, \ldots, \boldsymbol{W}_{\boldsymbol{y}_k}\}$.

It can be seen that several conditions in Assumption 2 are the same as the conditions in Assumption 1. However, based on Definition 2, in the choice of robust invariant set ϕ, it is necessary to include disturbance set $\mathcal{W}_1$. That is, conditions for the robust invariant set are stronger than that for the terminal invariant set. Hence, $\boldsymbol{K}$ and $\bar{\boldsymbol{K}}$ are designed separately in the proposed scheme.

Remark 2. As for the structure of $\boldsymbol{K}$ and $\bar{\boldsymbol{K}}$, in addition to the block-diagonal form used in this paper, it can also be chosen as other forms, which may provide more potentials to get better performance or reduce the requirements to the system. If we select a structure related to the strong neighbourhoods, when constructing the feasible solution for the next sample, $\bar{\boldsymbol{x}}$ should be firstly calculated and then sent to S_i. Subsystems in $\mathcal{H}_i$ could also be fed back to S_i with additional communication load and fixed communication topology. In either case, the structural feedback control law can be obtained by carefully constructing a corresponding linear matrix inequality (LMI) optimization problem to satisfy Assumption 1.1–1.3 and 2. However, in the nominal model considered, there is no direct relationship between the states of subsystems in $\mathcal{N}_{i(strong)}$ and the states of S_i, which makes it difficult to analyze the performance of different structures. Hence, for the simplicity in expression, we only provide a possible structure other than the best structure for feedback control gain in Assumption 1 and 2.

In the proposed EC-DMPC algorithm, at each sampling time, with a feasible input, the controllers of the subsystems solve the optimization problem in parallel. In order to ensure consistency, the optimization problems are solved iteratively. Define p as the number of iterations. This distributed optimization is of the Gauss-Jacobi type [34]. In the pth iteration of sample time t, the algorithm is as follows:

Firstly, for each subsystem S_i, $V_{i,N}$ fits

$$V_{iN}(\boldsymbol{x}_i, \boldsymbol{x}_r, \boldsymbol{y}_{\min}, p; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}). \tag{14}$$

Compute the optimal solution of the problem $\mathbb{P}_{i,N}$, which is defined by

$$\left(\bar{\boldsymbol{x}}_i'^{[p]}(t), \bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]}(t)\right) = \arg\min_{\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}} V_{iN}(\boldsymbol{x}_i, \boldsymbol{x}_r, \boldsymbol{y}_{\min}, p; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}), \tag{15}$$

subjects to constraints

$$\bar{\boldsymbol{x}}_i(t+l+1|t) = \boldsymbol{A}_{ii}\bar{\boldsymbol{x}}_i(t+l|t) + \boldsymbol{B}_{1,ii}\bar{\boldsymbol{u}}_i(t+l|t) + \boldsymbol{B}_{2,ii}\boldsymbol{d}_i(t+l|t) + \sum_{j\in\mathcal{N}_{i(strong)}} \boldsymbol{B}_{1,ij}\bar{\boldsymbol{u}}_j^{[p-1]}(t+l|t), l = 0, \ldots, N-1, \tag{16a}$$

$$\bar{\boldsymbol{x}}_{h_i}(t+l+1|t) = \boldsymbol{A}_{h_ih_i}\bar{\boldsymbol{x}}_{h_i}(t+l|t) + \boldsymbol{B}_{1,h_ih_i}\bar{\boldsymbol{u}}_{h_i}^{[p-1]}(t+l|t) + \boldsymbol{B}_{1,h_ii}\bar{\boldsymbol{u}}_i(t+1|t) + \boldsymbol{B}_{2,h_ih_i}\boldsymbol{d}_{h_i}(t+l|t) + \sum_{h_j\in\mathcal{N}_{h_i(strong)}, h_j\neq i} \boldsymbol{B}_{1,h_ih_j}\bar{\boldsymbol{u}}_{h_j}^{[p-1]}(t+l|t), \forall h_i \in \mathcal{H}_i, \quad l = 0, \ldots, N-1, \tag{16b}$$

$$\bar{\boldsymbol{y}}_i(t+l|t) = \boldsymbol{D}_{1,ii}\bar{\boldsymbol{u}}_i(t+l|t) + \boldsymbol{D}_{2,ii}\boldsymbol{d}_i(t+l|t) + \sum_{j\in\mathcal{N}_{h_i(strong)}} \boldsymbol{D}_{1,ij}\bar{\boldsymbol{u}}_j^{[p-1]}(t+l|t), l = 0, \ldots, N-1, \tag{16c}$$

$$\bar{\boldsymbol{y}}_{h_i}(t+l|t) = \boldsymbol{D}_{1,h_ih_i}\bar{\boldsymbol{u}}_{h_i}^{[p-1]}(t+l|t) + \boldsymbol{D}_{1,h_ii}\bar{\boldsymbol{u}}_i(t+l|t) + \boldsymbol{D}_{2,h_ih_i}\boldsymbol{d}_{h_i}(t+l|t) + \sum_{h_j\in\mathcal{N}_{h_i(strong)}, h_j\neq i} \boldsymbol{D}_{1,h_ih_j}\bar{\boldsymbol{u}}_{h_j}^{[p-1]}(t+l|t), \forall h_i \in \mathcal{H}_i, \quad l = 0, \ldots, N-1, \tag{16d}$$

$$\boldsymbol{x}_g(t) - \bar{\boldsymbol{x}}_g(t) \in \phi_g, \forall g \in \{i\} \cup \mathcal{H}_i, \tag{16e}$$

$$(\bar{\boldsymbol{x}}_g(t+l|t), \bar{\boldsymbol{u}}_g(t+l|t)) \in \mathcal{F}_g, \forall g \in \{i\} \cup \mathcal{H}_i, l = 0, \ldots, N-1, \tag{16f}$$

$$\bar{\boldsymbol{y}}_g(t+l|t) \in \mathcal{Y}_g \ominus \mathcal{W}_{2,g} \ominus \sum_{q\in\{g\}\cup\mathcal{N}_{g(strong)}} \boldsymbol{D}_{1,gq}\boldsymbol{K}_q\phi_q, \quad \forall g \in \{i\} \cup \mathcal{H}_i, l = 0, \ldots, N-1, \tag{16g}$$

$$\bar{\boldsymbol{x}}_g(t+N|t) \in \mathcal{X}_{g,f}, \forall g \in \{i\} \cup \mathcal{H}_i, \tag{16h}$$

where $\bar{\boldsymbol{x}}_{h_i}(t|t) = \bar{\boldsymbol{x}}_{h_i}^{[p-1]}(t|t)$, $\forall h_i \in \mathcal{H}_i$. The optimization function (14) updates S_i's initial state ($\bar{\boldsymbol{x}}_i'(0)$) and input in N steps ($\bar{\boldsymbol{u}}_{i,0:N-1}'$) based on the information from subsystems in $\mathcal{H}_i$, $\mathcal{N}_{i(strong)}$ and $\mathcal{N}_{h_i(strong)}$.

Next, set

$$\bar{\boldsymbol{u}}_{i,0:N-1}^{[p]}(t) = \gamma_i\bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]}(t) + (1-\gamma_i)\bar{\boldsymbol{u}}_{i,0:N-1}^{[p-1]}(t), \tag{17a}$$

$$\bar{\boldsymbol{x}}_i^{[p]}(t) = \gamma_i\bar{\boldsymbol{x}}_i'^{[p]}(t) + (1-\gamma_i)\bar{\boldsymbol{x}}_i^{[p-1]}(t), \tag{17b}$$

$$\sum_{i=1}^{k} \gamma_i = 1, \gamma_i > 0, \tag{17c}$$

where $\gamma_i \in \mathbb{R}$, $0 < \gamma_i < 1$, which is used to guarantee the consistency of the optimization problem, i.e. at the end of the current sample time, all shared variables converge.

Input: Initial time t_0, initial state $\boldsymbol{x}_i(t_0)$, reference $\boldsymbol{x}_{i,r}$ and $\boldsymbol{y}_{i,min}$, $\mathcal{N}_{i(strong)}$ and $\mathcal{H}_i$ for subsystem S_i, $\forall i \in \mathbb{I}_{1:k}$, the maximum number of iterations p_{max}.

Output: The control law $\boldsymbol{u}_i^*(t)$, for $t = t_0 : +\infty$ and $\forall i \in \mathbb{I}_{1:k}$.

$t \leftarrow t_0$;
$(\bar{\boldsymbol{x}}_i^{[0]}(t), \bar{\boldsymbol{u}}_{i,0:N-1}^{[0]}(t)) \leftarrow (\bar{\boldsymbol{x}}_i^{[0]}(t_0), \bar{\boldsymbol{u}}_{i,0:N-1}^{[0]}(t_0))$;
while True **do**
  $\boldsymbol{x}_i \leftarrow \boldsymbol{x}_i(t)$, $p \leftarrow 1$;
  while True **do**
    Receive $(\bar{\boldsymbol{x}}_j^{[p-1]}(t), \bar{\boldsymbol{u}}_{j,0:N-1}^{[p-1]}(t))$ from S_j, $\forall j \in \mathcal{H}_i$, $\bar{\boldsymbol{u}}_{h,0:N-1}^{[p-1]}$ from S_h, $\forall h \in \mathcal{N}_{i(strong)} \cup \mathcal{N}_{j(strong)}$;
    $(\bar{\boldsymbol{x}}_i'^{[p]}(t), \bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]}(t)) \leftarrow \arg\min\{V_{iN}(\boldsymbol{x}_i, \boldsymbol{x}_r, \boldsymbol{y}_{min}, p; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}) : s.t(16)\}$;
    $\bar{\boldsymbol{u}}_{i,0:N-1}^{[p]}(t) \leftarrow \gamma_i \bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]}(t) + (1-\gamma_i)\bar{\boldsymbol{u}}_{i,0:N-1}^{[p-1]}(t)$;
    $\bar{\boldsymbol{x}}_i^{[p]}(t) \leftarrow \gamma_i \bar{\boldsymbol{x}}_i'^{[p]}(t) + (1-\gamma_i)\bar{\boldsymbol{x}}_i^{[p-1]}(t)$;
    $\sum_{i=1}^{k} \gamma_i = 1, \gamma_i > 0$;
    if $\|\bar{\boldsymbol{u}}_i^{[p]}(t) - \bar{\boldsymbol{u}}_i^{[p-1](t)}\| \le 1e^{-6}$ or $p > p_{max}$ **then**
      Break;
    end if
    $p \leftarrow p+1$;
  end while
  $\bar{\boldsymbol{x}}_i^*(t) \leftarrow \bar{\boldsymbol{x}}_i^{[p]}(t)$, $\bar{\boldsymbol{u}}_{i,0:N-1}^* \leftarrow \bar{\boldsymbol{u}}_{i,0:N-1}^{[p]}(t)$;
  $\boldsymbol{u}_{i,0}^*(t) \leftarrow \bar{\boldsymbol{u}}_{i,0}^*(t) + \boldsymbol{K}_i(\boldsymbol{x}_i(t) - \bar{\boldsymbol{x}}_i^*(t))$;
  $\boldsymbol{u}_i^*(t) \leftarrow \boldsymbol{u}_{i,0}^*(t)$;
  $\bar{\boldsymbol{x}}_i^{[0]}(t+1) \leftarrow \bar{\boldsymbol{x}}_i^*(t+1|t)$, $\bar{\boldsymbol{u}}_{i,0:N-1}^{[0]}(t+1) \leftarrow [\bar{\boldsymbol{u}}_{i,1:N-1}^*(t), \bar{\boldsymbol{K}}_i(\bar{\boldsymbol{x}}_i^*(t+N|t) - \boldsymbol{x}_{i,r}) + \bar{\boldsymbol{u}}_{i,r}]^T$;
  $t \leftarrow t+1$;
end while

Algorithm 3. EC-DMPC

After these two steps, let $p = p+1$ to iterate until converge. Then we have $\bar{\boldsymbol{x}}_i^*(t) = \bar{\boldsymbol{x}}_i^{[p]}(t)$ and $\bar{\boldsymbol{u}}_{i,0:N-1}^*(t) = \bar{\boldsymbol{u}}_{i,0:N-1}^{[p]}(t)$.

Finally, take the control law of S_i as

$$\kappa_{i,N}^* : \boldsymbol{u}_{i,0}^*(t) = \bar{\boldsymbol{u}}_{i,0}^*(t) + \boldsymbol{K}_i(\boldsymbol{x}_i(t) - \bar{\boldsymbol{x}}_i^*(t)), \tag{18}$$

where, $\bar{\boldsymbol{u}}_{i,0}^*(t)$ is the first element of $\bar{\boldsymbol{u}}_{i,0:N-1}^*$. The assumption that there is a feasible solution $(\bar{\boldsymbol{x}}_i^{[0]}(t_0), \bar{\boldsymbol{u}}_{i,0:N-1}^{[0]}(t_0))$ at initial time t_0 holds. At the end of arbitrary $t > t_0$, a constructed feasible solution $(\hat{\boldsymbol{x}}_i^{[0]}(t+1), \hat{\boldsymbol{u}}_{i,0:N-1}^{[0]}(t+1))$ based on the solution at t is assigned to $(\bar{\boldsymbol{x}}_i^{[0]}(t+1), \bar{\boldsymbol{u}}_{i,0:N-1}^{[0]}(t+1))$. The overall algorithms are summarized in Algorithm 3.

4.3. Stability analysis

In this section, we analyze the closed-loop stability under the proposed EC-DMPC strategy. In order to facilitate the subsequent proof, the optimization problem solved in the EC-DMPC algorithm is firstly equivalent to an optimization problem with overall objective function and constraints of all the subsystems. The iterative convergence is proposed in Lemma 1 next. Then the recursive feasibility and convergence of the algorithm follow in Lemmas 2 and 3. The concrete theorem is provided as Theorem 1 at last.

As in (12), because all the strongly coupled downstream subsystems are considered in the optimization problem of S_i, at each iteration p, for each subsystem S_i, $i \in \mathbb{I}_{1:k}$, it has

$$\begin{aligned}
(\bar{\boldsymbol{x}}_i'^{[p]}(t), \bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]}(t)) &= \arg\min_{\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}} V_{iN} \\
&= \arg\min_{\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}} \sum_{j \in \{i\} \cup \mathcal{H}_i} V_{jN,L} \\
&= \arg\min_{\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}} \left(\sum_{j \in \{i\} \cup \mathcal{H}_i} V_{jN,L} + \sum_{h \in \mathbb{I}_{1:k}, h \notin \{i\} \cup \mathcal{H}_i} V_{hN,L} \right) \\
&= \arg\min_{\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}} \sum_{j \in \mathbb{I}_{1:k}} V_{jN,L} \\
&= \arg\min_{\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}} V_N,
\end{aligned} \tag{19}$$

subjects to (16). Where

$$\begin{aligned}
V_N(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1}) = \sum_{l=0}^{N-1} (&\|\bar{\boldsymbol{x}}(t+l|t) - \boldsymbol{x}_r\|_{\boldsymbol{W}_x}^2 \\
&+ \|\bar{\boldsymbol{y}}(t+l|t) - \boldsymbol{y}_{\min}\|_{\boldsymbol{W}_y}^2) + \|\bar{\boldsymbol{x}}(t+N|t) - \boldsymbol{x}_r\|_{\boldsymbol{P}}^2.
\end{aligned}$$

For subsystem S_i, constraints (16) contain the constraints for subsystem S_i and subsystems in $\mathcal{H}_i$. At iteration p, an optimization problem which takes V_{iN} as objective function and considers all the subsystems' constraints (where $h_i \in \mathbb{I}_{1:k}, h_i \neq i$ and $g \in \mathbb{I}_{1:k}$ in (16)) delivers the same control sequence as problem $\mathbb{P}_{i,N}$ because based on the proposed EC-DMPC algorithm, the information of all other subsystems used is that obtained at the $(p-1)$-th iteration, and then constraints for subsystems which are not in $\mathcal{H}_i$ are obviously satisfied. Hence, combining the result of (19), for subsystem S_i, $\forall i \in \mathbb{I}_{1:k}$, at the p-th iteration, the solutions of optimization problem $\mathbb{P}_{i,N}$ equal to the solutions of the overall optimization problem which takes V_N as objective function and considers all the subsystems' constraints. That is, optimization problems solved by all the subsystems' controllers have the same objective function and constraints, but the optimization variables $(\bar{\boldsymbol{x}}_i, \bar{\boldsymbol{u}}_{i,0:N-1})$ are different from each other.

For ease of analysis, here we take the overall optimization problem to prove feasibility and stability and denote

$$\boldsymbol{z} = \{\bar{\boldsymbol{x}}_1, \ldots, \bar{\boldsymbol{x}}_k, \bar{\boldsymbol{u}}_{1,0:N-1}, \ldots, \bar{\boldsymbol{u}}_{k,0:N-1}\} = \{\bar{\boldsymbol{x}}, \bar{\boldsymbol{u}}_{0:N-1}\},$$

$$\boldsymbol{z}^* = \left\{\bar{\boldsymbol{x}}_1^*, \ldots, \bar{\boldsymbol{x}}_k^*, \bar{\boldsymbol{u}}_{1,0:N-1}^*, \ldots, \bar{\boldsymbol{u}}_{k,0:N-1}^*\right\} = \left\{\bar{\boldsymbol{x}}^*, \bar{\boldsymbol{u}}_{0:N-1}^*\right\},$$

$$\boldsymbol{z}^{[p]} = \left\{\bar{\boldsymbol{x}}_1^{[p]}, \ldots, \bar{\boldsymbol{x}}_k^{[p]}, \bar{\boldsymbol{u}}_{1,0:N-1}^{[p]}, \ldots, \bar{\boldsymbol{u}}_{k,0:N-1}^{[p]}\right\} = \left\{\bar{\boldsymbol{x}}^{[p]}, \bar{\boldsymbol{u}}_{0:N-1}^{[p]}\right\},$$

$$\boldsymbol{z}_i'^{[p]} = \left\{\bar{\boldsymbol{x}}_1^{[p-1]}, \ldots, \bar{\boldsymbol{x}}_i'^{[p]}, \ldots, \bar{\boldsymbol{x}}_k^{[p-1]}, \bar{\boldsymbol{u}}_{1,0:N-1}^{[p-1]}, \ldots, \bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]}, \ldots, \bar{\boldsymbol{u}}_{k,0:N-1}^{[p-1]}\right\}.$$

Define $\mathcal{Z}_N$ as the admissible set for $\mathbb{P}_{i,N}$, $\forall i \in \mathbb{I}_{1:k}$ and $\mathcal{X}_N$ as the feasible set of initial states. The iterative convergence is shown in Lemma 1, where part of the proof references [19,35].

Lemma 1 (Iterative convergence)**.** *The sequence of cost functions* $\{V_N(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{min}; \boldsymbol{z}^{[p]})\}$ *generated by Algorithm 3 is non-increasing*

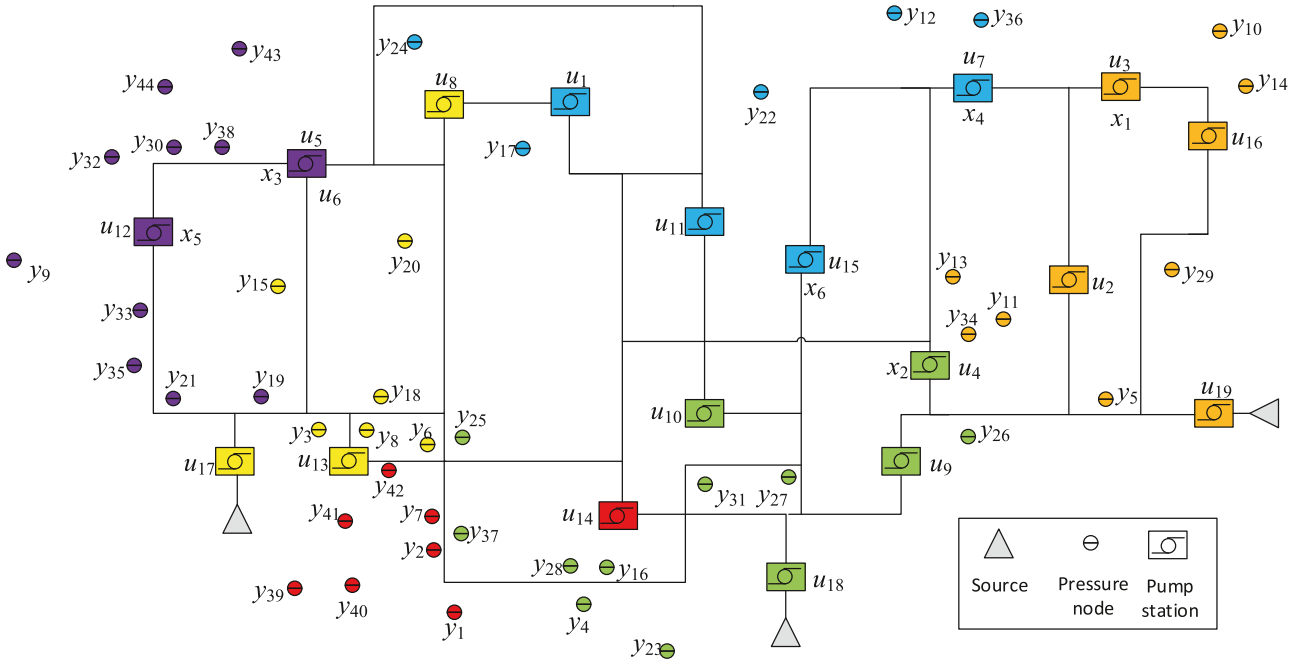


Fig. 4. Subsystem partitioning result.

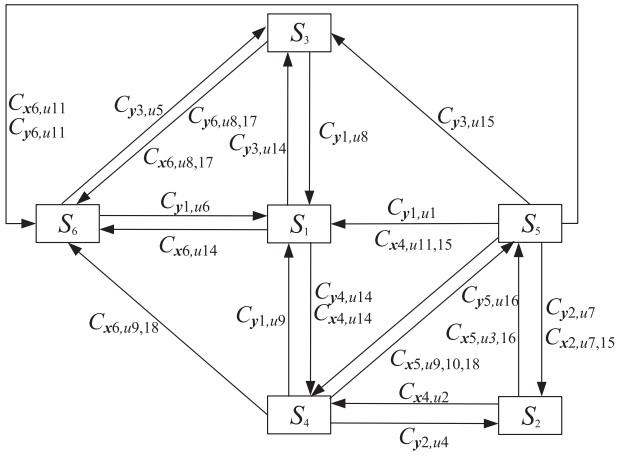


Fig. 5. Subsystem structure and strong couplings.

with iteration number p. The first state, input sequence and cost function of the entire system converge to a unique fixed solution $[\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{u}}_{1,0:N-1}, \ldots, \tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_{k,0:N-1}]$ *and fixed value* V_N^∞*, respectively, when iteration step* $p \to \infty$*. That is*

$$\lim_{p\to\infty}\left[\bar{\boldsymbol{x}}_1^{[p]}, \bar{\boldsymbol{u}}_{1,0:N-1}^{[p]}, \ldots, \bar{\boldsymbol{x}}_k^{[p]}, \bar{\boldsymbol{u}}_{k,0:N-1}^{[p]}\right] = \left[\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{u}}_{1,0:N-1}, \ldots, \tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_{k,0:N-1}\right], \tag{20a}$$

$$\lim_{p\to\infty} V_N^p = V_N^\infty. \tag{20b}$$

Proof. On the basis of the result in (19) and Algorithm 3, for every $p \geq 0$, we know that

$$V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}_i'^{[p]}\right) \leq V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}^{[p-1]}\right), \forall i \in \mathbb{I}_{1:k}, \tag{21}$$

and the cost function satisfies that

$$\begin{aligned}
&V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}^{[p]}\right)\\
&= V_N\Big(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \gamma_1\bar{\boldsymbol{x}}_1'^{[p]} + (1-\gamma_1)\bar{\boldsymbol{x}}_1^{[p-1]}, \ldots, \gamma_k\bar{\boldsymbol{x}}_k'^{[p]} + (1-\gamma_k)\bar{\boldsymbol{x}}_k^{[p-1]},\\
&\quad \gamma_1\bar{\boldsymbol{u}}_{1,0:N-1}'^{[p]} + (1-\gamma_1)\bar{\boldsymbol{u}}_{1,0:N-1}^{[p-1]}, \ldots, \gamma_k\bar{\boldsymbol{u}}_{k,0:N-1}'^{[p]} + (1-\gamma_k)\bar{\boldsymbol{u}}_{k,0:N-1}^{[p-1]}\Big)\\
&= V_N\Big(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \gamma_1\bar{\boldsymbol{x}}_1'^{[p]} + \gamma_2\bar{\boldsymbol{x}}_1^{[p-1]} + \ldots + \gamma_k\bar{\boldsymbol{x}}_1^{[p-1]}, \ldots, \gamma_1\bar{\boldsymbol{x}}_k^{[p-1]} + \ldots + \gamma_k\bar{\boldsymbol{x}}_k'^{[p]},\\
&\quad \gamma_1\bar{\boldsymbol{u}}_{1,0:N-1}'^{[p]} + \gamma_2\bar{\boldsymbol{u}}_{1,0:N-1}^{[p-1]} + \ldots + \gamma_k\bar{\boldsymbol{u}}_{1,0:N-1}^{[p-1]}, \ldots, \gamma_1\bar{\boldsymbol{u}}_{k,0:N-1}^{[p-1]} + \ldots + \gamma_k\bar{\boldsymbol{u}}_{k,0:N-1}'^{[p]}\Big)\\
&\leq \sum_{i\in\mathbb{I}_{1:k}} \gamma_i V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}_i'^{[p]}\right)\\
&\leq \sum_{i\in\mathbb{I}_{1:k}} \gamma_i V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}^{[p-1]}\right)\\
&= V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}^{[p-1]}\right).
\end{aligned} \tag{22}$$

The first and second equalities follow from (17). The first inequality follows from the convexity of $V(\cdot)$. The second inequality follows from the optimality of $(\bar{\boldsymbol{x}}_i'^{[p]}, \bar{\boldsymbol{u}}_{i,0:N-1}'^{[p]})$, $\forall i \in \mathbb{I}_{1:k}$ (i.e. (22)), and the final line follows from $\sum_{i=1}^k \gamma_i = 1$. Equality is obtained if $\boldsymbol{z}^{[p]} = \boldsymbol{z}^{[p-1]}$. Because the cost is bounded below, it converges. The second result (20b) is proved.

A proof for the convergence to a unique fixed solution $[\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{u}}_{1,0:N-1}, \ldots, \tilde{\boldsymbol{x}}_k, \tilde{\boldsymbol{u}}_{k,0:N-1}]$ and fixed value V_N^∞ is given in the Appendix A of [35]. □

Based on the iterative convergence result of Lemma 1, we have

$$V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}^{[p+1]}\right) \leq V_N\left(\boldsymbol{x}, \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \boldsymbol{z}^{[0]}\right). \tag{23}$$

Lemma 2 (Feasibility). *Let Assumption 2 hold, and at time* $t = 0$, $\boldsymbol{x}(0) \in \mathcal{X}_N$ *is given such that the optimization problem* $\mathbb{P}_{i,N}$, $\forall i \in \mathbb{I}_{1:k}$, *is feasible for* $\boldsymbol{z}(0)$*, then the proposed EC-DMPC algorithm is feasible for all time steps* $t \geq 0$.

Proof. This lemma can be proved by assume $\boldsymbol{z}(t) \in \mathcal{Z}_N$, then there exists $\boldsymbol{z}(t+1) \in \mathcal{Z}_N$.

At sample t, $\boldsymbol{z}^*(t) = \left\{\bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}_{0:N-1}^*(t)\right\}$ is feasible for $\mathbb{P}_{i,N}(t)$, $\forall i \in \mathbb{I}_{1:k}$, and $\bar{\boldsymbol{x}}_{i,0:N}^*(t) = \left\{\bar{\boldsymbol{x}}_i^*(t|t), \ldots, \bar{\boldsymbol{x}}_i^*(t+N|t)\right\}$, $\forall i \in \mathbb{I}_{1:k}$ is the associated state trajectory for the nominal system. That is, constraints in (16) are satisfied by $\bar{\boldsymbol{u}}_{i,0:N-1}^*(t)$ and $\bar{\boldsymbol{x}}_{i,0:N}^*(t)$, $\forall i \in \mathbb{I}_{1:k}$.

At sample $t+1$, take the terminal invariant control law in Assumption 2, and get $\hat{\boldsymbol{z}}^{[0]}(t+1) = \left\{\hat{\boldsymbol{x}}^{[0]}(t+1), \hat{\boldsymbol{u}}_{0:N-1}^{[0]}(t+1)\right\}$,

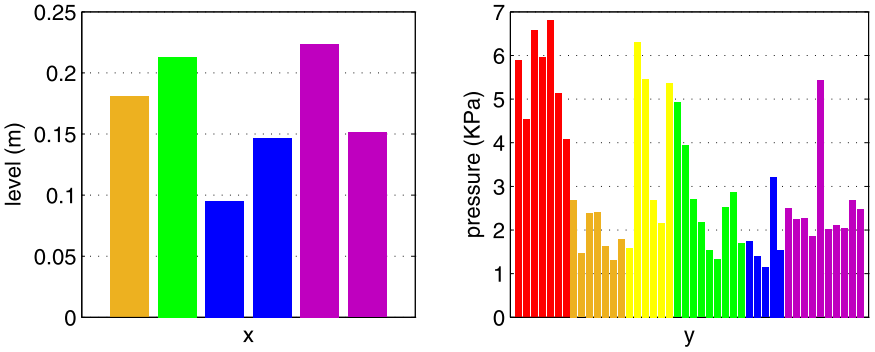


Fig. 6. RMSE for each variable in $\mathbb{X}$ and $\mathbb{Y}$.

where $\hat{\boldsymbol{x}}^{[0]}(t+1) = \bar{\boldsymbol{x}}^*(t+1|t)$ and

$$\hat{\boldsymbol{u}}^{[0]}_{0:N-1}(t+1) = \left\{\bar{\boldsymbol{u}}^*(t+1|t), \ldots, \bar{\boldsymbol{u}}^*(t+N-1|t), \bar{\boldsymbol{K}}(\bar{\boldsymbol{x}}^*(t+N|t) - \boldsymbol{x}_r) + \bar{\boldsymbol{u}}_r\right\}.$$

The state sequences associated with $\hat{\boldsymbol{u}}^{[0]}_{0:N-1}(t+1)$ are $\hat{\boldsymbol{x}}^{[0]}_{0:N}(t+1) = \left\{\bar{\boldsymbol{x}}^*(t+1|t), \ldots, \bar{\boldsymbol{x}}^*(t+N|t), (\boldsymbol{A}+\bar{\boldsymbol{B}}_1\bar{\boldsymbol{K}})(\bar{\boldsymbol{x}}^*(t+N|t) - \boldsymbol{x}_r) + \boldsymbol{x}_r\right\}$.

It is obvious that the first $N-1$ elements in $\hat{\boldsymbol{u}}^{[0]}_{i,0:N-1}(t+1)$ and the first N elements in $\hat{\boldsymbol{x}}^{[0]}_{i,0:N}(t+1)$ satisfy the corresponding constraints in (16), $\forall i \in \mathbb{I}_{1:k}$. Since $\bar{\boldsymbol{x}}^*(t+N|t) \in \mathcal{X}_f$, it follows from Assumption 2 that $(\boldsymbol{A}+\bar{\boldsymbol{B}}_1\bar{\boldsymbol{K}})$ $(\bar{\boldsymbol{x}}^*(t+N|t) - \boldsymbol{x}_r) + \boldsymbol{x}_r \in \mathcal{X}_f$, $\bar{\boldsymbol{K}}(\bar{\boldsymbol{x}}^*(t+N|t) - \boldsymbol{x}_r) + \bar{\boldsymbol{u}}_r \in \mathcal{U} \ominus \boldsymbol{K}\phi$ and $\bar{\boldsymbol{y}} \in \mathcal{Y} \ominus \bar{\boldsymbol{D}}_1\boldsymbol{K}\phi \ominus \mathcal{W}_2$. Hence, the last element in $\hat{\boldsymbol{u}}^{[0]}_{i,0:N-1}(t+1)$ and the last element in $\hat{\boldsymbol{x}}^{[0]}_{i,0:N}(t+1)$ satisfy the corresponding constraints in (16), $\forall i \in \mathbb{I}_{1:k}$. Thus $\hat{\boldsymbol{z}}^{[0]}(t+1)$ is feasible for $\mathbb{P}_{i,N}(t+1)$, $\forall i \in \mathbb{I}_{1:k}$.

In the iteration process, assuming that the convergence is reached at the $\bar{p}$-th iteration, $\left\{\boldsymbol{z}_1'^{[\bar{p}+1]}, \ldots, \boldsymbol{z}_k'^{[\bar{p}+1]}\right\}$ are feasible obviously. And then $\boldsymbol{z}^{[\bar{p}+1]} = \left\{\bar{\boldsymbol{x}}^{[\bar{p}+1]}, \bar{\boldsymbol{u}}^{[\bar{p}+1]}_{0:N-1}\right\} = \left\{\bar{\boldsymbol{x}}'^{[\bar{p}+1]}, \bar{\boldsymbol{u}}'^{[\bar{p}+1]}_{0:N-1}\right\}$ are feasible. As a result, the converged simplified solution $\boldsymbol{z}^*(t+1)$ is feasible. Thus, the optimization for the nominal system is recursively feasible. According to (16e), (16f), (16g) and (18), it has $\boldsymbol{x}_i \in \mathcal{X}_i$, $\boldsymbol{u}_i \in \mathcal{U}_i$ and $\boldsymbol{y}_i \in \mathcal{Y}_i$, $\forall i \in \mathbb{I}_{1:k}$ when $\boldsymbol{z}$ is feasible. Thus, the proposed EC-DMPC algorithm is recursively feasible. □

Take the objective function $V_N(\cdot)$ as Lyapunov function and the Lemma 3 follows.

Lemma 3 (Convergence of the nominal system)**.** *Let Assumption 2 hold, then for all* $\boldsymbol{x} \in \mathcal{X}_N$,

$$V_N\left(\boldsymbol{x}(t+1), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t+1), \bar{\boldsymbol{u}}^*_{0:N-1}(t+1)\right) \le V_N\left(\boldsymbol{x}(t), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*_{0:N-1}(t)\right) - l\left(\bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*(t)\right). \quad (24)$$

Proof. At sample $t+1$, let take the feasible solution in Lemma 2, then

$$\begin{aligned} &V_N\left(\boldsymbol{x}(t+1), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \hat{\boldsymbol{x}}^{[0]}(t+1), \hat{\boldsymbol{u}}^{[0]}_{0:N-1}(t+1)\right) \\ &= V_N\left(\boldsymbol{x}(t), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*_{0:N}(t)\right) - l\left(\bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*(t)\right) \\ &-V_f\left(\bar{\boldsymbol{x}}^*(t+N|t)\right) + l(\bar{\boldsymbol{x}}(t+N|t), \bar{\boldsymbol{u}}(t+N|t)) \\ &+V_f(\bar{\boldsymbol{x}}(t+N+1|t)). \end{aligned} \quad (25)$$

As $\bar{\boldsymbol{x}}^*(t+N|t) \in \mathcal{X}_f$, it follows Assumption 2 that the sum of the last three terms of (25) is no more than 0, then

$$V_N\left(\boldsymbol{x}(t+1), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \hat{\boldsymbol{x}}^{[0]}(t+1), \hat{\boldsymbol{u}}^{[0]}_{0:N-1}(t+1)\right) \le V_N\left(\boldsymbol{x}(t), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*_{0:N}(t)\right) - l\left(\bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*(t)\right). \quad (26)$$

From (23), it is obviously that

$$V_N\left(\boldsymbol{x}(t+1), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t+1), \bar{\boldsymbol{u}}^*_{0:N-1}(t+1)\right) \le V_N\left(\boldsymbol{x}(t+1), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \hat{\boldsymbol{x}}^{[0]}(t+1), \hat{\boldsymbol{u}}^{[0]}_{0:N-1}(t+1)\right). \quad (27)$$

Combining (26) and (27), we have

$$V_N\left(\boldsymbol{x}(t+1), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t+1), \bar{\boldsymbol{u}}^*_{0:N-1}(t+1)\right) \le V_N\left(\boldsymbol{x}(t), \boldsymbol{x}_r, \boldsymbol{y}_{\min}; \bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{u}}^*_{0:N-1}(t)\right) - l\left(\bar{\boldsymbol{x}}^*(t), \bar{\boldsymbol{x}}^*(t)\right). \quad (28)$$

□

Theorem 1 (Closed-loop stability)**.** *Let Assumptions 1 and 2 hold. Then, for any initial state* $\boldsymbol{x}(t) \in \mathcal{X}_N$*, the closed-loop system based on EC-DMPC algorithm is stable.*

Proof. Obviously, there is a lower bound for $V_N(\cdot)$. With the feasible initial state and input $\boldsymbol{z}$, the value of $V_N(\cdot)$ is also upper bounded. Then, combining the result of Lemma 3, the nominal system can be stabilized. Following Assumption 1, when the nominal system trends to be stable, the original system is closed-loop stable. Combining Lemma 2, the result that for any initial state $\boldsymbol{x}(t) \in \mathcal{X}_N$, the closed-loop system based on EC-DMPC algorithm is stable, is proved. □

As there are constraints for $\boldsymbol{y}$, $\boldsymbol{u}$ and $\boldsymbol{x}$, $\boldsymbol{x}_r$ may be not a feasible solution, so the state $\boldsymbol{x}$ and output $\boldsymbol{y}$ converge to some constants when the system is stable.

5. Experiment result

5.1. Case description

The case used is part of WDNs of Shanghai, which is the largest city in China. This WDN operates as described in Section 2. It covers an area of about 360 km^2 and supplies clean water for about 3.2 million people. There are 18 pump stations totally, including 6 pump stations with reservoirs. One of the pump stations connects to two outlet pipes, so there are 19 input variables. There are 44 sensors used to monitor users' pressure. Besides, x_1 and u_3, x_2 and u_4, x_3 and u_5, x_4 and u_7, x_5 and u_{12}, x_6 and u_{15}, respectively, form a pair. Each pair belongs to the same pump station.

5.2. Partitioning result

The time-series data for every variable used in the partitioning approach have the sampling time of 10 minutes and the length of 800. In Step 1 of Algorithm 1, the UPP method is applied with $n_{max} = 10$. The acceptable clustering results are obtained when the number of subsystems is 3, 4 and 6. The corresponding variance of the number of output variables in the subsystem is 54.9, 9.5 and 2.9, respectively. Because the minimum variance is obtained when

the number of subsystems is 6, thus $k = 6$. The obtained ordinal matrices $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$ and $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{V}}}$ in Step 2 and 3 are as follows:

$$\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}} = \begin{array}{c} \begin{array}{cccccc} S_1 & S_2 & S_3 & S_4 & S_5 & S_6 \end{array} \\ \left[\begin{array}{cccccc} \boxed{2} & 18 & 9 & 16 & \textcircled{1} & 12 \\ 15 & \textcircled{2} & 12 & 9 & 10 & 9 \\ 16 & \textcircled{8} & 17 & 13 & 11 & 15 \\ 19 & \boxed{5} & 18 & \textcircled{4} & 17 & 17 \\ 7 & 15 & \boxed{4} & 17 & 14 & \textcircled{3} \\ \boxed{4} & 17 & 10 & 19 & 18 & \textcircled{1} \\ 18 & \boxed{3} & 14 & 12 & \textcircled{2} & 13 \\ \boxed{3} & 10 & \textcircled{2} & 6 & 8 & \boxed{5} \\ \boxed{5} & 7 & 7 & \textcircled{1} & 13 & 10 \\ 9 & 16 & 15 & \textcircled{5} & 15 & 18 \\ 6 & 6 & 6 & 10 & \textcircled{3} & \boxed{4} \\ 12 & 13 & 11 & 11 & 16 & \textcircled{7} \\ 17 & 19 & \textcircled{16} & 18 & 19 & 19 \\ \textcircled{1} & 12 & \boxed{3} & \boxed{2} & 6 & 8 \\ 10 & 14 & \boxed{5} & 15 & \textcircled{4} & 6 \\ 14 & \textcircled{1} & 19 & 7 & \boxed{5} & 16 \\ 8 & 9 & \textcircled{1} & 8 & 7 & \boxed{2} \\ 13 & 11 & 8 & \textcircled{3} & 12 & 11 \\ 11 & \textcircled{4} & 13 & 14 & 9 & 14 \end{array}\right] \end{array},$$

$$\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{V}}} = \begin{array}{c} \begin{array}{cccccc} S_2(v_1) & S_4(v_2) & S_5(v_4) & S_5(v_6) & S_6(v_3) & S_6(v_5) \end{array} \\ \left[\begin{array}{cccccc} 7 & 13 & 7 & \textcircled{5} & 17 & 9 \\ \textcircled{5} & \boxed{3} & 12 & 13 & 10 & 16 \\ 19 & 9 & \boxed{5} & 9 & 14 & 14 \\ 12 & 19 & 14 & 15 & 18 & 18 \\ 13 & 18 & 18 & 11 & 19 & \textcircled{3} \\ 6 & 17 & 15 & 12 & 9 & 6 \\ \boxed{2} & 16 & 19 & \textcircled{4} & 6 & 12 \\ 8 & 10 & 13 & 10 & \boxed{1} & 15 \\ 14 & 8 & \boxed{4} & \boxed{2} & \boxed{4} & \boxed{5} \\ 11 & \textcircled{5} & \boxed{3} & 7 & 13 & \boxed{2} \\ 9 & \boxed{2} & 8 & \textcircled{1} & \boxed{2} & \boxed{4} \\ 18 & 11 & 17 & 14 & 8 & 19 \\ 17 & 15 & 16 & 18 & 16 & 17 \\ 15 & \boxed{4} & 6 & 8 & \boxed{3} & 7 \\ \boxed{3} & \boxed{1} & \textcircled{1} & 19 & 7 & 8 \\ \textcircled{1} & 6 & \boxed{2} & 6 & 12 & 11 \\ 16 & 14 & 10 & 17 & 11 & \boxed{1} \\ 10 & 12 & 9 & \boxed{3} & \boxed{5} & 13 \\ \textcircled{4} & 7 & 11 & 16 & 15 & 10 \end{array}\right] \end{array}. \quad (29)$$

The threshold values are set to 5 for both matrices. In (29), the ith row of the matrix corresponds to u_i, $\forall i \in \mathbb{I}_{1:n_u}$ according to Algorithm 1. In $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$, if u_i is assigned to S_j, then the corresponding number is circled; if u_i and S_j are strongly coupled, then there is a square outside the corresponding number. For example, the first row of $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{Y}}}$, means that u_1 is allocated to S_5, and it is strongly coupled with S_1. In $\boldsymbol{M}_{\boldsymbol{b}_{\mathbb{V}}}$, if the strong coupling exists inside the subsystem, then the corresponding number is circled; if the strong coupling is between other subsystems, then there is a square outside the corresponding number. Overall, the WDN is decomposed into 6 subsystems as depicted in Fig. 4 in different colors. Due to the confidentiality reasons, this figure only illustrates main topology of the case, and the length of the connecting line is not related to the actual distance. The variables in subsystems and the corresponding color in Fig. 4 are summarized in Table 1.The strong couplings among subsystems are $C_{\boldsymbol{y}_1,u_{1,6,8,9}}$, $C_{\boldsymbol{y}_2,u_{4,7}}$, $C_{\boldsymbol{x}_2,u_{7,15}}$ $C_{\boldsymbol{y}_3,u_{5,14,15}}$, $C_{\boldsymbol{y}_4,u_{14}}$, $C_{\boldsymbol{x}_4,u_{2,11,14,15}}$, $C_{\boldsymbol{y}_5,u_{16}}$, $C_{\boldsymbol{x}_5,u_{3,9,10,16,18}}$, $C_{\boldsymbol{y}_6,u_{8,11,17}}$ and $C_{\boldsymbol{x}_6,u_{8,9,10,11,14,17,18}}$. The structure of the resulted subsystems and the strong couplings are shown in Fig. 5.

It can be seen in Fig. 4 that variables in each subsystem are relatively geographically concentrated. Combining Fig. 4 and Fig. 5, it is shown that strong couplings only exist between subsystems which are geographically adjacent. These results are consistent with our perceptions about WDNs: (1) variables that are geographically close to each other have relatively strong coupling and then should be assigned to the same subsystem. (2) subsystems which are geographically close to each other are more likely to have impacts on each other. Therefore, it can be considered that the resulted subsystem is a proper division based on the proposed partitioning algorithm in this paper.

Table 1
Subsystem partitioning result.

Subsystem	$\mathbb{Y}_i$	$\mathbb{U}_i$	$\mathbb{X}_i$	Color
1	$y_1, y_2, y_7, y_{39}, y_{40}, y_{41}, y_{42}$	u_{14}		Red
2	$y_5, y_{10}, y_{11}, y_{13}, y_{14}, y_{29}, y_{34}$	u_2, u_3, u_{16}, u_{19}	x_1	Orange
3	$y_3, y_6, y_8, y_{15}, y_{18}, y_{20}$	u_8, u_{13}, u_{17}		Yellow
4	$y_4, y_{16}, y_{23}, y_{25}, y_{26}, y_{27}, y_{28}, y_{31}, y_{37}$	u_4, u_9, u_{10}, u_{18}	x_2	Green
5	$y_{12}, y_{17}, y_{22}, y_{24}, y_{36}$	u_1, u_7, u_{11}, u_{15}	x_4, x_6	Blue
6	$y_9, y_{19}, y_{21}, y_{30}, y_{32}, y_{33}, y_{35}, y_{38}, y_{43}, y_{44}$	u_5, u_6, u_{12}	x_3, x_5	Purple

Table 2
RMSE of subsystem models.

	S_1	S_2	S_3	S_4	S_5	S_6	max	Ave
$\text{RMSE}_{\boldsymbol{x}}$(m)	–	0.18	–	0.21	0.12	0.19	0.22	0.17
$\text{RMSE}_{\boldsymbol{y}}$(KPa)	5.57	1.96	3.93	2.64	1.81	2.57	6.81	3.06

Table 3
$\boldsymbol{x}_{min}$ and $\boldsymbol{x}_{max}$ for each reservoir, $\boldsymbol{u}_{min}$ and $\boldsymbol{u}_{max}$ for all the pump stations.

	x_1	x_2	x_3	x_4	x_5	x_6	$u_i, \forall i \in \mathbb{I}_{1:n_u}$
$\boldsymbol{x}_{\text{min}}$(m)	1.00	1.00	1.00	1.00	1.00	1.00	–
$\boldsymbol{x}_{\text{max}}$(m)	6.25	4.60	4.85	5.25	5.10	5.00	–
$\boldsymbol{x}_r$(m)	3.75	2.76	2.91	3.15	3.06	3.00	–
$\boldsymbol{u}_{\text{min}}$(KPa)			–				160
$\boldsymbol{u}_{\text{max}}$(KPa)			–				400

5.3. Model verification

In the identification, the sampling time interval is one hour, and a set of data, including 200 sample time, is used. One day's data are used in verification. The root mean square error (RMSE), $\text{RMSE} = \sqrt{\frac{1}{n}\sum_1^n (y_{i,\text{actual}} - y_{i,\text{model}})^2}$, is used to evaluate the performance of the predictions of the identified models. Where, $y_{i,\text{actual}}$ and $y_{i,\text{model}}$ are the actual value and predicted value respectively, and n is the number of test data. As the subsystems have multiple

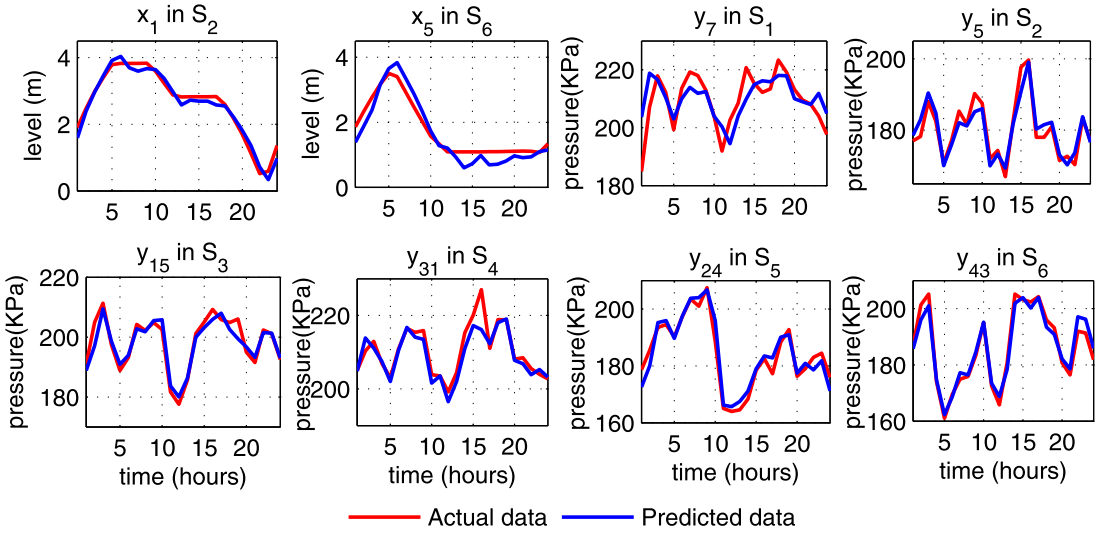


Fig. 7. Actual data and predicted data of some variables in different subsystems.

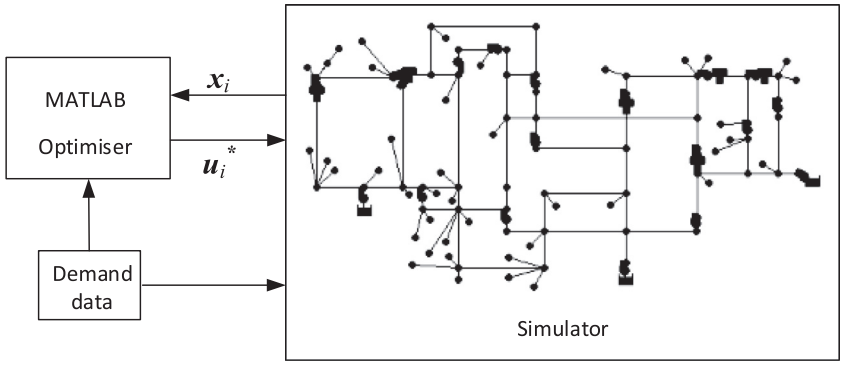


Fig. 8. Experiment platform.

output variables, the arithmetic average of all the variables' RMSE is used to represent the error of the subsystem. The RMSE for each variable is shown in Fig. 6. The average RMSE for each subsystem, maximum RMSE and total average RMSE are gathered in Table 2.

As shown in Table 2, the maximum $\text{RMSE}_{\boldsymbol{x}}$ and $\text{RMSE}_{\boldsymbol{y}}$ of the 6 subsystems are 0.22 m and 6.81 KPa respectively, and the average $\text{RMSE}_{\boldsymbol{x}}$ and $\text{RMSE}_{\boldsymbol{y}}$ are 0.17 m and 3.06 KPa respectively. The maximum level of the reservoir is about 5 m, and the user's pressure is about 200 KPa. The nominal model has some deviations from the real system, but the precision of the identified nominal model could meet the control requirements of WDNs. The results can also be seen in Fig. 7, where the actual data and predicted data for some variables in different subsystems are depicted.

Remark 3. For the model to predict reservoir's level $\boldsymbol{x}$, as $\boldsymbol{x}(t)$ is measurable in the actual operation, and $\boldsymbol{x}(t+1)$ can be calculated using the measured $\boldsymbol{x}(t)$, so in the model verification for $\boldsymbol{x}$, the predict value used is the one step prediction.

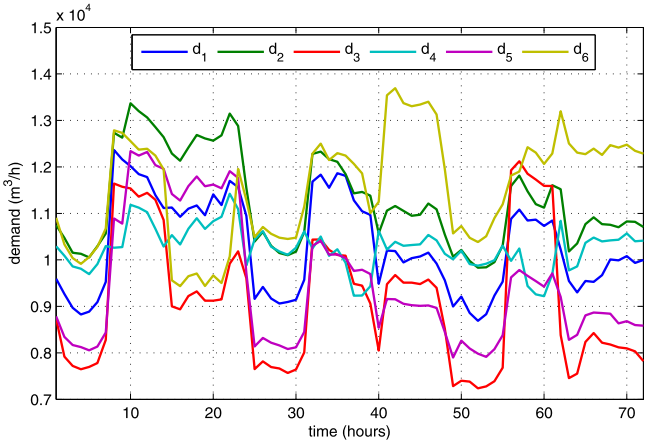


Fig. 9. Three days' real water demand data for each subsystem.

5.4. Pressure optimization results based on EC-DMPC scheme

The experiment platform is based on MATLAB and EPANET, which is a widely used hydraulic software. The structure of the simulation platform is shown in Fig. 8. The optimization toolbox in MATLAB is used for optimization calculations based on the identified nominal model and the water demand data. The calculated optimal control law $\boldsymbol{u}_i^*$, $\forall i \in \mathbb{I}_{1:k}$ is communicated to the hydraulic simulator, EPANET. Then the state variables obtained in the hydraulic simulator are transferred to the optimiser for the optimization of the next sample. The network running in EPANET has been verified with actual data.

The results obtained by applying EC-DMPC strategy are compared with those obtained by applying other methods. The decentralized MPC (De-MPC) strategy is the most widely used method in the relevant literature [12,13]. The actual operation of WDNs is based on the heuristic method, which is marked as Actual here.

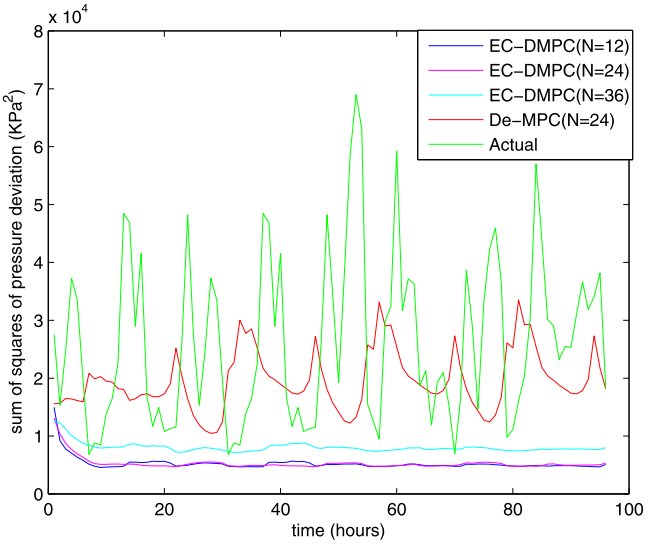


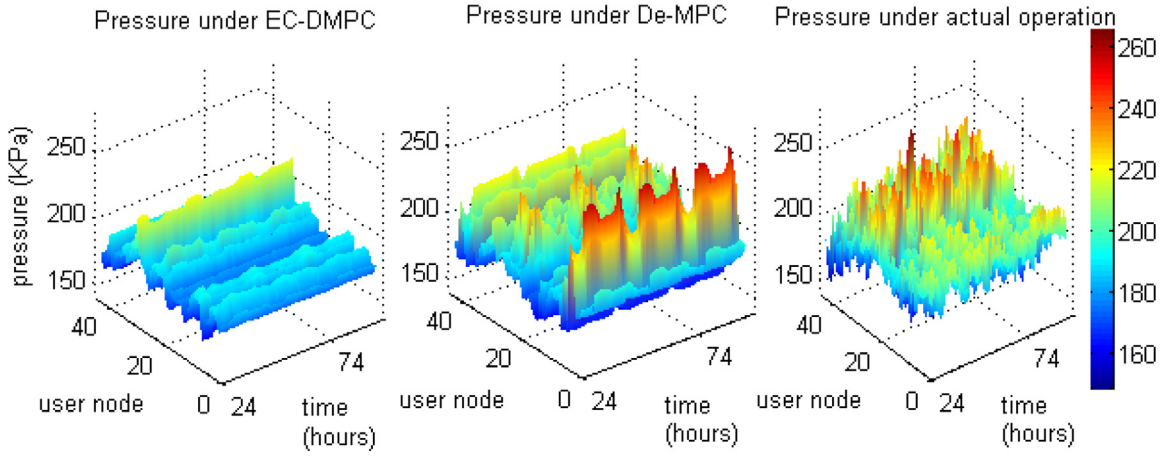
Fig. 10. Global pressure optimization performance of the above 3 methods.

Then the comparison of the results includes two scenarios: (i) in the scenario with actual users' water demand data, the optimization results are compared with the actual operating results and results with applying De-MPC. (ii) in order to further demonstrate the coordination effects of the EC-DMPC strategy, the results between EC-DMPC and De-MPC are compared in a scenario where users' water demand is changed in one subsystem and is constant in all other subsystems.

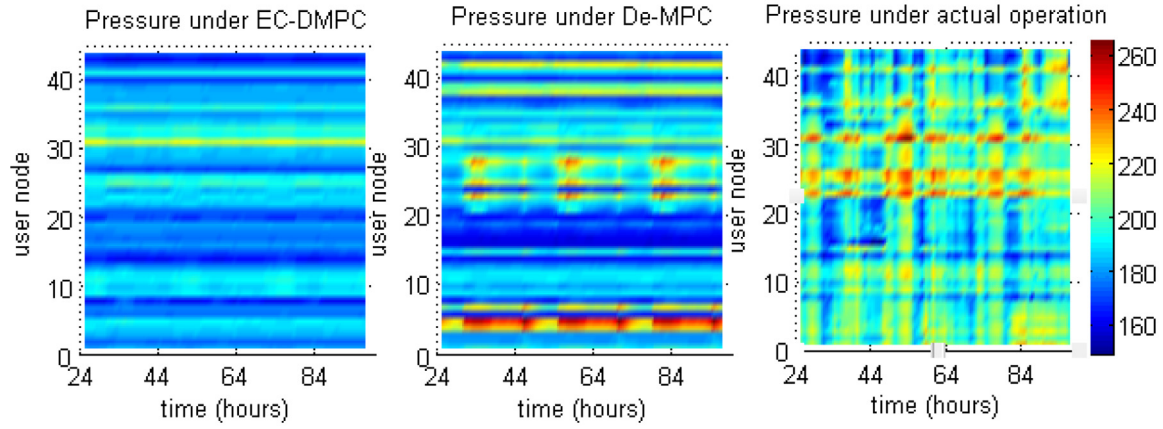
5.4.1. Scenario with actual users' water demand data

In this scenario, all results have been obtained with considering 3 days' actual water demand data, which are shown in Fig. 9. Based on the management criteria of Shanghai, the lower bound for users' pressure is 160 KPa. Regarding the level of reservoirs $\boldsymbol{x}$, the reference value $\boldsymbol{x}_r$ is set as 60% of the maximum level. Constraints for $\boldsymbol{x}$ and $\boldsymbol{u}$ are based on the physical property and summarized in Table 3. Weighting terms for pressure $\boldsymbol{y}$ and reservoir's level $\boldsymbol{x}$ are set as 1 and 0.1 respectively. The sample time is 1 h. The disturbance set is selected as the maximum value of the deviations between the model's one-step prediction and the actual data. As there is a transient of system behavior in the first simulation day, so the first day's water demand data are repeated once in the experiment, which means the experiment lasts 4 days together.

The global pressure optimization objective is used to demonstrate the performance of these three methods. In the EC-DMPC strategy, the optimization horizon N is set to 12, 24 and 36, respectively, while the optimization horizon N is set to 24 in the De-MPC strategy. The performance with different strategies is shown in Fig. 10. As the goal of the optimization problem is to minimize this objective, the performance of EC-DMPC is better than those of two other methods. The performance of actual operation follows



(a) Comparison results in 3D view.



(b) Comparison results in 2D view.

Fig. 11. User's pressure ($\boldsymbol{y}$) under EC-DMPC, De-MPC and actual operation.

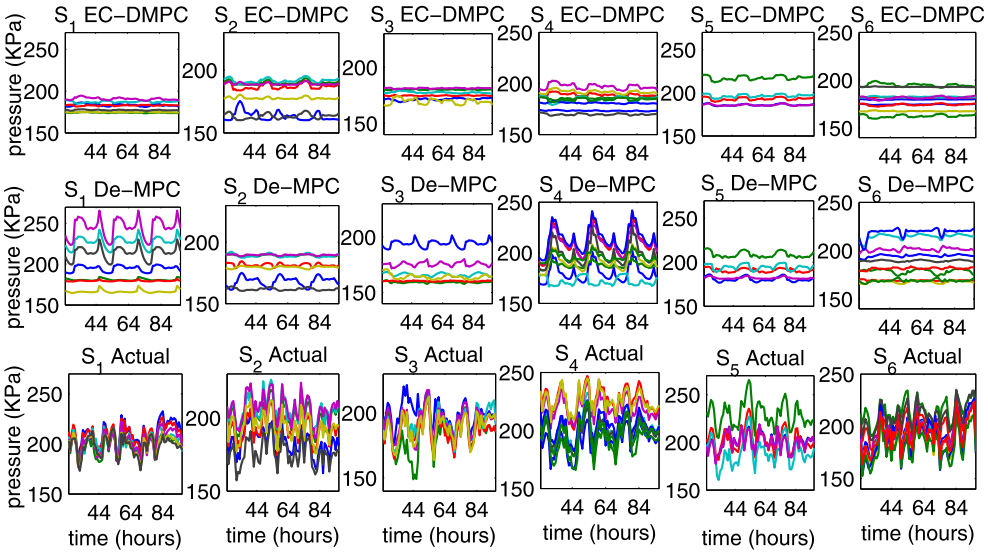


Fig. 12. Users' pressure ($\boldsymbol{y}_i$) in each subsystem under 3 methods, where lines of the same color represent the same variable for each subsystem.

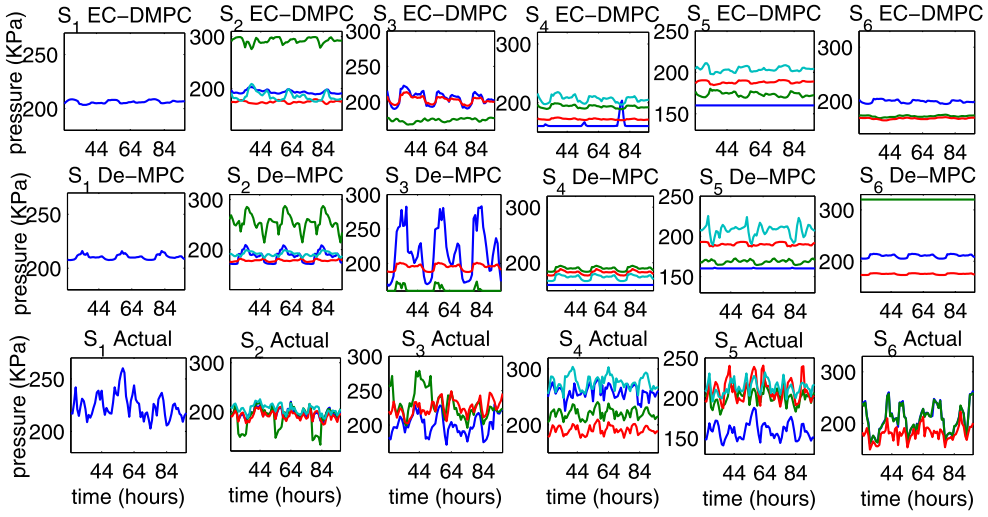


Fig. 13. Pump station's outlet pressure ($\boldsymbol{u}_i$) in each subsystem under 3 methods, where lines of the same color represent the same variable for each subsystem.

the performance of De-MPC. For the EC-DMPC scheme with different N, all results are convergent, but the relationship between the optimization result and N is not clear because of the disturbance caused by weakly coupled neighbours. Moreover, with the increase of N, the computation time becomes longer. When $N = 12$, $N = 24$ and $N = 36$, the total computation time for EC-DMPC strategy is 293.9 s, 536.7 s and 921.4 s, respectively. Considering the system performance and calculation time results at $N = 24$ are used for the subsequent analysis. The total computation time of De-MPC when $N = 24$ is 420.5 s. Although the computation time increases for EC-DMPC compared with De-MPC, the EC-DMPC strategy can still meet the requirements of real-time control because the sampling time is 1 h. The detailed comparison of the results of the 3 methods follows.

With the application of EC-DMPC and De-MPC, the average pressure data for all subsystems from day 2 to day 4, the average, minimum and maximum pressure data for the WDNs in the simulator and the variance for all the subsystems' pressure are gathered in Table 4. This table also presents the corresponding statistical data under actual operation. It is noticed from Table 4 that the results obtained by employing EC-DMPC strategy is better than those of other methods, because the overall average pressure is reduced by more than 7% compared with the actual operation results (from 195.7 KPa to 182.1 KPa), the maximum pressure is reduced from 265.5 KPa to 220.7 KPa, the minimum value is 158.2 KPa, which is close to the lower bound, and the variance of the pressure is reduced significantly. The results obtained by employing De-MPC strategy improve in some ways, such as the average pressure is reduced (from 195.7 KPa to about 188.5 KPa), the minimum value is 157.7 KPa, but the maximum value is 265.8 KPa, which even exceeds the maximum value under actual operation. The results of all subsystems also show these conclusions. Every subsystem's performance is improved in almost the same range under EC-DMPC strategy by the coordination among subsystems. However, in the results of the application of De-MPC, the improvement amplitude of each subsystem is different, and the result of S_1 is even worse. All the users' pressure data during the last 3 days with these 3 methods are depicted in Fig. 11. And the pressure situation in each subsystem for the last 3 days is shown in Fig. 12. It can be seen that users' pressure fluctuates with the change of water demand, it

Table 4
Statistical data of users' pressure.

	day2(KPa)			day3(KPa)			day4(KPa)			variance		
	EC-DMPC	De-MPC	Actual	EC-DMPC	De-MPC	Actual	EC-DMPC	De-MPC	Actual	EC-DMPC	De-MPC	Actual
S_1	181.7	201.2	200.2	181.4	201.2	201.1	181.3	201.2	206.7	0.8	38.0	99.3
S_2	180.3	179.9	195.0	181.1	179.8	196.3	180.5	179.8	191.8	4.1	2.9	79.2
S_3	175.8	170.1	186.7	175.7	169.8	191.7	175.6	169.7	193.3	1.2	3.5	121.4
S_4	184.2	193.8	202.6	184.0	195.9	209.0	183.8	196.1	208.9	2.5	78.8	115.9
S_5	195.2	191.4	201.8	195.7	191.4	208.3	196.2	191.4	203.8	2.5	6.8	132.7
S_6	179.2	188.0	189.1	179.3	188.7	197.6	179.4	188.8	199.5	0.8	4.3	185.9
Ave	182.1	187.9	195.7	182.1	188.5	200.7	182.2	188.5	201.0	1.9	24.8	126.0
Max	220.9	265.8	244.5	220.7	265.8	265.5	220.9	265.8	251.0	13.0	162.5	318.5
Min	158.2	158.3	149.1	158.4	157.7	158.5	158.3	157.8	158.2	0.1	0.1	74.7

Table 5
Statistical data of pump station's outlet pressure.

	day2(KPa)			day3(KPa)			day4(KPa)			Variance		
	EC-DMPC	De-MPC	Actual	EC-DMPC	De-MPC	Actual	EC-DMPC	De-MPC	Actual	EC-DMPC	De-MPC	Actual
S_1	205.7	210.4	223.9	206.4	210.4	228.4	206.8	210.5	218.9	2.1	4.9	140.9
S_2	212.6	202.5	197.7	211.2	202.4	196.9	210.3	202.4	192.6	29.0	108.6	186.0
S_3	198.1	191.5	221.1	193.5	191.2	216.3	192.6	191.5	220.1	36.4	427.6	241.8
S_4	183.8	176.4	235.4	182.7	176.4	239.0	183.1	176.4	234.6	21.6	10.0	105.2
S_5	181.2	181.6	194.1	181.3	181.7	199.4	181.4	181.7	194.2	4.6	15.3	111.5
S_6	179.6	235.3	194.2	179.9	235.3	201.9	180.1	235.3	202.4	4.1	4.0	594.8
Ave	191.9	196.5	209.4	190.9	196.4	221.8	190.7	196.5	209.0	18.1	96.9	224.3

is more homogeneous, and the average pressure decreases with the application of EC-DMPC. It is consistent with the statistical analysis results of pressure data.

Acting as the pressure source in WDNs, the output pressure of the pump station should decrease with the decrease in users' pressure. Table 5 gathers the statistical data of the pump station's outlet pressure from day2 to day4 based on the 3 methods. It is noticed from this table that the pump station's outlet pressure is reduced about 9% with the application of EC-DMPC (the average value is reduced from more than 209.0 KPa to less than 191.9 KPa). With the application of EC-DMPC, part of the pump station's outlet pressure increases, and other decreases. The overall result is that the average outlet pressure of pump stations is reduced. Fig. 13, which describes the pump station's outlet pressure of the last 3 days in each subsystem, illustrates this result intuitively. Pump station's outlet pressure also fluctuates with the change in water demand. The variance is smaller with the application of EC-DMPC than those by applying other methods. These results correspond to the results of users' pressure.

With the optimization of pressure, the level of the reservoir also fluctuates with the change of users' water demand. Reservoirs' levels with 3 methods for the last 3 days are depicted in Fig. 14. It is noticed from the figure that the constraints are satisfied at all time by using EC-DMPC and De-MPC. With the application of EC-DMPC, the level of the reservoir fluctuates with less deviation, which could provide emergency capacity.

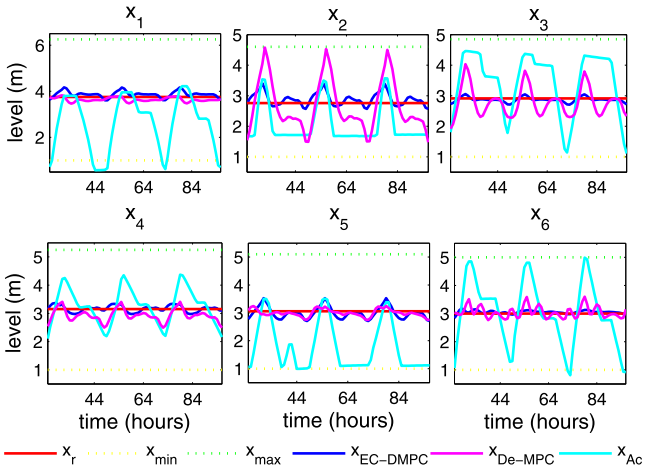


Fig. 14. Reservoir's level (x_i) under 3 methods.

Table 6
Variance of pressure values of all users at $t = 60$ and $t = 120$.

Strategy	$Var_{t=60}(KPa^2)$	$Var_{t=120}(KPa^2)$	Chang ratio
EC-DMPC	133.4	121.1	9.2%
De-MPC	251.5	237.2	5.7%

5.4.2. Scenario with users' water demand changed in one subsystem

In this scenario, at the 72nd sampling time, we change the water demand in S_2 from 10000 m^3/h to 12000 m^3/h, and the users' water demand of all the other subsystems is maintained at 10000 m^3/h. The settings for other parameters are the same as before. The experiment lasts for 5 days. With the application of EC-DMPC strategy and De-MPC method, the users' pressure ($\boldsymbol{y}_i$) and pump station's outlet pressure ($\boldsymbol{u}_i$) for each subsystem are shown in Fig. 15 and 16, respectively. The variance of the pressure values of all users at one moment is used to reflect the degree of pressure homogenization under different methods. The smaller the variance, the more homogeneous the pressure. The variance data for $t = 60$ and $t = 120$ is gathered in Table 6. It is shown in Fig. 16 that when users' water demand in S_2 changes, not only the inputs of S_2 but also the inputs of other subsystems change as well. With EC-DMPC strategy, inputs in subsystems S_4 and S_5, which are strongly coupled with S_2, change with larger magnitude. The performance of EC-DMPC is better than that of De-MPC because the variance of EC-DMPC is smaller than that of De-MPC before and after the change in Table 6. Furthermore, when the water demand of S_2 changes, the global performance is improved by the coordination of the neighbouring subsystems with the application of EC-DMPC strategy because the change ratio (9.2%), which is calculated as $(Var_{t=60} - Var_{t=120})/Var_{t=60}$, is greater than the case with De-MPC strategy (5.7%).

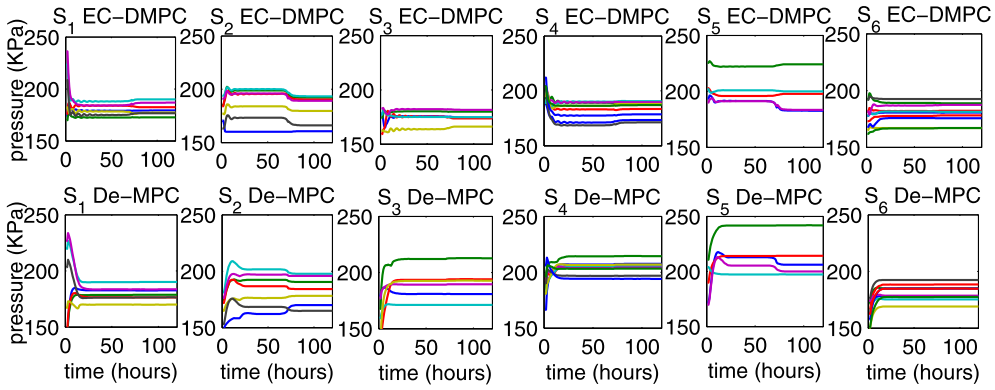


Fig. 15. User's pressure ($\boldsymbol{y}_i$) in each subsystem with a variation demand for S_2, where lines of the same color represent the same variable for each subsystem.

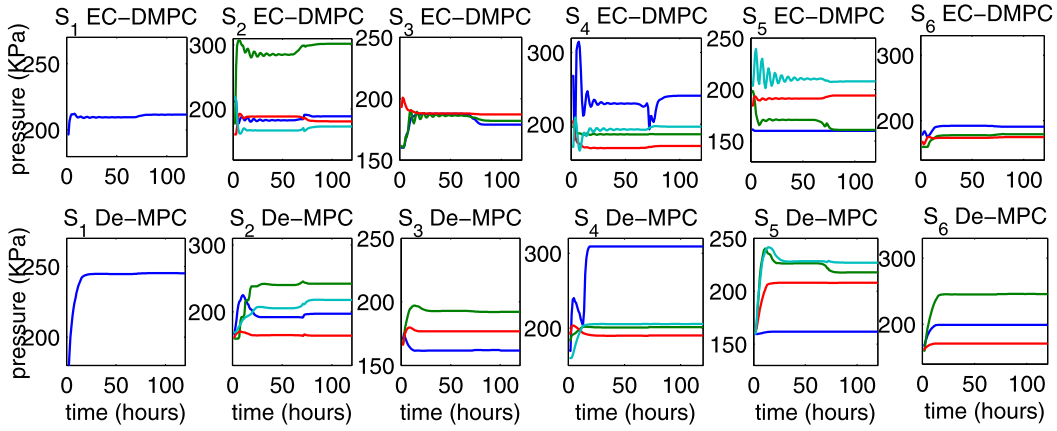


Fig. 16. Pump station's outlet pressure ($\boldsymbol{u}_i$) in each subsystem with a variation demand for S_2, where lines of the same color represent the same variable for each subsystem.

6. Conclusion

In this paper, an enhancing cooperative model predictive control strategy is proposed to optimize the supply pressure to users in water distribution networks. The pressure optimization goal is to maintain the user's pressure homogeneous with lower average value, and the user's water demand should also be guaranteed. The studied large-scale WDN is a common type in most cities in China, where the reservoir is an essential component in the pressure optimization problem. In the modeling of WDNs, a new model structure is given with the analysis of the WDNs' mechanism. The variables in the model are assigned to non-overlapping subsystems by using the proposed partitioning method, which is based on the clustering method and the CCA result of the operation data of variables. The EC-DMPC algorithm, which uses neighbourhood optimization for coordination and robust invariant set control law to handle the disturbance caused by the weakly coupled neighbours, is employed to solve the pressure optimization problem. The closed-loop stability of the algorithm is guaranteed. The EC-DMPC scheme is applied in an actual case, which is part of Shanghai's WDNs. The results have shown that the proposed strategy can make users' pressure homogeneous with smaller average value, the pump station's outlet pressure decreases along with the users' pressure, and the level of the reservoir is kept within the constraints. The reduction of pressure in WDNs is of considerable significance in reducing leakages and energy consumption in practice.

Declaration of Competing Interest

None.

Acknowledgments

This work has received funding from the National Natural Science Foundation of China (Grant/Award Number: 61590924, 61673273, 61833012). The authors thank the anonymous reviewers for the comments, which helped to improve the work.

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.jprocont.2019.09.009.

References

[1] G. Cembrano, J. Quevedo, V. Puig, R. Perez, J. Figueras, G. Ramon, P. Rodriguez, G. Barnet, M. Casas, J. Verdejo, et al., First results of predictive control application on water supply and distribution in santiago-chile, IFAC Proc. Vol. 38 (1) (2005) 25–30.
[2] A. Lambert, What do we know about pressure: Leakage relationships in distribution systems? in: Proc. IWA System Approach to Leakage Control and Water Distribution Systems Management, 2001.
[3] P.D. Dai, Q.C. Le, B.V. Dai, Optimal pump scheduling to pressure management for large-scale water distribution systems, in: International Conference on Advanced Engineering Theory and Applications, 2017, pp. 532–541.
[4] B. Ulanicki, P.L.M. Bounds, J.P. Rance, L. Reynolds, Open and closed loop pressure control for leakage reduction, Urban Water 2 (2) (2000) 105–114.

[5] D. Liu, Y. Zheng, J. Wu, S. Li, Zone model predictive control for pressure management of water distribution network, Asian J. Control (2019).
[6] Y. Wang, T. Alamo, V. Puig, G. Cembrano, Periodic economic model predictive control with nonlinear-constraint relaxation for water distribution networks, in: Control Applications, 2016.
[7] Y. Wang, V. Puig, G. Cembrano, Non-linear economic model predictive control of water distribution networks, J. Process Control 56 (2017a) 23–34.
[8] Y. Wang, J.R. Salvador, D.M.D.L. Pena, V. Puig, G. Cembrano, Periodic nonlinear economic model predictive control with changing horizon for water distribution networks, in: Ifac World Congress, 2017b.
[9] V.P. Ye Wang Teodoro Alamo, G. Cembrano, Economic model predictive control with nonlinear constraint relaxation for the operational management of water distribution networks, Energies (2018).
[10] S. Leirens, C. Zamora, R.R. Negenborn, B. De Schutter, Coordination in urban water supply networks using distributed model predictive control, in: American Control Conference, 2010, pp. 3957–3962.
[11] V. Fambrini, C. Ocampo-Martinez, et al., Modelling and Decentralized Model Predictive Control of Drinking Water Networks: the Barcelona Case Study, Technical Report, 2009.
[12] C. Ocampo-Martinez, D. Barcelli, V. Puig, A. Bemporad, Hierarchical and decentralised model predictive control of drinking water networks: application to barcelona case study, IET Control Theory Appl. 6 (1) (2012) 62–71.
[13] C. Ocampo-Martinez, S. Bovo, V. Puig, Partitioning approach oriented to the decentralised predictive control of large-scale systems, J. Process Control 21 (5) (2011) 775–786.
[14] S. Li, Y. Zheng, Z. Lin, Impacted-region optimization for distributed model predictive control systems with constraints, IEEE Trans. Autom. Sci. Eng. 12 (4) (2015) 1447–1460.
[15] R.R. Negenborn, B.D. Schutter, J. Hellendoorn, Multi-agent model predictive control for transportation networks: serial versus parallel schemes, Int. Fed. Autom. Control 21 (3) (2006) 353–366.
[16] S. Li, Y. Zheng, Distributed Model Predictive Control for Plant-Wide Systems, John Wiley & Sons, 2016.
[17] Y. Zheng, S. Li, R. Tan, Distributed model predictive control for on-connected microgrid power management, IEEE Trans. Control Syst. Technol. 26 (3) (2018) 1028–1039.
[18] P. Stadler, A. Ashouri, F. Marechal, Distributed model predictive control of energy systems in microgrids, in: Systems Conference, 2016.
[19] B.T. Stewart, A.N. Venkat, J.B. Rawlings, S.J. Wright, G. Pannocchia, Cooperative distributed model predictive control, Syst. Control Lett. 59 (8) (2010) 460–469.
[20] Y. Zhang, S. Li, Networked model predictive control based on neighbourhood optimization for serially connected large-scale processes, J. Process Control 17 (1) (2007) 37–50.
[21] M. Brdys, B. Ulanicki, Operational Control of Water Systems: Structures, Algorithms and Applications, Prentice Hall, 1994.
[22] Z. Ren, S. Li, Short-term demand forecasting for distributed water supply networks: a multi-scale approach, in: Intelligent Control and Automation, 2016, pp. 1860–1865.
[23] G. Ji, J. Wang, Y. Ge, H. Liu, Urban water demand forecasting by LS-SVM with tuning based on elitist teaching-learning-based optimization, in: Chinese Control and Decision Conference, 2014, pp. 3997–4002.
[24] A.K. Sampathirao, J.M. Grosso, P. Sopasakis, C. Ocampo-Martinez, A. Bemporad, V. Puig, Water demand forecasting for the optimal operation of large-scale drinking water networks: the barcelona case study, IFAC Proc. Vol. 47 (3) (2014) 10457–10462.
[25] S. Wold, M. Sjostrom, L. Eriksson, Pls-regression: a basic tool of chemometrics, Chemom. Intell. Lab. Syst. 58 (2) (2001) 109–130.
[26] A.D. Nardo, G.F. Santonastaso, S. Venticinque, An automated tool for smart water network partitioning, Water Resour. Manage. 27 (13) (2013) 4493–4508.
[27] A.D. Nardo, M.D. Natale, R. Greco, G.F. Santonastaso, Ant algorithm for smart water network partitioning, Procedia Eng. 70 (2014) 525–534.
[28] A.D. Nardo, M.D. Natale, C. Giudicianni, R. Greco, G.F. Santonastaso, Water supply network partitioning based on weighted spectral clustering, in: International Workshop on Complex Networks and their Applications, 2016, pp. 797–807.
[29] L. Li, L. Xiong, O. Xu, S. Hu, H. Su, Decomposition-based modeling algorithm by CCA-PLS for large scale processes, in: American Control Conference, 2015, pp. 1321–1326.
[30] T.W. Liao, Clustering of time series data-a survey, Pattern Recognit. 38 (11) (2005) 1857–1874.
[31] J. Paparrizos, L. Gravano, k-shape: Efficient and accurate clustering of time series, in: ACM SIGMOD International Conference on Management of Data, 2015, pp. 1855–1870.
[32] P.J. Rousseeuw, Silhouettes: a graphical aid to the interpretation and validation of cluster analysis, J. Comput. Appl. Math. 20 (1987) 53–65.
[33] D.Q. Mayne, M.M. Seron, S.V. Rakovic, Robust model predictive control of constrained linear systems with bounded disturbances, Automatica 41 (2) (2005) 219–224.
[34] D.P. Bertsekas, J.N. Tsitsiklis, Parallel and Distributed Computation: Numerical Methods, 23, Prentice Hall, Englewood Cliffs, NJ, 1989.
[35] A.N. Venkat, I.A. Hiskens, J.B. Rawlings, S.J. Wright, Distributed MPC strategies with application to power system automatic generation control, IEEE Trans. Control Syst. Technol. 16 (6) (2008) 1192–1206.