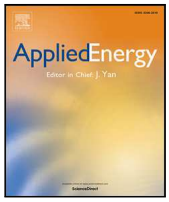
Applied Energy 359 (2024) 122588

Contents lists available at ScienceDirect

Applied Energy

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



Decentralized optimization of energy-water nexus based on a mixed-integer boundary compatible algorithm☆

Santosh Sharma, Qifeng Li *

Department of Electrical and Computer Engineering, University of Central Florida, Orlando, FL, United States

ARTICLE INFO

Keywords:
Coordinated optimal power flow and pump scheduling
Decentralized optimization with mixed-integer boundary variables
Energy-water nexus

ABSTRACT

The electric power distribution system (PDS) and the water distribution system (WDS) are coupled with each other through electricity-driven water facilities (EdWFs), such as pumps, water desalination plants, and wastewater treatment facilities. However, they are generally owned and operated by different utilities, and there does not exist an operator that possesses full information of both systems. As a result, centralized methods are not applicable for coordinating the operation of the two systems. This paper proposes a decentralized framework where the PDS and WDS operators solve their own operation problems, respectively, by sharing only limited information. Nevertheless, the boundary variables (i.e., the variables shared between two systems) are discontinuous due to their dependence on the on/off nature of EdWFs. Unfortunately, mature decentralized/distributed optimization algorithms like the alternating direction method of multipliers (ADMM) cannot guarantee convergence and optimality for a case like this. Therefore, this paper develops a novel algorithm that can guarantee convergence and optimality for the decentralized optimization of PDS and WDS based on a recently developed algorithm called the SD-GS-AL method. The SD-GS-AL method is a combination of the simplicial decomposition (SD), gauss–seidel (GS), and augmented Lagrangian (AL) methods, which can guarantee convergence and optimality for mixed-integer programs (MIPs) with continuous boundary variables. Nonetheless, the original SD-GS-AL algorithm does not work for the PDS-WDS coordination problem where the boundary variables are discontinuous. This paper modifies and improves the original SD-GS-AL algorithm by introducing update rules to discontinuous boundary variables (called the Auxiliary Variables Update step). The proposed mixed-integer boundary compatible (MIBC) SD-GS-AL algorithm has the following benefits: (1) it is capable of handling cases whose boundary variables are discontinuous with convergence and optimality guaranteed for mild assumptions, and (2) it only requires limited information exchange between PDS and WDS operators, which will help preserve the privacy of the two utilities and reduce the investment in building additional communication channels. Simulations on two coupled PDS and WDS test cases (Case 1: IEEE-13 node PDS and 11-node WDS, and Case 2: IEEE-37 node PDS and 36-node WDS) show that the proposed MIBC algorithm converges to the optimal solutions while the original SD-GS-AL does not converge for both test cases. The ADMM does not converge for the first test case while it converges to a sub-optimal solution, 63 % more than the optimal solution for the second test case.

1. Introduction

Power operators generally need to adjust the control settings of power systems, e.g., power generation, periodically, i.e., every 5 minutes to one hour, to meet the time-varying electricity demands and other operating conditions. Ideally, power operators use optimization technology to determine the periodic adjustments of power generation in order to reduce operational costs, which is generally referred to as optimal power flow (OPF) [1]. An analogous process in the water sector is called optimal pump scheduling (OPS) [2,3]. In this problem, water system operators pursue an operation schedule of pumps that can satisfy the requirement of water supply with the objective of minimizing the electricity cost for operating the pumps. As a matter of fact, OPF and OPS are two coupled problems since the power distribution system (PDS) and water distribution system (WDS) are interconnected via electricity-driven water facilities (EdWF). However, the two problems are solved independently by the power and water operators, respectively.

☆ This work is supported by the U.S. National Science Foundation under Award CBET#2124849.
* Corresponding author.
E-mail address: qifeng.li@ucf.edu (Q. Li).

https://doi.org/10.1016/j.apenergy.2023.122588
Received 29 September 2023; Received in revised form 10 December 2023; Accepted 27 December 2023
Available online 18 January 2024

Nomenclature

Sets: OPS

$\mathcal{E}_{\mathrm{W}}^{\mathrm{P}}$	Set of pipes with a pump installed
$\mathcal{N}_{\mathrm{W}}, \mathcal{E}_{\mathrm{W}}$	Node, edge sets of water network
$\mathcal{N}_{\mathrm{w}}^{\mathrm{T}}$	Set of nodes connected to a tank

Sets: MIBC SD-GS-AL

$\mathcal{K}_k$	Set of iteration k, $\{1, 2, \dots k\}$
$\mathcal{M}$	Set of iteration k with non-zero $\boldsymbol{y}$
$\mathcal{X}_{\mathrm{p}}$	Feasible set of OPF
$\mathcal{X}_{\mathrm{w}}$	Feasible set of OPS
$\mathrm{CH}(.)$	Convex hull

Parameters: OPF

Δt	Duration of a time period
$\overline{\ell}_{ik}$	Squared of current carrying capacity of line ik
$\overline{S}_{ik}$	MVA limit of line ik
$\underline{v}_i, \overline{v}_i$	Minimum and maximum voltage limits
$c_{i,t}, \boldsymbol{c}$	Grid electricity price
E_i^{Int}	Initial SoC of ESS at node i
$E_i^{\mathrm{Rat}}, S_i^{\mathrm{Rat}}$	SoC and MVA capacity of ESS at node i
$p_{i,t}^{\mathrm{Load}}, q_{i,t}^{\mathrm{Load}}$	Active, reactive power demand at node i
$p_{i,t}^{\mathrm{l}}, q_{i,t}^{\mathrm{l}}$	Active, reactive power demand of water pump at node i
$p_{i,t}^{\mathrm{PV}}, q_{i,t}^{\mathrm{PV}}$	Active, reactive power output from solar PV at node i
$r_i^{\mathrm{Batt}}, r_i^{\mathrm{Cvt}}$	Resistances of battery and converter in ESS
r_{ik}, x_{ik}	Resistance and reactance of line ik

Parameters: OPS

$\underline{f}_i^{\mathrm{R}}, \overline{f}_i^{\mathrm{R}}$	Minimum, maximum water injection limits of water source at node i
$\underline{f}_i^{\mathrm{T}}, \overline{f}_i^{\mathrm{T}}$	Minimum, maximum water flow limits of the water tank at node i
$\underline{f}_k, \overline{f}_k$	Minimum, maximum water flow of pipe k
$\underline{S}_i^{\mathrm{w}}, \overline{S}_i^{\mathrm{w}}$	Minimum, maximum volume of water tank at node i
$\underline{z}_i, \overline{z}_i$	Minimum, maximum head gain limits at node i
A_i	Cross-sectional area of the water tank at node i
$a_{1,k}, a_{0,k}$	Characteristics coefficients of pump at pipe k
$f_{i,t}^{\mathrm{D}}$	Water demand at node i
h_i	Elevation at node i
M	Big-M parameter
R_k^{P}	Head loss coefficient of pipe k
$S_{i,0}^{\mathrm{w}}$	Initial volume of water in a tank at node i

Parameters: MIBC SD-GS-AL

ϵ	Convergence tolerance
ρ	Penalty parameter
J	Maximum number of inner loop iterations
K	Maximum number of outer loop iterations

Variables: OPF

$\ell_{ik,t}$	Squared of current flow on line ik at time t
$p_{i,t}^{\mathrm{ES}}, q_{i,t}^{\mathrm{ES}}$	Active, reactive power output of ESS at node i
$p_{i,t}^{\mathrm{G}}, q_{i,t}^{\mathrm{G}}$	Active and reactive power from grid at node i
$p_{i,t}^{\mathrm{Loss}}$	Active power loss in ESS at node i
$p_{ik,t}, q_{ik,t}$	Active, reactive power flow on line ik
$v_{i,t}$	Squared of voltage at node i at time t

Variables: OPS

$\alpha_{k,t}$	Pump on/off status variable at pipe k
$f_{i,t}^{\mathrm{R}}$	Water flow injected from water source at node i
$f_{i,t}^{\mathrm{T}}$	Water flow to water tank at node i
$f_{ik,t}/f_{k,t}$	Water flow in pipe ik/k
$p_{k,t}^{\mathrm{p}}$	Power consumed by pump at pipe k
w	Virtual objective function
$z_{i,t}$	Water head at node i
$z_{k,t}^{\mathrm{R}}$	Water head imposed by pump at pipe k

Variables: MIBC SD-GS-AL

$\breve{\varphi}_{\mathrm{p}}$	Lagrangian lower bound of power subproblem
$\breve{\varphi}_{\mathrm{w}}$	Lagrangian lower bound of water subproblem
$\hat{\varphi}_{\mathrm{p}}$	Lagrangian upper bound of power subproblem
$\hat{\varphi}_{\mathrm{w}}$	Lagrangian upper bound of water subproblem
$\lambda_{\mathrm{p}} \backslash \lambda_{\mathrm{w}}$	Lagrangian multipliers
LR_{p}	Value of Lagrangian relaxation of power subproblem
LR_{w}	Value of Lagrangian relaxation of water subproblem
$y_{n,t}$	Auxiliary variable

Ignoring the coupling of PDS and WDS may result in conflicting solutions of OPF and OPS, which may increase the cost and/or risk [4–6]. More and more researchers have realized this issue in recent years and considered PDS and WDS as a coupled system which is referred to as the distribution-level energy-water nexus (DEWN) [7,8]. For example, [7] investigated the optimal operation problem of DEWN by combining the OPF and OPS problems into one that is called optimal power-water flow in [9]. Nevertheless, the original DEWN optimization problem is a large-scale mixed-integer nonlinear program (MINLP), which is computationally intractable to solve. A quasi-convex hull relaxation is developed in [8] to convexify the MINLP model into a mixed-integer convex programming (MICP) model, which is much easier to solve. The MINLP model of DEWN optimization is linearized in [10] to further improve the computational efficiency with the cost of a reduction in model accuracy.

Moreover, the aggregation of EdWFs is considered a virtual power plant and virtual energy storage to provide demand response services to the power systems in [11–21]. The water booster pressure systems are modeled as flexible loads for demand response in [22]. In [23], authors investigated a market-clearing mechanism in a co-optimization model that coordinates the operation of grid-connected reverse osmosis water desalination plants and renewable-rich power systems for demand response. Optimal placement of pumps-as-turbines and demand response through water storage tanks is proposed in [24]. Ref. [25] proposed an analytical model for quantifying the interdependence between the

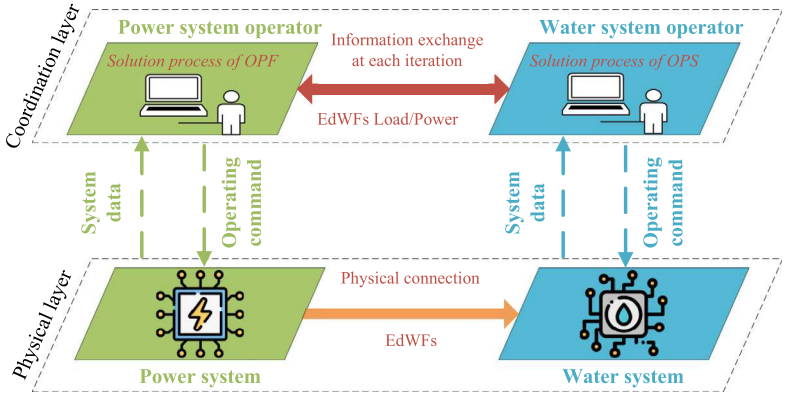


Fig. 1. Proposed framework for the decentralized coordination of power and water utilities.

resilience of power and water distribution systems. Restoration of PDS after a disaster considering energy-water-gas interdependency is considered in [26]. The same authors of [26] used a two-stage moment-based distributionally robust approach to capture and represent the uncertainties in renewable generation for managing the water-energy-carbon nexus in [27]. Risk-based two-stage stochastic co-optimization framework for the coordination of renewable-rich power systems and water desalination plants is proposed in [28]. Literature [29] proposed a coordinated restoration framework for a coupled power water system to respond to disruptions using a two-stage risk-averse stochastic programming. Ref. [30] further considered the uncertainty of renewable energy based on the DEWN optimization model developed in [8]. In short, whether it is for demand response, renewable management, or resilience, the existing research converges to a conclusion—it is beneficial to coordinate the operation of PDS and WDS, i.e., OPF and OPS.

However, an important fact was ignored in most of the above-mentioned research: the PDS and WDS in a specific city or region are generally owned and operated by different utilities, i.e., the power and water utilities. In other words, there does not exist an entity that possesses full information of both systems. As a result, the centralized schemes proposed in existing research are not practical. To this end, we propose to coordinate the OPF and OPS under a decentralized framework in this paper. Namely, the OPF and OPS will be solved by power and water operators, respectively, with only a limited amount of information needed to be shared between the two operators, as shown in Fig. 1. The variables that are shared by two subproblems in a decentralized optimization problem are called boundary variables [31] in this paper. In the context of this research, OPF and OPS are the two subproblems, and the boundary variables are the power consumption of EdWFs, e.g., pumps, water desalination plants, and wastewater treatment facilities. It is worth noting that the boundary variables in this decentralized optimization problem are discontinuous due to the on/off characteristics of the EdWFs.

Recently, there have been a number of algorithms for decentralized/distributed optimization in the literature. For example, authors in [9] leveraged the alternating direction method of multipliers (ADMM) [32,33] for the distributed optimization of optimal power-water flow. However, they did not consider the on–off operation of the EdWFs (i.e., binary variables). Moreover, the ADMM requires subproblems to be convex and boundary variables to be continuous for the guarantee of optimality and convergence. Similar issues apply to the benders decomposition method (BDM) leveraged in [34–36]. Moreover, the BDM may not always work when both master and subproblem models contain integer variables. In addition, other mature algorithms, such as the analytical target cascading [37,38], auxiliary problem principle [39], and cutting plane consensus [40] are also proven to converge for simple cases where all subproblems are continuous and convex.

In recent years, other algorithms like the Alternating Direction Inexact Newton (ALADIN) [41] method and the SD-GS-AL method [42], which is the combination of the simplicial decomposition (SD), gauss–seidel (GS), and augmented Lagrangian (AL) methods, are developed and geared for the mixed-integer cases. Namely, some subproblems are mixed-integer programs (MIP) where the integer variables are inside the subproblems. Nevertheless, these algorithms are not guaranteed to converge and be optimal for cases like the PDS-WDS coordinated optimization problem where integer variables are located on the boundary of subproblems. Therefore, this paper proposes a mixed-integer boundary compatible (MIBC) SD-GS-AL algorithm, which can guarantee convergence and optimality for these cases by incorporating update rules to discontinuous boundary variables (called the Auxiliary Variables Update step) in the original SD-GS-AL algorithm. In summary, the main contributions of the paper are given as follows:

1. From the engineering perspective, this research respects the fact that the two systems are operated by different operators and proposes to coordinate the two systems via a new decentralized method, i.e., the MIBC SD-GS-AL algorithm. The proposed method has the following benefits: (a) it is capable of handling mixed-integer boundary variables with convergence and optimality guaranteed for mild assumptions (i.e., (1) the global optimal solution of the coordinated OPF-OPS is unique, and (2) the objective function is linear), (b) it only requires limited information exchange between PDS and WDS operators, which will help preserve the privacy of the two systems and reduce the investment in building communication channels.
2. From the perspective of mathematical method, being different from the original SD-GS-AL algorithm [42], the MIBC SD-GS-AL algorithm guarantees optimality and convergence for MIP subproblems that share discontinuous boundary variables, which is made possible by introducing the Auxiliary Variables Update step to the original SD-GS-AL algorithm. Note that the proposed algorithm is general and applicable to other problems that have similar features.

The rest of the paper is organized as follows: Section 2 describes the decentralized formulation of the coordinated OPF-OPS problem. Section 3 introduces the MIBC SD-GS-AL decentralized algorithm for the coordination of PDS and WDS. Section 4 provides the simulation results of the proposed framework and algorithm. Section 5 provides the conclusion and potential future research.

2. Decentralized formulation of the coordinated OPF-OPS problem

2.1. Original formulation of OPF in PDS

The distributed energy resources (DERs), such as energy storage systems (ESSs) and solar photovoltaics (PVs), and grid power, which

supply water pumps in the water network and other loads in the power distribution network, are modeled in this paper. The resulting distribution OPF, adopted from [8,43], is given as follows:

$$\text{Min.} \sum_i \sum_t c_{i,t} p^{\mathrm{G}}_{i,t} = \boldsymbol{c}^{\top} \boldsymbol{p}^{\mathbf{G}} \tag{1a}$$

S.t.:

$$\left(p_{ik,t}\right)^2 + \left(q_{ik,t}\right)^2 = v_{i,t} \ell_{ik,t} \tag{1b}$$

$$v_{i,t} - v_{k,t} - 2\left(r_{ik} p_{ik,t} + x_{ik} q_{ik,t}\right) + \left(\left(r_{ik}\right)^2 + \left(x_{ik}\right)^2\right) \ell_{ik,t} = 0 \tag{1c}$$

$$0 \le \ell_{ik,t} \le \overline{\ell}_{ik} \tag{1d}$$

$$\left(p_{ik,t}\right)^2 + \left(q_{ik,t}\right)^2 \le \left(\overline{S}_{ik}\right)^2 \tag{1e}$$

$$\left(\underline{v}_i\right)^2 \le v_{i,t} \le \left(\overline{v}_i\right)^2 \tag{1f}$$

$$0 \le E^{\mathrm{Int}}_i - \sum_t \left(p^{\mathrm{ES}}_{i,t} + p^{\mathrm{Loss}}_{i,t}\right) \Delta t \le E^{\mathrm{Rat}}_i \tag{1g}$$

$$\left(p^{\mathrm{ES}}_{i,t}\right)^2 + \left(q^{\mathrm{ES}}_{i,t}\right)^2 \le \left(S^{\mathrm{Rat}}_i\right)^2 \tag{1h}$$

$$\left(r^{\mathrm{Batt}}_i + r^{\mathrm{Cvt}}_i\right)\left(p^{\mathrm{ES}}_{i,t}\right)^2 + r^{\mathrm{Cvt}}_i \left(q^{\mathrm{ES}}_{i,t}\right)^2 = p^{\mathrm{Loss}}_{i,t} v_{i,t} \tag{1i}$$

$$p^{\mathrm{G}}_{i,t} + p^{\mathrm{ES}}_{i,t} + p^{\mathrm{PV}}_{i,t} - p^{\mathrm{Load}}_{i,t} - p^{\mathrm{l}}_{i,t} = \sum_j \left(p_{ji,t} - r_{ji} \ell_{ji,t}\right) + \sum_k p_{ik,t} \tag{1j}$$

$$q^{\mathrm{G}}_{i,t} + q^{\mathrm{ES}}_{i,t} + q^{\mathrm{PV}}_{i,t} - q^{\mathrm{Load}}_{i,t} - q^{\mathrm{l}}_{i,t} = \sum_j \left(q_{ji,t} - x_{ji} \ell_{ji,t}\right) + \sum_k q_{ik,t} \tag{1k}$$

$$q^{\mathrm{l}}_i = p^{\mathrm{l}}_i tan(\theta^{\mathrm{l}}_i), \tag{1l}$$

where $\boldsymbol{c}$ is a vector of grid electricity price and $\boldsymbol{p}^{\mathbf{G}}$ is a vector of electric power purchased by PDS from the grid. As such, the objective function (1a) minimizes the power purchased from the grid. The *DistFlow* model [44,45] is adopted to model balanced power flows as in (1b). Note that the index $ik(ji)$ refers to a distribution line connecting node $i(j)$ and $k(i)$. The voltage drop on a distribution line is represented by constraint (1c). Thermal and power carrying limits of distribution lines are given by constraints (1d) and (1e), respectively. Constraint (1f) denotes the voltage limits. Constraints (1g)–(1i) represent the operating constraints of ESSs. For a detailed description of this high-fidelity ESS model, please refer to [8]. Constraints (1j) and (1k) are nodal active and reactive power balance equations, respectively. The constraint (1l) represents the reactive power demand due to the EdWFs, where θ^{l}_i (is considered fixed) is the power factor angle of the EdWFs. Unless otherwise stated, the **bold** symbols represent matrices/vectors of corresponding variables throughout the paper.

2.2. Original formulation of OPS in WDS

The water distribution network comprises water sources, tanks, pumps, and pipes. The water network is considered a directed graph, $\mathcal{G}_{\mathrm{W}} = \left(\mathcal{N}_{\mathrm{W}}, \mathcal{E}_{\mathrm{W}}\right)$. We assume that a water pump is a type of pipe that imposes a head gain when the pump is on and acts as a closed pipe when the pump is off. Moreover, we assume that the pumps convert electric power into mechanical power at a constant efficiency of η and operate at a constant power factor. The resulting optimal pump scheduling (OPS) model, adopted from [8,46], is given as follows :

$$\text{Min.} \sum_{k \in \mathcal{E}^{\mathrm{P}}_{\mathrm{W}}} \sum_t \lambda_{w k,t} p^{\mathrm{p}}_{k,t} = \boldsymbol{\lambda}^{\top}_{\boldsymbol{w}} \boldsymbol{p}^{\mathbf{p}} \tag{2a}$$

S.t.:

$$f^{\mathrm{R}}_{i,t} + f^{\mathrm{T}}_{i,t} + \sum_j f_{ji,t} = f^{\mathrm{D}}_{i,t} + \sum_k f_{ik,t}, (i \in \mathcal{N}_{\mathrm{W}}) \tag{2b}$$

$$z_{i,t} - z_{j,t} + h_i - h_j = R^{\mathrm{P}}_k \mathrm{sgn}(f_{k,t}) f^2_{k,t}, (k \in \mathcal{E}_{\mathrm{W}} \setminus \mathcal{E}^{\mathrm{P}}_{\mathrm{W}}) \tag{2c}$$

$$\begin{cases} z_{i,t} - z_{j,t} + h_i - h_j + z^{\mathrm{R}}_{k,t} = R^{\mathrm{P}}_k f^2_{k,t}, & \text{if } \alpha_{k,t} = 1 \\ f_{k,t} = 0, & \text{if } \alpha_{k,t} = 0 \end{cases}, (k \in \mathcal{E}^{\mathrm{P}}_{\mathrm{W}}) \tag{2d}$$

$$\underline{S}^{\mathrm{w}}_i \le S^{\mathrm{w}}_{i,0} - \sum_{t=0}^{t} f^{\mathrm{T}}_{i,t} \le \overline{S}^{\mathrm{w}}_i, \quad (i \in \mathcal{N}^{\mathrm{T}}_{\mathrm{w}}) \tag{2e}$$

$$z_{i,t-1} = z_{i,t} - \frac{f^{\mathrm{T}}_{i,t} \Delta t}{A_i} \quad (i \in \mathcal{N}^{\mathrm{T}}_{\mathrm{w}}) \tag{2f}$$

$$\underline{f}_k \le f_{k,t} \le \overline{f}_k \tag{2g}$$

$$\underline{z}_i \le z_{i,t} \le \overline{z}_i \tag{2h}$$

$$\underline{f}^{\mathrm{R}}_i \le f^{\mathrm{R}}_{i,t} \le \overline{f}^{\mathrm{R}}_i \tag{2i}$$

$$\underline{f}^{\mathrm{T}}_i \le f^{\mathrm{T}}_{i,t} \le \overline{f}^{\mathrm{T}}_i \tag{2j}$$

$$z^{\mathrm{R}}_{k,t} = a_{1,k} f_{k,t} + a_{0,k} \tag{2k}$$

$$\eta p^{\mathrm{p}}_{k,t} = f_{k,t} z^{\mathrm{R}}_{k,t} = a_{1,k} f^2_{k,t} + a_{0,k} f_{k,t}, \tag{2l}$$

where, $\lambda_{w k,t}$ is the electricity price rate and $p^{\mathrm{p}}_{k,t}$ is the power consumed by EdWFs. More information on how to calculate $\lambda_{w k,t}$ will be provided when we introduce the MIBC SD-GS-AL algorithm. As such, the objective function (2a) minimizes the cost of power purchased from the PDS. Eq. (2b) represents a nodal water balance in the water network. Constraints (2c) and (2d) describe the hydraulic characteristics of a pipe without a pump and with a pump, respectively. The filling and emptying of a water tank are modeled as in constraint (2e). The constraint (2f) represents the head change of water tanks due to water filling and emptying. Constraints (2g)–(2j) describes the operating constraints of a water network. The constraint (2k) is the head gain characteristics of water pumps. The constraint (2l) describes the power consumed by the water pump.

2.3. Convex hull relaxation

Due to constraints (1b) and (1i) in (1) and (2c), (2d) and (2l) in (2), the OPF model (1) and the OPS model (2) are in non-linear programming (NLP) and MINLP forms, respectively. It has been found in [8] that if one can find the convex hull relaxation for the OPF problem of the MINLP form (1), solving the corresponding MICP model can obtain an exact solution to the original OPF problem with less computational burden. Therefore, in this paper, convex (and quasi-convex) hull relaxation [8,47] is adopted for the convexification of the aforementioned non-convex constraints. For the accuracy of convex (and quasi-convex) hull relaxations, readers are referred to [8,47]. The convex hulls relaxation of (1b) is given as follows:

$$\begin{cases} p^2_{ik,t} + q^2_{ik,t} \le v_{i,t} \ell_{ik,t} \\ \overline{S}^2_{ik} v_i + \underline{v}_i \overline{v}_i \ell_{ik} \le \overline{S}^2_{ik} (\underline{v}_i + \overline{v}_i). \end{cases} \tag{3}$$

The convex hulls relaxation of (1i) is given as follows:

$$\begin{cases} (r^{\mathrm{Batt}}_i + r^{\mathrm{Cvt}}_i)(p^{\mathrm{ES}}_{i,t})^2 + r^{\mathrm{Cvt}}_i (q^{\mathrm{ES}}_{i,t})^2 \le p^{\mathrm{Loss}}_{i,t} v_{i,t} \\ r^{\mathrm{Batt}}_i (q^{\mathrm{ES}}_{i,t})^2 + \underline{v}_i p^{\mathrm{Loss}}_{i,t} \le (\overline{S}^{\mathrm{ES}}_i)^2 (r^{\mathrm{Batt}}_i + r^{\mathrm{Cvt}}_i) \\ (\overline{S}^{\mathrm{ES}}_i)^2 v_i + \underline{v}_i \overline{v}_i p^{\mathrm{Loss}}_{i,t} \le (\overline{S}^{\mathrm{ES}}_i)^2 (\underline{v}_i + \overline{v}_i). \end{cases} \tag{4}$$

Quasi-convex hulls relaxation of (2c) is given as follows:

$$z_{i,t} - z_{j,t} + h_i - h_j \begin{cases} \leqslant (2\sqrt{2} - 2) R^P_k \overline{f}_k f_{k,t} + (3 - 2\sqrt{2}) R^P_k \overline{f}^2_k \\ \geqslant (2\sqrt{2} - 2) R^P_k \underline{f}_k f_{k,t} - (3 - 2\sqrt{2}) R^P_k \underline{f}^2_k \\ \geqslant 2 R^P_k \overline{f}_k f_{k,t} - R^P_k \overline{f}^2_k \\ \leqslant 2 R^P_k \underline{f}_k f_{k,t} + R^P_k \underline{f}^2_k. \end{cases} \tag{5}$$

Similarly, the quasi-convex hulls relaxation of (2d) is given as follows:

$$R^P_k f^2_{k,t} - Z_1 \le 0 \tag{6a}$$

$$Z_2 - R_k^P f_{k,t}^2 \le 0 \tag{6b}$$

$$0 \le f_{k,t} \le M * \alpha, \tag{6c}$$

where, $Z_1 = z_{i,t} - z_{j,t} + h_i - h_j + z_{k,t}^G + M * (1 - \alpha_{k,t})$ and $Z_2 = z_{i,t} - z_{j,t} + h_i - h_j + z_{k,t}^G + M * (\alpha_{k,t} - 1)$. The convex hull relaxation of (6b) is given as follows:

$$Z_2 - R_k^P \overline{f}_{k,t} f_{k,t} \le 0. \tag{7}$$

Finally, the convex hulls relaxation of (2l) is given as follows:

$$\eta p_{k,t}^{\mathrm{p}} \ge a_{1,k} f_{k,t}^2 + a_{0,k} f_{k,t} \tag{8a}$$

$$\eta p_{k,t}^{\mathrm{p}} \le (a_{1,k} \overline{f}_{k,t} + a_{0,k}) f_{k,t} \tag{8b}$$

2.4. Coordinated OPFOPS formulation under the decentralized framework

This subsection presents the formulations of the coordinated OPF-OPS problem under the decentralized framework. Originally, the coordinated OPF-OPS is a single optimization problem, which is given as follows:

$$\text{(CO.)} \ \min \ f := \sum_t c_t p_t^{\mathrm{G}} = \boldsymbol{c}^\top \boldsymbol{p}^{\mathbf{G}} \tag{9a}$$

$$\text{s.t.} \ \ \boldsymbol{p}^{\mathbf{l}} = \boldsymbol{p}^{\mathbf{p}} \tag{9b}$$

$$\boldsymbol{x}_{\mathbf{p}} \in \mathcal{X}_{\mathrm{p}}, \boldsymbol{x}_{\mathbf{w}} \in \mathcal{X}_{\mathrm{w}}, \tag{9c}$$

where,

$$\mathcal{X}_{\mathrm{p}} := \{\text{(1c)–(1h), (1j)–(1l), (3), (4)}\} \tag{10}$$

is a convex constraint set of the OPF model (1) while the mixed-integer convex constraint set ($\mathcal{X}_{\mathrm{w}}$) of the OPS model (2) is defined as follows:

$$\mathcal{X}_{\mathrm{w}} := \{\text{(2b), (2e)–(2k), (5), (6a), (6c), (7), (8)}\}. \tag{11}$$

Moreover, the decision variables of the OPF model (1) and the OPS model (2) are collectively referred to as $\boldsymbol{x}_{\mathbf{p}}$ and $\boldsymbol{x}_{\mathbf{w}}$, respectively. The expression (9a) represents the objective function of the coordinated OPF-OPS, e.g., minimization of power purchased from the grid. Constraint (9b) links the OPF and OPS models, where $\boldsymbol{p}^{\mathbf{l}}$ represents the power demand of EdWFs (e.g., pumps) from WDS in the OPF model, while $\boldsymbol{p}^{\mathbf{p}}$ represents EdWF power from PDS in the OPS model.

To solve the problem (9) in a decentralized manner, it is decomposed into two subproblems. In our proposed framework, the power operator solves the following subproblem:

$$\text{(PO.)} \ \min_{\boldsymbol{x}_{\mathbf{p}}} \ f_p(\boldsymbol{x}_{\mathbf{p}}) := \sum_t c_t p_t^{\mathrm{G}} = \boldsymbol{c}^\top \boldsymbol{p}^{\mathbf{G}} \tag{12a}$$

$$\text{s.t.} \ \ \boldsymbol{p}^{\mathbf{l}} = \boldsymbol{y} \tag{12b}$$

$$\boldsymbol{x}_{\mathbf{p}} \in \mathcal{X}_{\mathrm{p}}, \tag{12c}$$

while the water operator solves the following subproblem:

$$\text{(WO.)} \ \min_{\boldsymbol{x}_{\mathbf{w}}} \ w \ \text{(i.e., } (\lambda_{\mathbf{w}}^k)^\top \boldsymbol{p}^{\mathbf{p}}) \tag{13a}$$

$$\text{s.t.} \ \ \boldsymbol{p}^{\mathbf{p}} = \boldsymbol{y} \tag{13b}$$

$$\boldsymbol{x}_{\mathbf{w}} \in \mathcal{X}_{\mathrm{w}}, \tag{13c}$$

where $\boldsymbol{y}$ is a vector of auxiliary variables introduced to render a decomposable structure, the expression (12a) represents the objective function of the power subproblem, e.g., minimization of power purchased from the grid, and w is the virtual objective function of the OPS, introduced to facilitate the introduction of the algorithm. The reason why we use a virtual objective function for the OPS subproblem is detailed in Section 3. The proposed decentralized formulation of (9) (i.e., (12) and (13)) has one significant advantage: it does not require any entity with access to both $\mathcal{X}_{\mathrm{p}}$ and $\mathcal{X}_{\mathrm{w}}$. It is important to note that there does not exist any entity that has access to both PDS and WDS information. Therefore, the proposed decentralized formulation of coordinated OPF-OPS provides a real-world-compatible framework for the coordination of PDS and WDS.

It can be observed from (12b) and (13b) that the two subproblems are still coupled through $\boldsymbol{y}$ as EdWFs powers $\boldsymbol{p}^{\mathbf{p}}$ in a WDS act as a load $\boldsymbol{p}^{\mathbf{l}}$ in a PDS. If two models are solved independently without being coordinated by a proper decentralized algorithm, the boundary variables, i.e., $\boldsymbol{p}^{\mathbf{l}}$ and $\boldsymbol{p}^{\mathbf{p}}$ may not match with each other, which will result in increased cost and/or insecure operation of both systems. Note that the EDWFs powers $\boldsymbol{p}^{\mathbf{l}}$ and $\boldsymbol{p}^{\mathbf{p}}$ are boundary variables which are discontinuous due to their dependence on binary variables that represent the on/off nature of EdWFs. To be specific, while Eq. (2l) means that pump power ($p_{k,t}^{\mathrm{p}}$) is dependent on the water flow ($f_{k,t}$) on a pipe where a pump is installed, (2d) indicates that a water flow on a pipe where a pump is installed is dependent on binary variable ($\alpha_{k,t}$). As a result, pump power ($p_{k,t}^{\mathrm{p}}$) is a binary-dependent and discontinuous boundary variable.

3. MIBC SD-GS-AL decentralized algorithm

As mentioned in the introduction section, existing decentralized or distributed optimization algorithms are not guaranteed to converge and be optimal when they are used to coordinate subproblems (12) and (13) since the boundary variables of these two problems are discontinuous (as explained in Section 2.4). Therefore, this paper proposes a MIBC SD-GS-AL decentralized optimization algorithm. We first provide an overview of the proposed MIBC SD-GS-AL algorithm in Section 3.1 and then discuss the privacy-preserving feature of the MIBC SD-GS-AL algorithm in Section 3.2. Last but not least, the comparison of ADMM, original SD-GS-AL, and MIBC SD-GS-AL algorithms is provided in the last subsection.

3.1. Overview of the proposed algorithm

The key steps of the proposed MIBC SD-GS-AL algorithm can be found in Algorithm 1. Note the boundary variables $\boldsymbol{p}^{\mathbf{l}}$ and $\boldsymbol{p}^{\mathbf{p}}$ need to match at each time instant as the optimization models (12) and (13) are multi-period optimization models with a total of T periods. There is a $N \times T$ number of y, and they are collectively referred to as $\boldsymbol{y}$ here, i.e., $\boldsymbol{y} = (y_{n,t}) \in \mathbb{R}^{N \times T}$, where N is the number of coupling points (i.e., EdWFs) between PDS and WDS and T is the number of time periods. To make the formulations and algorithm brief, the subscripts are eliminated in our formulations. Before we formally introduce the algorithm, we would like to introduce some new variables. We use $\boldsymbol{\alpha}_{\mathbf{p}}$ and $\boldsymbol{\alpha}_{\mathbf{w}}$ to collectively represent all the binary/integer decision variables (including those that are located at the boundary) of the OPF and the OPS models, respectively. Even though there are no integer (including binary) variables present in the PDS constraint set (10), we have introduced binary variables $\boldsymbol{\alpha}_{\mathbf{p}}$ here for the generalization (for the potential future adoption) of the proposed algorithm. Moreover, we use $\boldsymbol{\alpha}$ to represent integer/binary variables that are located at the boundary of the subproblem, i.e., $\boldsymbol{\alpha}$ is a subset of $\boldsymbol{\alpha}_{\mathbf{w}}$ in this case. And, $\boldsymbol{\alpha}_{\mathbf{p}}$ and $\boldsymbol{\alpha}_{\mathbf{w}}$ are subsets of $\boldsymbol{x}_{\mathbf{p}}$ and $\boldsymbol{x}_{\mathbf{w}}$, respectively.

The algorithm is initialized by assigning convergence tolerance ϵ and penalty parameter ρ in Step 1, along with maximum inner and outer loop iterations, J and K, respectively. Note that Step 2 to Step 8 constitutes the outer loop while Step 3 is the inner loop. Moreover, the starting points for auxiliary variable $\boldsymbol{y}$, binary variables $\boldsymbol{\alpha}_{\mathbf{p}}$ and $\boldsymbol{\alpha}_{\mathbf{w}}$, Lagrangian multipliers $\lambda_{\mathbf{p}}^k$ and $\lambda_{\mathbf{w}}^k$, and Lagrangian lower bounds $\breve{\varphi}_{\mathrm{p}}$ and $\breve{\varphi}_{\mathrm{w}}$ are assigned. For the initial values of auxiliary variable $\boldsymbol{y}$ and Lagrangian multipliers $\lambda_{\mathbf{p}}^k$ and $\lambda_{\mathbf{w}}^k$, we can use zero. For the initial values of binary variables $\boldsymbol{\alpha}_{\mathbf{p}}$ and $\boldsymbol{\alpha}_{\mathbf{w}}$, we can use any feasible solution. For the Lagrangian lower bounds $\breve{\varphi}_{\mathrm{p}}$ and $\breve{\varphi}_{\mathrm{w}}$, we can use any small negative number.

For the current iteration k, initial values of auxiliary variable $\boldsymbol{y}$, binary variables $\boldsymbol{\alpha}_{\mathbf{p}}$ and $\boldsymbol{\alpha}_{\mathbf{w}}$, Lagrangian multipliers $\lambda_{\mathbf{p}}^k$ and $\lambda_{\mathbf{w}}^k$, and Lagrangian lower bounds $\hat{\varphi}_{\mathrm{p}}$ and $\hat{\varphi}_{\mathrm{w}}$ are set to that of the previous

Algorithm 1 MIBC SD-GS-AL decentralized Algorithm

1: **Parameters initialization:**

1. **Initial parameters:** Choose the initial parameters ϵ, ρ, J, K.
2. **Starting point:** Choose starting points for $\boldsymbol{y}^0, \boldsymbol{\lambda_p}^0, \boldsymbol{\lambda_w}^0, \boldsymbol{\alpha_p}^0, \boldsymbol{\alpha_w}^0, \breve{\varphi}_{\mathrm{p}}^0, \breve{\varphi}_{\mathrm{w}}^0$. Set $k = 1$.

2: **Iteration initialization:** Set $\{\boldsymbol{y}, \boldsymbol{\lambda_p}, \boldsymbol{\lambda_w}, \boldsymbol{\alpha_p}, \boldsymbol{\alpha_w}, \breve{\varphi}_{\mathrm{p}}, \breve{\varphi}_{\mathrm{w}}\}^k := \{\boldsymbol{y}, \boldsymbol{\lambda_p}, \boldsymbol{\lambda_w}, \boldsymbol{\alpha_p}, \boldsymbol{\alpha_w}, \breve{\varphi}_{\mathrm{p}}, \breve{\varphi}_{\mathrm{w}}\}^{k-1}$.

3: **Continuous primal iteration:** Solve the following optimization problems (14a) and (14b) in parallel and then (14c) with the latest available updates of $\boldsymbol{p^{\mathrm{I}}}, \boldsymbol{p^{\mathrm{P}}}, \boldsymbol{y}, \boldsymbol{\lambda_p}, \boldsymbol{\lambda_w}, \boldsymbol{\alpha_p}, \boldsymbol{\alpha_w}$. Repeat (14) J times.

$$LR_{\mathrm{p}}^k, \boldsymbol{x_p}^k, \boldsymbol{p^{\mathrm{I}}}^{(k)} \leftarrow \min_{\boldsymbol{x_p}, \boldsymbol{p^{\mathrm{I}}}} \{L_{\rho,\mathrm{p}}\left(\boldsymbol{x_p}, \boldsymbol{y}^k, \boldsymbol{\lambda_p}^k\right) : \boldsymbol{\alpha_p} \in \boldsymbol{\alpha_p}^k\} \tag{14a}$$

$$LR_{\mathrm{w}}^k, \boldsymbol{x_w}^k, \boldsymbol{p^{\mathrm{P}}}^{(k)} \leftarrow \min_{\boldsymbol{x_w}, \boldsymbol{p^{\mathrm{P}}}} \{L_{\rho,\mathrm{w}}\left(\boldsymbol{x_w}, \boldsymbol{y}^k, \boldsymbol{\lambda_w}^k\right) : \boldsymbol{\alpha_w} \in \boldsymbol{\alpha_w}^k\} \tag{14b}$$

$$\boldsymbol{y}^k \leftarrow \min_{\boldsymbol{y}} \left\{ \left\| \boldsymbol{y} - \boldsymbol{p^{\mathrm{I}}}^{(k)} \right\|_2^2 + \left\| \boldsymbol{p^{\mathrm{P}}}^{(k)} - \boldsymbol{y} \right\|_2^2 \right\} \tag{14c}$$

Finally, the Lagrangian upper bounds are computed as follows:

$$\hat{\varphi}_{\mathrm{p}}^k \leftarrow LR_{\mathrm{p}}^k + \frac{\rho}{2} \left\| \boldsymbol{y}^k - \boldsymbol{p^{\mathrm{I}}}^{(k)} \right\|_2^2 \tag{15a}$$

$$\hat{\varphi}_{\mathrm{w}}^k \leftarrow LR_{\mathrm{w}}^k + \frac{\rho}{2} \left\| \boldsymbol{p^{\mathrm{P}}}^{(k)} - \boldsymbol{y}^k \right\|_2^2 \tag{15b}$$

4: **Stopping test:** If $(\hat{\varphi}_{\mathrm{p}}^k + \hat{\varphi}_{\mathrm{w}}^k) - (\breve{\varphi}_{\mathrm{p}}^k + \breve{\varphi}_{\mathrm{w}}^k) \leq \epsilon$, Stop: $\left(\boldsymbol{x_p}^k, \boldsymbol{x_w}^k, \boldsymbol{y}^k, \boldsymbol{\lambda_p}^k, \boldsymbol{\lambda_w}^k, \boldsymbol{\alpha_p}^k, \boldsymbol{\alpha_w}^k, \breve{\varphi}_{\mathrm{p}}^k, \breve{\varphi}_{\mathrm{w}}^k\right)$ is the solution. Otherwise, continue.

5: **MIP primal iteration:** Solve the following MIP subproblems (16a) and (16b) in parallel to obtain intermediate Lagrangian lower bounds and to update binary variables

$$\tilde{\varphi}_{\mathrm{p}}, \boldsymbol{\alpha_p}^k \leftarrow \varphi_{\mathrm{p}}\left(\boldsymbol{x_p}, \boldsymbol{\lambda_p}^k + \rho\left(\boldsymbol{y}^k - \boldsymbol{p^{\mathrm{I}}}^{(k)}\right)\right), \tag{16a}$$

$$\tilde{\varphi}_{\mathrm{w}}, \boldsymbol{\alpha_w}^k \leftarrow \varphi_{\mathrm{w}}\left(\boldsymbol{x_w}, \boldsymbol{\lambda_w}^k + \rho\left(\boldsymbol{p^{\mathrm{P}}}^{(k)} - \boldsymbol{y}^k\right)\right). \tag{16b}$$

6: **Iteration declaration and dual updates:** The iteration is declared *forward* iteration if the following inequality holds:

$$(\hat{\varphi}_{\mathrm{p}} + \hat{\varphi}_{\mathrm{w}}) \geq (\tilde{\varphi}_{\mathrm{p}} + \tilde{\varphi}_{\mathrm{w}}) \geq (\breve{\varphi}_{\mathrm{p}} + \breve{\varphi}_{\mathrm{w}}). \tag{17}$$

Perform the following updates if the iteration is declared *forward*:

$$\boldsymbol{\lambda_p}^k \leftarrow \boldsymbol{\lambda_p}^k + \rho\left(\boldsymbol{y}^k - \boldsymbol{p}^{I(k)}\right)$$
$$\boldsymbol{\lambda_w}^k \leftarrow \boldsymbol{\lambda_w}^k + \rho\left(\boldsymbol{p}^{p(k)} - \boldsymbol{y}^k\right)$$
$$\breve{\varphi}_{\mathrm{p}}^k \leftarrow \tilde{\varphi}_{\mathrm{p}}, \ \breve{\varphi}_{\mathrm{w}}^k \leftarrow \tilde{\varphi}_{\mathrm{w}}$$

Otherwise, the iteration is declared *neutral*: Algorithm continues without updates.

7: **Auxiliary variables update:** If $\alpha_{n,t}^k - \alpha_{n,t}^{k-1} = 1$, perform following update for all n and t:

$$y_{n,t}^k \leftarrow y_{n,t}^a : a = \max\{\mathcal{M}\}, \mathcal{M} \subset \mathcal{K}_k.$$

If $\alpha_{n,t}^k - \alpha_{n,t}^{k-1} = 0$, perform: $y_{n,t}^k \leftarrow y_{n,t}^k$. If $\alpha_{n,t}^k - \alpha_{n,t}^{k-1} = -1$, do: $y_{n,t}^k \leftarrow 0$.

8: **Loop:** Set $k := k + 1$ and go back to Step 2.

iteration $k-1$ in Step 2. The $L_{\rho,\mathrm{p}}$ in (14a) and $L_{\rho,w}$ in (14b) have the following detailed expressions in Step 3:

$$L_{\rho,\mathrm{p}} = \boldsymbol{c}^\top \boldsymbol{p^{\mathrm{G}}} - (\boldsymbol{\lambda_p}^k)^\top \boldsymbol{p^{\mathrm{I}}} + \frac{\rho}{2} \left\| \boldsymbol{y}^k - \boldsymbol{p^{\mathrm{I}}} \right\|_2^2, \tag{18a}$$

$$L_{\rho,\mathrm{w}} = (\boldsymbol{\lambda_w}^k)^\top \boldsymbol{p^{\mathrm{P}}} + \frac{\rho}{2} \left\| \boldsymbol{p^{\mathrm{P}}} - \boldsymbol{y}^k \right\|_2^2. \tag{18b}$$

Note that (18a) and (18b) are the augmented Lagrangian relaxations of (12) and (13) and are computed in parallel by P-DSO and W-DSO, respectively. In addition, the auxiliary variable $\boldsymbol{y}$ is computed as in (14c). The auxiliary variable computation (14c) can be assigned to either of the operators (in our study, we assign it to the P-DSO) as the only information shared is the boundary variables from both networks. Finally, the Lagrangian upper bounds $\hat{\varphi}_{\mathrm{p}}$ and $\hat{\varphi}_{\mathrm{w}}$ are computed as in (15). It is worth noting that, in Step 3, binary variables are fixed so that PDS and WDS sub-problems are continuous. Note that binary variables are fixed from the solutions of the previous iteration of MIP subproblems in Step 5.

Note that Algorithm 1 is said to converge if the difference of Lagrangian bounds $((\hat{\varphi}_{\mathrm{p}}+\hat{\varphi}_{\mathrm{w}})-(\breve{\varphi}_{\mathrm{p}}+\breve{\varphi}_{\mathrm{w}}))$ is within the limit of tolerance, as stated in Step 4. When Algorithm 1 converges, $\boldsymbol{p^{\mathrm{P}}} = \boldsymbol{p^{\mathrm{I}}} = \boldsymbol{y}$, $\boldsymbol{\lambda_w} = \boldsymbol{\lambda_p}$ and the second term of (18a) and (18b) becomes zero. Therefore, the WDS essentially minimizes the cost of power purchased $(\boldsymbol{\lambda_w}^\top \boldsymbol{p^{\mathrm{P}}})$, where $\boldsymbol{\lambda_w}$ can be interpreted as the rate of electricity paid by WDS to PDS. Note that the virtual objective function w of the OPS model will be dropped onwards. In this paper, the proposed MIBC SD-GS-AL algorithm is used to coordinate the MICP subproblems. Therefore, it converges to the global optimal solution of the centralized implementation of MICP subproblems (9).

In Step 5, the intermediate Lagrangian lower bounds and the binary variables are obtained. The φ_{p} and φ_{w} ((16a) and (16b), respectively) used to obtain the intermediate Lagrangian lower bounds $\tilde{\varphi}_{\mathrm{p}}$ and $\tilde{\varphi}_{\mathrm{w}}$ in Step 5 are also computed in parallel by P-DSO and W-DSO, respectively, and are given as follows:

$$\varphi_{\mathrm{p}}\left(\boldsymbol{x_p}, \boldsymbol{\lambda_p}^k\right) = \min_{\boldsymbol{x_p}} \left\{ \boldsymbol{c}^\top \boldsymbol{p^{\mathrm{G}}} - (\boldsymbol{\lambda_p}^k)^\top \boldsymbol{p^{\mathrm{I}}} : \boldsymbol{x_p} \in \mathcal{X}_{\mathrm{p}} \right\}, \tag{19a}$$

$$\varphi_{\mathrm{w}}\left(\boldsymbol{x_w}, \boldsymbol{\lambda_w}^k\right) = \min_{\boldsymbol{x_w}} \left\{ (\boldsymbol{\lambda_w}^k)^\top \boldsymbol{p^{\mathrm{P}}} : \boldsymbol{x_w} \in \mathcal{X}_{\mathrm{w}} \right\}. \tag{19b}$$

Note that the binary variables are not fixed in Step 5, although they are fixed in Step 3. The intermediate Lagrangian lower bounds obtained in Step 5 go through a quality check (17) in Step 6. The current iteration k is declared either *forward* or *neutral* step iteration based on the inequality (17). If the inequality is satisfied, the current iteration is declared *forward* step iteration. Otherwise, the current iteration is declared *neutral* step iteration. If the iteration is declared *forward*, Lagrangian lower bounds and Lagrangian multipliers are updated in a decentralized manner. Otherwise, the algorithm continues without updates.

Step 7 is proposed to handle the boundary variables $\boldsymbol{p^{\mathrm{P}}}/\boldsymbol{p^{\mathrm{I}}}$ which are binary-dependent (i.e., discontinuous). To solve continuous subproblems in Step 3 in kth iteration, auxiliary variables $\boldsymbol{y}$ from the previous iteration $(k-1)$th are used. However, if the binary variables, i.e., $\boldsymbol{\alpha}$ (note that this refers to those binary variables that are located at the boundary of the subproblem) change in successive iterations in Step 5, auxiliary variables need to be updated accordingly. In the original SD-GS-AL algorithm [42], this step does not exist, which leaves values of auxiliary variables (that are binary dependent) unchanged. Consequently, the original SD-GS-AL algorithm uses unchanged auxiliary variables and fails to converge. For the update of auxiliary variables, the following rules are introduced in Step 7:

1. If the difference of binary variables in successive iterations is 1, the auxiliary variable y for the next iteration is assigned to the most recent non-zero y value.
2. If binary variables do not change in successive iterations, auxiliary variables are left unchanged.
3. If the difference of binary variables in successive iterations is -1, auxiliary variables are assigned to 0.

Table 1
Comparison of ADMM, Original SD-GS-AL, and MIBC SD-GS-AL algorithms based on mathematical operations.

ADMM	Original SD-GS-AL	MIBC SD-GS-AL
(1) P-DSO and W-DSO solve the following two models in parallel: $\boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}(k)} \leftarrow \min_{\boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}}} \{\boldsymbol{c}^{\top}\boldsymbol{p}^{\mathbf{G}} - (\lambda^{k})^{\top}\boldsymbol{p}^{\mathbf{I}} + \frac{\rho}{2}\left\|\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{p}^{\mathbf{I}}\right\|_2^2\}$ $\boldsymbol{x}_{\mathrm{w}}{}^{k}, \boldsymbol{p}^{\mathbf{p}(k)} \leftarrow \min_{\boldsymbol{x}_{\mathrm{w}}, \boldsymbol{p}^{\mathbf{p}}} \{(\lambda^{k})^{\top}\boldsymbol{p}^{\mathbf{p}} + \frac{\rho}{2}\left\|\boldsymbol{p}^{\mathbf{p}} - \boldsymbol{p}^{\mathbf{I}(k)}\right\|_2^2\}$ (2) The central coordinator updates Lagrangian multipliers as follows: $\lambda^{k} \leftarrow \lambda^{k} + \rho\left(\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{p}^{\mathbf{I}(k)}\right)$	(1) P-DSO and W-DSO solve the following two models in parallel with integer variables fixed: $LR_{\mathrm{p}}^{k}, \boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}(k)} \leftarrow \min_{\boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}}} \{\boldsymbol{c}^{\top}\boldsymbol{p}^{\mathbf{G}} - (\lambda_{\mathbf{p}}^{k})^{\top}\boldsymbol{p}^{\mathbf{I}} + \frac{\rho}{2}\left\|\boldsymbol{y}^{(k)} - \boldsymbol{p}^{\mathbf{I}}\right\|_2^2\}$ $LR^{k_{\mathrm{w}}}, \boldsymbol{x}_{\mathrm{w}}{}^{k}, \boldsymbol{p}^{\mathbf{p}(k)} \leftarrow \min_{\boldsymbol{x}_{\mathrm{w}}, \boldsymbol{p}^{\mathbf{p}}} \{(\lambda_{\mathrm{w}^{k}})^{\top}\boldsymbol{p}^{\mathbf{p}} + \frac{\rho}{2}\left\|\boldsymbol{p}^{\mathbf{p}} - \boldsymbol{y}^{(k)}\right\|_2^2\}$ P-DSO also computes the auxiliary variables as follows: $\boldsymbol{y}^{k} \leftarrow \min_{\boldsymbol{y}} \left\{\left\|\boldsymbol{y} - \boldsymbol{p}^{\mathbf{I}(k)}\right\|_2^2 + \left\|\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{y}\right\|_2^2\right\}$ Finally, P-DSO and W-DSO compute the Lagrangian upper bounds as follows: $\hat{\varphi}_{\mathrm{p}}^{k} \leftarrow LR_{\mathrm{p}}^{k} + \frac{\rho}{2}\left\|\boldsymbol{y}^{k} - \boldsymbol{p}^{\mathbf{I}(k)}\right\|_2^2$ $\hat{\varphi}_{\mathrm{w}}^{k} \leftarrow LR_{\mathrm{w}}^{k} + \frac{\rho}{2}\left\|\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{y}^{k}\right\|_2^2$ (2) P-DSO and W-DSO solve the following two models in parallel to obtain intermediate Lagrangian lower bounds without fixing integer variables $\tilde{\varphi}_{\mathrm{p}}, \boldsymbol{\alpha}_{\mathbf{p}}{}^{k} \leftarrow \min_{\boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}}} \{\boldsymbol{c}^{\top}\boldsymbol{p}^{\mathbf{G}} - (\lambda_{\mathbf{p}}^{k})^{\top}\boldsymbol{p}^{\mathbf{I}}\}$ $\tilde{\varphi}_{\mathrm{p}}, \boldsymbol{\alpha}_{\mathbf{w}^{k}} \leftarrow \min_{\boldsymbol{x}_{\mathrm{w}}, \boldsymbol{p}^{\mathbf{p}}} \{(\lambda_{\mathbf{w}^{k}})^{\top}\boldsymbol{p}^{\mathbf{p}}\}$ (3) P-DSO and W-DSO update Lagrangian multipliers and Lagrangian lower bounds as follows if the current iteration is evaluated forward step iteration using (17): $\boldsymbol{\lambda}_{\mathbf{p}}{}^{k} \leftarrow \boldsymbol{\lambda}_{\mathbf{p}}{}^{k} + \rho\left(\boldsymbol{y}^{k} - \boldsymbol{p}^{\mathbf{I}(k)}\right),\ \hat{\varphi}_{\mathrm{p}}^{k} \leftarrow \tilde{\varphi}_{\mathrm{p}}$ $\boldsymbol{\lambda}_{\mathbf{w}}{}^{k} \leftarrow \boldsymbol{\lambda}_{\mathbf{w}}{}^{k} + \rho\left(\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{y}^{k}\right),\ \hat{\varphi}_{\mathrm{w}}^{k} \leftarrow \tilde{\varphi}_{\mathrm{w}}$	(1) P-DSO and W-DSO solve the following two models in parallel with integer variables fixed: $LR_{\mathrm{p}}^{k}, \boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}(k)} \leftarrow \min_{\boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}}} \{\boldsymbol{c}^{\top}\boldsymbol{p}^{\mathbf{G}} - (\lambda_{\mathbf{p}}^{k})^{\top}\boldsymbol{p}^{\mathbf{I}} + \frac{\rho}{2}\left\|\boldsymbol{y}^{(k)} - \boldsymbol{p}^{\mathbf{I}}\right\|_2^2\}$ $LR^{k_{\mathrm{w}}}, \boldsymbol{x}_{\mathrm{w}}{}^{k}, \boldsymbol{p}^{\mathbf{p}(k)} \leftarrow \min_{\boldsymbol{x}_{\mathrm{w}}, \boldsymbol{p}^{\mathbf{p}}} \{(\lambda_{\mathrm{w}^{k}})^{\top}\boldsymbol{p}^{\mathbf{p}} + \frac{\rho}{2}\left\|\boldsymbol{p}^{\mathbf{p}} - \boldsymbol{y}^{(k)}\right\|_2^2\}$ P-DSO also computes the auxiliary variables as follows: $\boldsymbol{y}^{k} \leftarrow \min_{\boldsymbol{y}} \left\{\left\|\boldsymbol{y} - \boldsymbol{p}^{\mathbf{I}(k)}\right\|_2^2 + \left\|\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{y}\right\|_2^2\right\}$ Finally, P-DSO and W-DSO compute the Lagrangian upper bounds as follows: $\hat{\varphi}_{\mathrm{p}}^{k} \leftarrow LR_{\mathrm{p}}^{k} + \frac{\rho}{2}\left\|\boldsymbol{y}^{k} - \boldsymbol{p}^{\mathbf{I}(k)}\right\|_2^2$ $\hat{\varphi}_{\mathrm{w}}^{k} \leftarrow LR_{\mathrm{w}}^{k} + \frac{\rho}{2}\left\|\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{y}^{k}\right\|_2^2$ (2) P-DSO and W-DSO solve the following two models in parallel to obtain intermediate Lagrangian lower bounds without fixing integer variables $\tilde{\varphi}_{\mathrm{p}}, \boldsymbol{\alpha}_{\mathbf{p}}{}^{k} \leftarrow \min_{\boldsymbol{x}_{\mathbf{p}}, \boldsymbol{p}^{\mathbf{I}}} \{\boldsymbol{c}^{\top}\boldsymbol{p}^{\mathbf{G}} - (\lambda_{\mathbf{p}}^{k})^{\top}\boldsymbol{p}^{\mathbf{I}}\}$ $\tilde{\varphi}_{\mathrm{p}}, \boldsymbol{\alpha}_{\mathbf{w}^{k}} \leftarrow \min_{\boldsymbol{x}_{\mathrm{w}}, \boldsymbol{p}^{\mathbf{p}}} \{(\lambda_{\mathbf{w}^{k}})^{\top}\boldsymbol{p}^{\mathbf{p}}\}$ (3) P-DSO and W-DSO update Lagrangian multipliers and Lagrangian lower bounds as follows if the current iteration is evaluated forward step iteration using (17): $\boldsymbol{\lambda}_{\mathbf{p}}{}^{k} \leftarrow \boldsymbol{\lambda}_{\mathbf{p}}{}^{k} + \rho\left(\boldsymbol{y}^{k} - \boldsymbol{p}^{\mathbf{I}(k)}\right),\ \hat{\varphi}_{\mathrm{p}}^{k} \leftarrow \tilde{\varphi}_{\mathrm{p}}$ $\boldsymbol{\lambda}_{\mathbf{w}}{}^{k} \leftarrow \boldsymbol{\lambda}_{\mathbf{w}}{}^{k} + \rho\left(\boldsymbol{p}^{\mathbf{p}(k)} - \boldsymbol{y}^{k}\right),\ \hat{\varphi}_{\mathrm{w}}^{k} \leftarrow \tilde{\varphi}_{\mathrm{w}}$ (4) W-DSO performs the following update (a) If $\alpha_{n,t}^{k} - \alpha_{n,t}^{k-1} = 1$, do following for all n and t: $y_{n,t}^{k} \leftarrow y_{n,t}^{a} : a = \max\{\mathcal{M}\}, \mathcal{M} \subset \mathcal{K}_{k}$. (b) If $\alpha_{n,t}^{k} - \alpha_{n,t}^{k-1} = 0$, do: $y_{n,t}^{k} \leftarrow y_{n,t}^{k}$. (c) If $\alpha_{n,t}^{k} - \alpha_{n,t}^{k-1} = -1$, do: $y_{n,t}^{k} \leftarrow 0$.

3.2. Privacy-preserving and cost-saving on communication

To the best of our knowledge, the proposed MIBC SD-GS-AL algorithm is the only applicable algorithm when boundary variables are discontinuous. Additionally, the MIBC SD-GS-AL algorithm requires the exchange of less information than other mature distributed/decentralized algorithms do, which will bring at least two benefits: (1) preserves the privacies of PDS and WDS, and (2) reduces the cost of building communication channels between the two operators. For example, in ADMM [32], the dual parameters, i.e., the Lagrangian multipliers, need to be updated and communicated centrally, requiring more information exchange (i.e., less privacy preservation) and more communication channels. A similar issue applies to ALADIN [41], as the Hessian matrix needs to be centrally updated and communicated. However, in the proposed MIBC SD-GS-AL algorithm, WDS and PDS update Lagrangian multipliers in a decentralized fashion (see Step 6), and the hessian does not need to be computed.

3.3. Comparison of ADMM, original SD-GS-AL, and MIBC SD-GS-AL algorithms: A focus on mathematical formulas

In this subsection, we provide a comparison of ADMM, original SD-GS-AL, and MIBC SD-GS-AL algorithms from the perspective of mathematical operations and formulas utilized in these algorithms. The Table 1 succinctly provides what each operator has to do for all three algorithms, highlighting the difference of these algorithms in terms of mathematical operations and formulas.

4. Case study

This section first describes the simulation setup. Second, the advantages of the proposed framework and the algorithm are illustrated via simulation results. The proposed approach is tested for both weak coupling (single EdWF) and strong coupling (multiple EdWFs) of PDS and WDS. For the sake of convenience, the electric pump is used as an example of EdWFs in this paper.

Table 2
Pump characteristics parameters (Case 1 and 2).

	Pump #	Pipe	a_0	a_1
Case 1	Pump 1	9–10	94.46	0.043
Case 2	Pump 1	1–2	204.46	0.043
	Pump 2	26–25	65.23	0.023

4.1. Simulation setup

For Case 1, the modified IEEE 13-node test feeder [48] is adopted to represent the PDS, while the 11-node water network [49] is used to represent the WDS, as shown in Fig. 2. For Case 2, the modified IEEE 37-node test feeder [48] is used to represent the PDS, while the 36-node water network from Cherry Hills/Brushy Plains, New Haven, CT [49] is adopted to represent the WDS, as shown in Fig. 3. In Figs. 2 and 3, the network drawn with green color represents PDS while the network drawn with blue color represents WDS. Power and water distribution systems are chosen such that their area of coverage is similar.

As shown in Fig. 2, a weak coupling exists between PDS and WDS in Case 1, i.e., an electric pump (between node 9 (i.e., reservoir) and node 10 in WDS) is supplied by node 680 of the PDS. In contrast, as shown in Fig. 3, a stronger coupling exists in Case 2, i.e., the PDS and WDS are coupled through two pumps. An electric pump, i.e., Pump 1 (between node 1 (i.e., reservoir) and node 2 in WDS), is supplied by node 731 of the PDS, while another pump (Pump 2) is supplied by node 724 of the PDS. In Case 2 (Fig. 3), Pump 2 is equipped with a bypass pipe. Pump 2 is utilized when the tank is supplying the WDS, while the bypass pipe is utilized when the tank is being filled. The pump characteristics parameters used for both cases are provided in Table 2. Moreover, the capacity of energy storage systems and solar photovoltaics used in PDS of both cases is 700 KWh/150 KVA and 200 kW, respectively.

The power and water subproblems are multi-period optimization models. To realize the multi-period operation, load and PV profiles were used to modify the load and PV output, which are given in Fig. 4. Moreover, algorithm parameters used for both Case 1 and Case 2 are provided in Table 3. For the initialization of $\boldsymbol{\alpha}^{0}$, a feasible

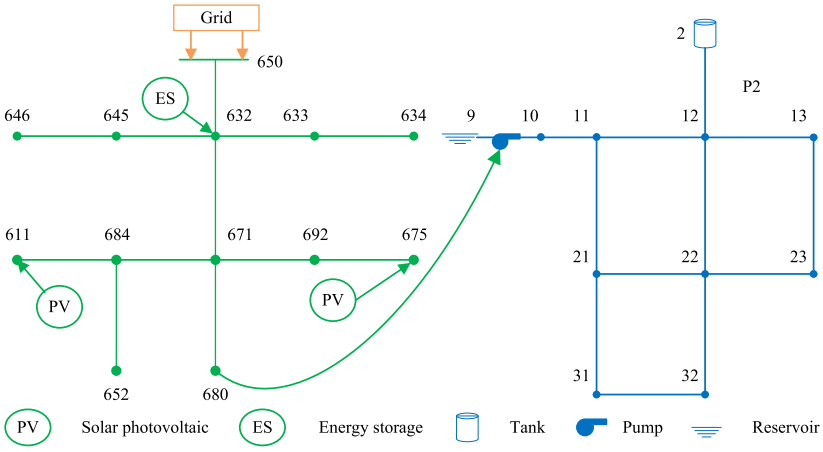


Fig. 2. PDS and WDS topology (Case 1).

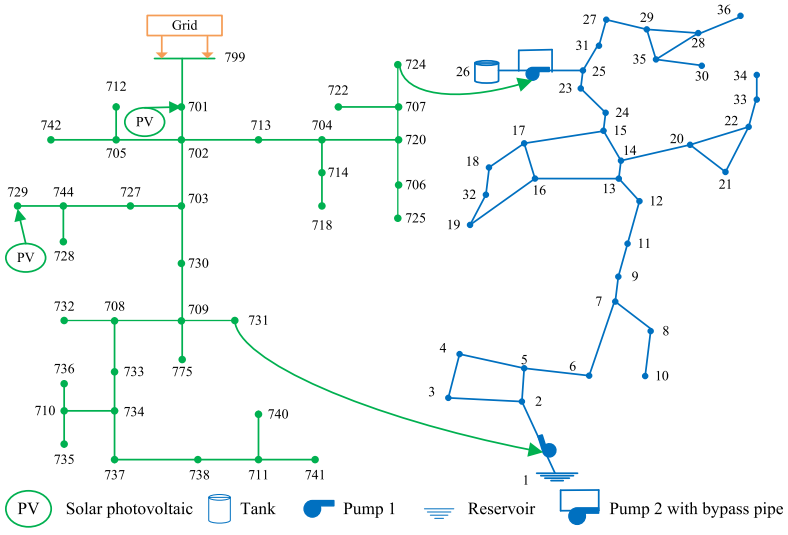


Fig. 3. PDS and WDS topology (Case 2).

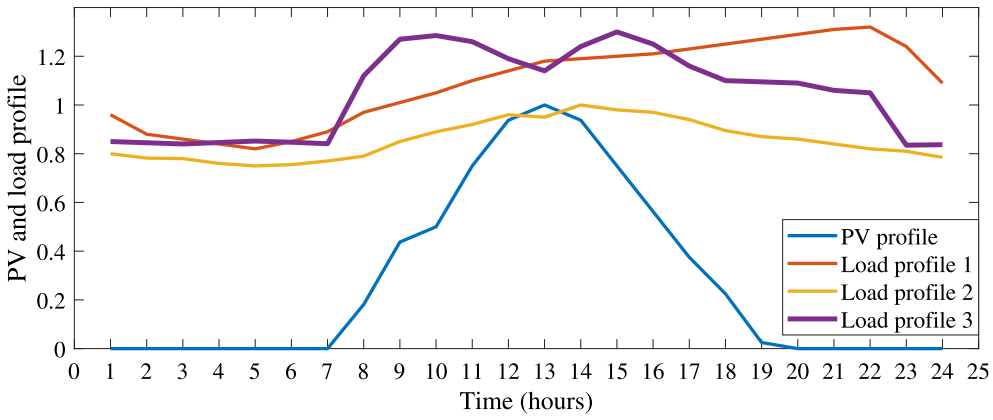


Fig. 4. Load and PV profiles (Case 1 and 2) [8].

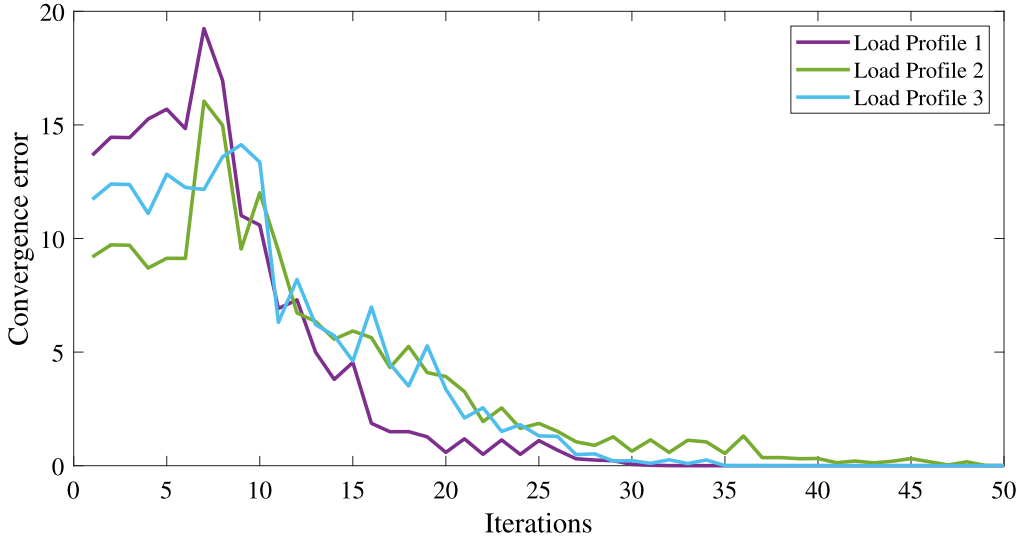


Fig. 5. Convergence error of MIBC SD-GS-AL for different load profiles (Case 1).

Table 3
Algorithm parameters used (Case 1 and 2).

	ϵ	ρ	K	J	$\mathbf{y}^0$	$\boldsymbol{\lambda}_{\mathbf{p}}{}^0 \backslash \boldsymbol{\lambda}_{\mathbf{w}}{}^0$	$\breve{\varphi}_{\mathrm{p}}^0$	$\breve{\varphi}_{\mathrm{w}}^0$
Case 1	6e−3	4e−4	300	3	**0**	**0**	−9999	−9999
Case 2	6e−3	6e−4	300	5	**0**	**0**	−9999	−9999

(definitely not optimal) scenario where all pumps operate at all times is considered. As such, $\boldsymbol{\alpha}^0 = \mathbf{1}$ is used. Nonetheless, it is worth noting that we can use any feasible solution for the initialization.

4.2. Algorithm validation

In this subsection, the MIBC SD-GS-AL algorithm is validated and compared with the original SD-GS-AL algorithm and ADMM for three different load profiles, provided in Fig. 4.

4.2.1. Case 1: 13-node PDS and 11-node WDS

Case 1 presents the performance of the proposed MIBC SD-GS-AL algorithm when weaker coupling exists between PDS and WDS. Fig. 5 shows the convergence error (i.e., the difference between the upper and lower bounds) of the proposed MIBC SD-GS-AL algorithm for three different load profiles. The figure shows that the proposed MIBC SD-GS-AL algorithm converges for all three different load profiles (note that the zero convergence error refers to convergence). For the comparison with the original SD-GS-AL and ADMM, since the difference of boundary variables is used as a convergence criterion in the ADMM [32], the boundary error (i.e., the sum of the square of the difference of boundary variables for all time instances) has been utilized in this paper for consistency. Figs. 6, 7, and 8 compare the proposed MIBC SD-GS-AL algorithm with the original SD-GS-AL algorithm and ADMM for three different load profiles. The Figures show that the boundary error for both the original SD-GS-AL algorithm and ADMM fails to converge while that of the proposed MIBC SD-GS-AL algorithm converges.

4.2.2. Case 2: 37-node PDS and 36-node WDS

In Case 2, we study the performance of the proposed MIBC SD-GS-AL algorithm when stronger coupling exists between PDS and WDS. Fig. 9 shows the convergence error of the proposed MIBC SD-GS-AL algorithm for three different load profiles. The figure shows that the proposed MIBC SD-GS-AL algorithm converges for all three different load profiles. In addition, Figs. 10, 11, and 12 compare the proposed MIBC SD-GS-AL algorithm with the original SD-GS-AL algorithm and ADMM for three different load profiles. The Figures show that the boundary error for the original SD-GS-AL algorithm fails to converge while that of the proposed MIBC SD-GS-AL algorithm and ADMM converge. However, the ADMM converged to a sub-optimal solution, 63% more than the global optimal solution while the proposed MIBC SD-GS-AL algorithm converged to the global optimal solution for all three load profiles.

Table 4
Validation of relaxation technique.

	Method	Load profile	Objective value
Case 1	MINLP centralized	1	$ 10855.72
		2	$ 8333.41
		3	$ 10569.71
	MICP centralized	1	$ 10855.70
		2	$ 8333.42
		3	$ 10569.70
	MINLP decentralized	1	N/A
		2	N/A
		3	N/A
	MICP decentralized (MIBC SD-GS-AL)	1	$ 10855.74
		2	$ 8333.42
		3	$ 10569.72
Case 2	MINLP centralized	1	$ 7141.75
		2	$ 5479.18
		3	$ 6954.12
	MICP centralized	1	$ 7141.75
		2	$ 5479.22
		3	$ 6954.23
	MINLP decentralized	1	N/A
		2	N/A
		3	N/A
	MICP decentralized (MIBC SD-GS-AL)	1	$ 7141.75
		2	$ 5479.18
		3	$ 6954.11

4.2.3. Validation of relaxation technique

Both the original and MIBC SD-GS-AL algorithms need subproblems to be in the MICP form. Since neither the PDS nor WDS subproblems are originally mixed-integer convex, this paper leverages convex hulls relaxation for the PDS subproblem and convex and quasi-convex hulls relaxation for the WDS subproblem from [8]. Although there is no theoretical guarantee of the accuracy of the relaxed optimization problem,

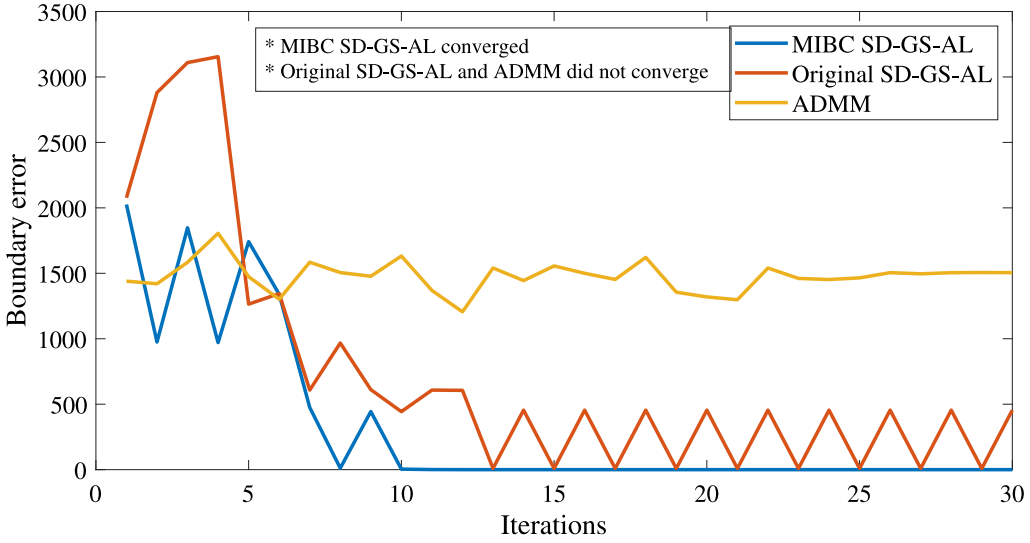


Fig. 6. Comparison of MIBC SD-GS-AL, original SD-GS-AL, and ADMM for load profile 1 (Case 1).

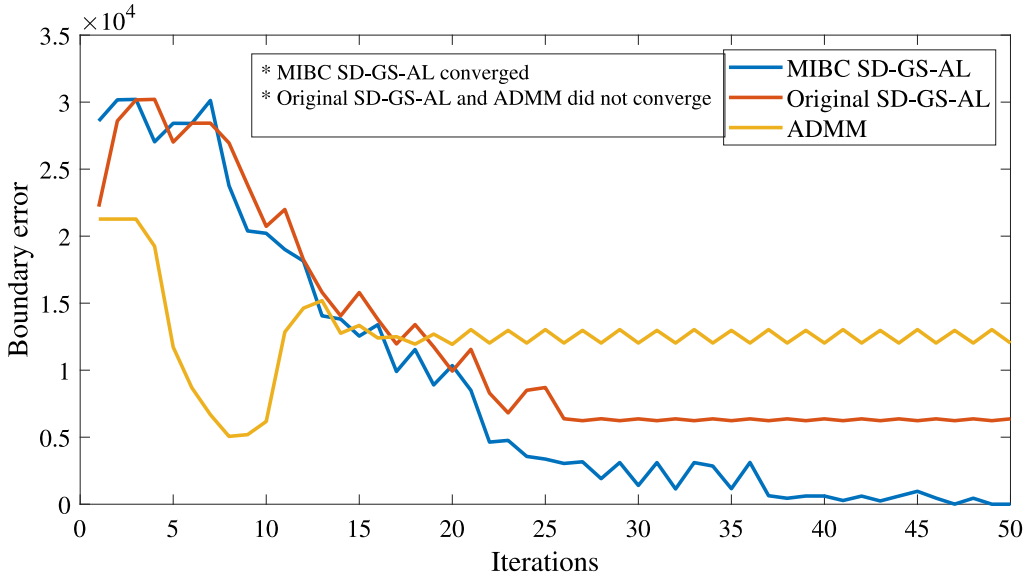


Fig. 7. Comparison of MIBC SD-GS-AL, original SD-GS-AL, and ADMM for load profile 2 (Case 1).

the convex (and quasi-convex) hull relaxations have been proven to be very tight and can provide feasible and optimal solutions to many cases, as shown in [8]. For the study of the accuracy of the relaxation techniques, we have compared the centralized mixed-integer non-linear program (MINLP) model, the centralized MICP model, the decentralized MICP model using MIBC SD-GS-AL, and the decentralized MINLP model in terms of objective value. The centralized MICP model refers to the model (9), while the centralized MINLP model refers to the same model (9) with the original non-linear and non-convex constraints (1b), (1i), (2d), and (2l). Note that constraint (2c) is also non-linear and non-convex. However, to model the *sign* function in (2c), a huge number of binary variables (=2*number of pipes*time periods) are needed. The centralized MINLP model with non-linear non-convex constraint (2c) did not terminate after more than 4 h. Hence, the centralized MINLP model includes non-linear and non-convex constraints (1b), (1i), (2d), and (2l) and convex relaxation (5) of (2c) in the Table 4. Note that the adopted convex relaxation (5) of (2c) does not require binary variables to convexify it. From the Table 4, it is seen that the objective value for all three models matches for all three load profiles in both test cases. Note that the MIBC SD-GS-AL algorithm is only applicable to MICP models (not MINLP models); therefore, results of MIBC SD-GS-AL are provided for MICP models only in the Table.

4.3. Engineering validation of the simulation results

In this subsection, the engineering validation of the simulation results is made. For brevity, results for only Load Profile 1 of both test cases are presented.

4.3.1. Case 1: 13-node PDS and 11-node WDS

Fig. 13 shows how the operation of a water pump in a WDS varies with the electricity price in a PDS. It is seen that the water pump operates when the electricity price is low. Figs. 13 and 14 illustrate that even when it has to operate during high electricity price periods, it consumes low power. As a result, the reservoir supplies none or a low amount of water to the network, and the tank supplies more (the positive tank flow indicates the tank supplying the network), as seen from Fig. 15. Moreover, during the low electricity price periods, the reservoir supplies more water, and the tank gets filled (the negative tank flow indicates the tank being filled), as depicted in Fig. 15. Unless

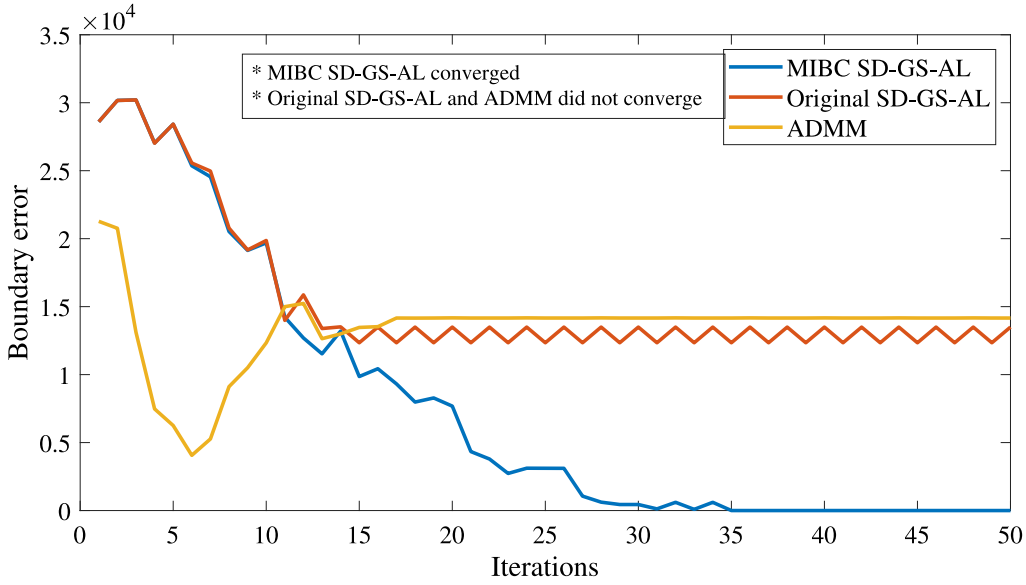


Fig. 8. Comparison of MIBC SD-GS-AL, original SD-GS-AL, and ADMM for load profile 3 (Case 1).

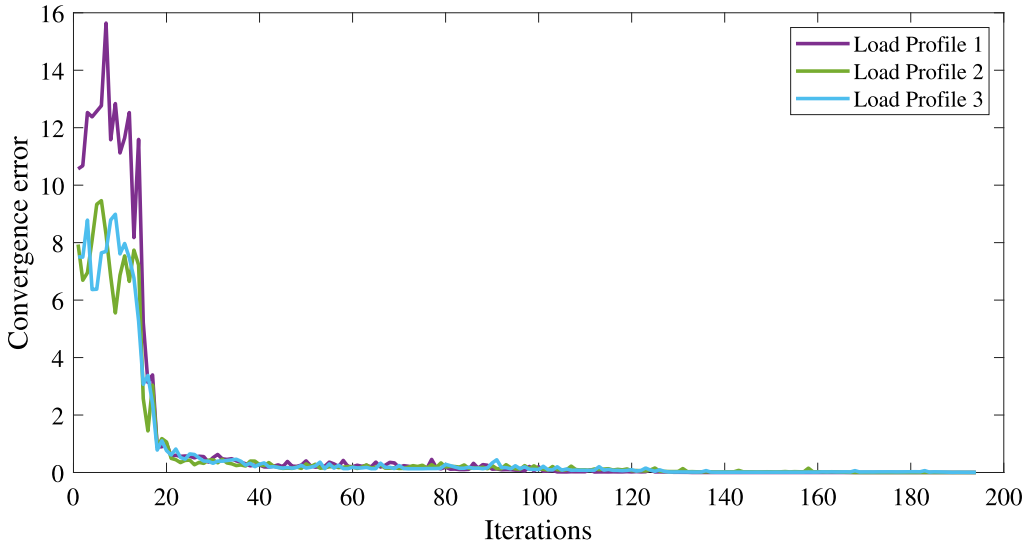


Fig. 9. Convergence error of MIBC SD-GS-AL for different load profiles (Case 2).

otherwise stated, cms refers to the water flow measured in cubic meters per second.

4.3.2. Case 2: 37-node PDS and 36-node WDS

Here, we make the engineering validation of the obtained results for Case 2. From Table 2, it can be observed that Pump 1 is a high-power-consuming pump. Therefore, Pump 1 operates when the electricity price is low while the less-power-consuming Pump 2 operates when the electricity price is high, as seen from Figs. 16 and 17. From Fig. 3, it is seen that Pump 1 supplies water to the network from the reservoir while Pump 2 supplies water to the network from the tank. Hence, at least one of the pumps has to operate always in order to supply water to the customers. Therefore, as seen from Figs. 17 and 18, the reservoir supplies water to the network, and the water tank gets filled when Pump 1 operates (during low electricity price periods) while the water tank gets emptied (the positive tank flow indicates the tank being emptied) when Pump 2 operates (during high electricity price periods).

5. Conclusion

This paper presents a MIBC SD-GS-AL algorithm for the coordination of PDS and WDS. Unlike the existing distributed/decentralized algorithms like ADMM [32] and the original SD-GS-AL algorithm [42], this paper deals with a unique situation: the sub-problems, in our case, have integer-dependent boundary variables. Therefore, a new decentralized algorithm was needed. The proposed MIBC SD-GS-AL algorithm has been shown to work in such a setting. Moreover, unlike existing algorithms, the proposed algorithm requires limited information exchange only, resulting in cost savings on communication channels and privacy preservation. Last but not least, the convergence and optimality of the proposed algorithm are guaranteed. The proposed algorithm was tested on two coupled PDS and WDS test cases. The test results show that the proposed MIBC algorithm converges to the optimal solutions while the original SD-GS-AL does not converge for both test cases. Moreover, the ADMM does not converge for the first test case while it converges to a sub-optimal solution, 63% more than the optimal solution for the second test case.

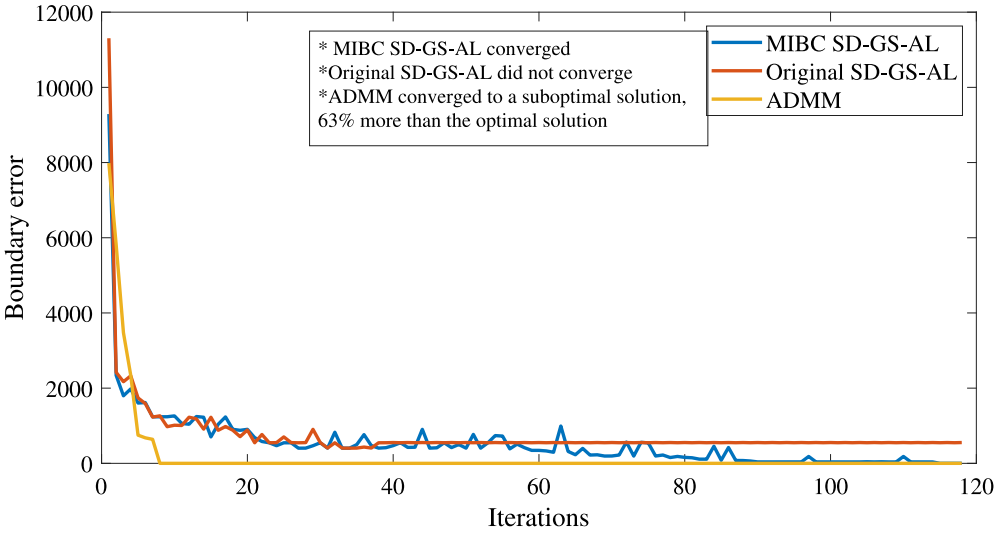


Fig. 10. Comparison of MIBC SD-GS-AL, original SD-GS-AL, and ADMM for load profile 1 (Case 2).

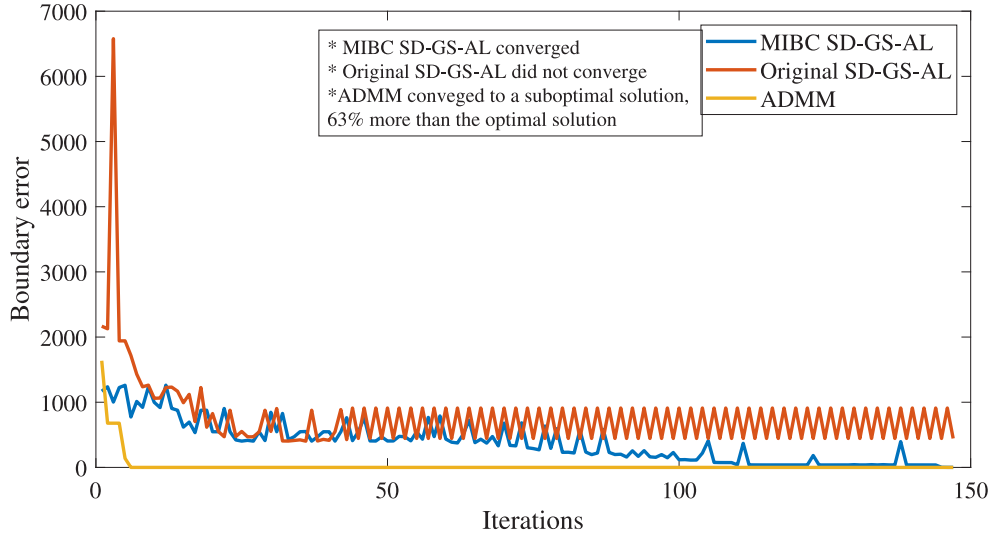


Fig. 11. Comparison of MIBC SD-GS-AL, original SD-GS-AL, and ADMM for load profile 2 (Case 2).

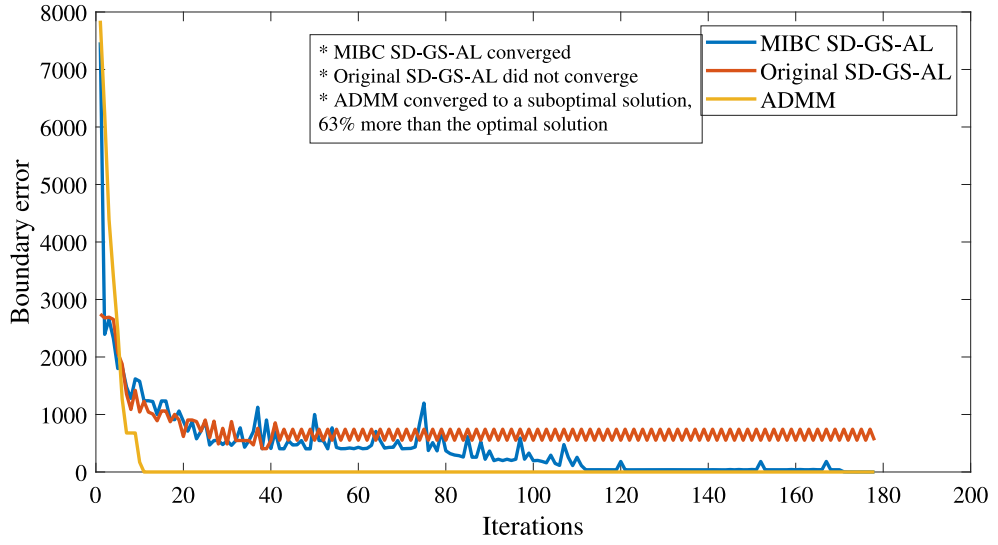


Fig. 12. Comparison of MIBC SD-GS-AL, original SD-GS-AL, and ADMM for load profile 3 (Case 2).

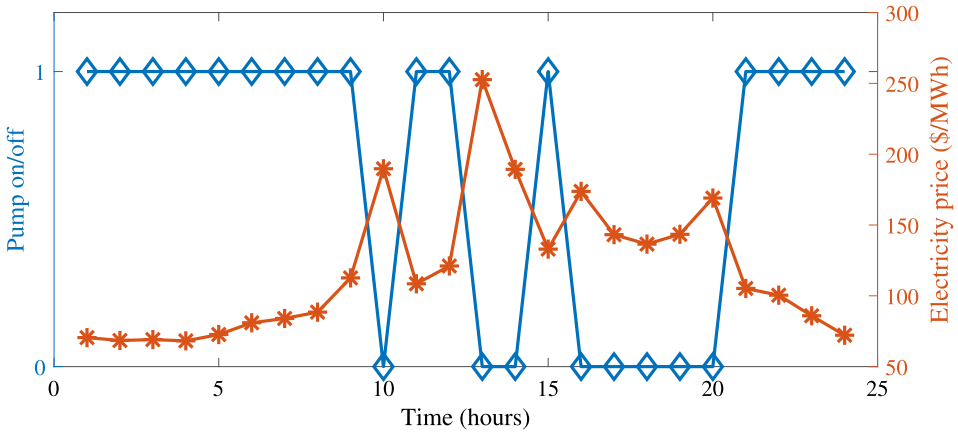


Fig. 13. Electricity price vs. pump operation (Case 1).

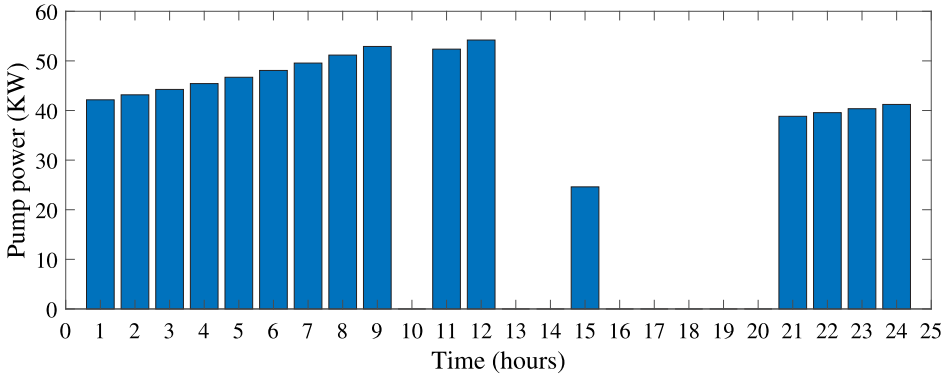


Fig. 14. Consumption of electric power by the water pump (Case 1).

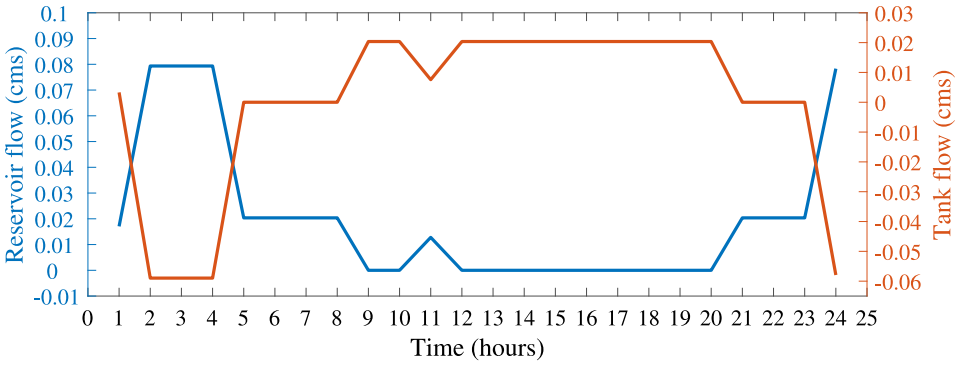


Fig. 15. Tank and reservoir operation (Case 1).

CRediT authorship contribution statement

Santosh Sharma: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Qifeng Li:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

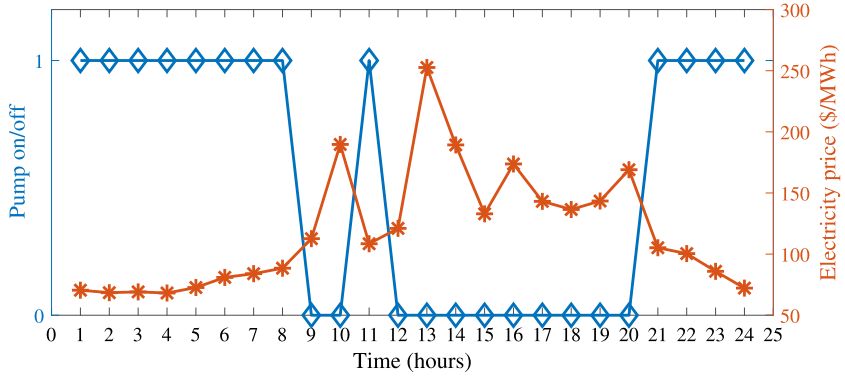
Data availability

Data will be made available on request.

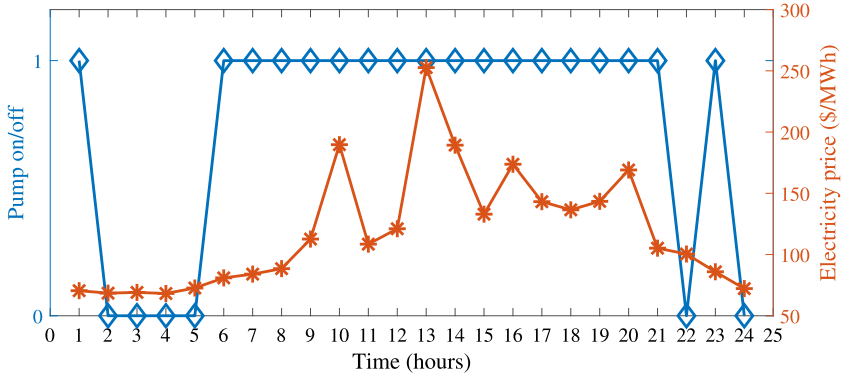
Appendix. Optimality and convergence

In Ref. [42], the original SD-GS-AL algorithm has been proved for MIP subproblems that share continuous boundary variables. In this subsection, we show that optimality and convergence still hold for MIP subproblems with discontinuous boundary variables. It is worth noting that the following key features ensure that the proposed algorithm converges to the global optimal solution of (9):

1. The Lagrangian upper bound, computed using continuous subproblems as stated in Step 3, is a global upper bound.

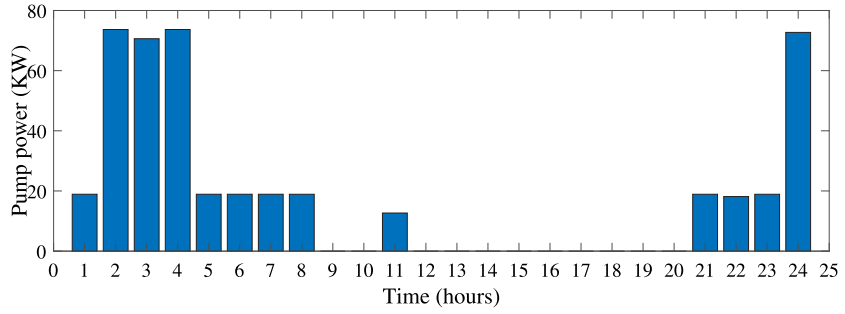


(a) Electricity price vs. Pump 1.

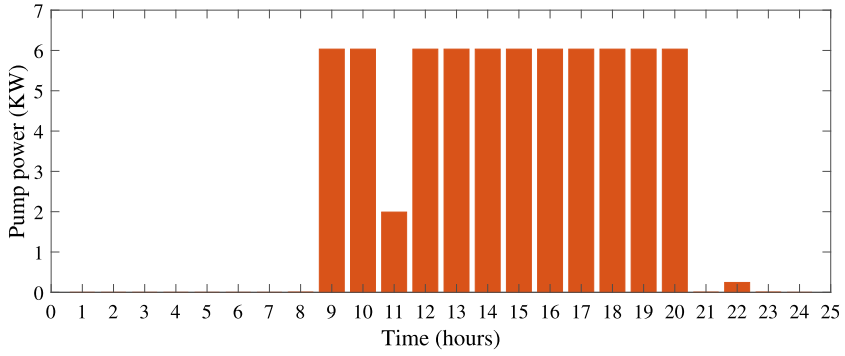


(b) Electricity price vs. Pump 2.

Fig. 16. Electricity price vs. pumps operation (Case 2).



(a) Pump 1



(b) Pump 2.

Fig. 17. Power consumption by water pumps (Case 2).

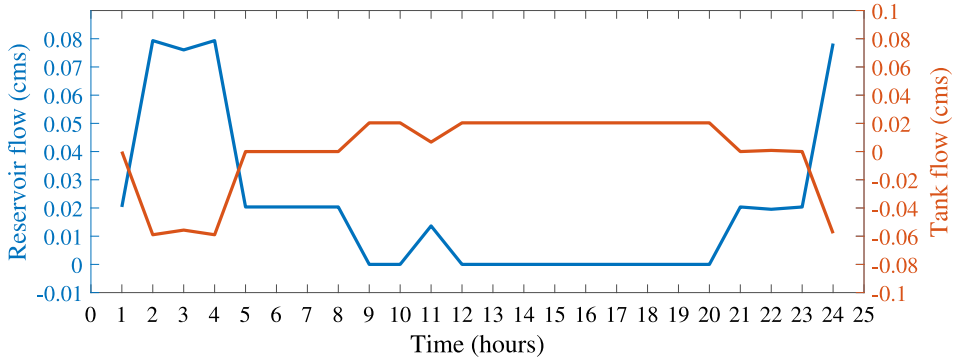


Fig. 18. Tank and reservoir operation (Case 2).

2. For the computation of the Lagrangian lower bound, mixed-integer convex sets $\mathcal{X}_{\mathrm{p}}$ and $\mathcal{X}_{\mathrm{w}}$ are utilized. However, as mentioned in [50], minimizing linear objective function over mixed-integer convex sets $\mathcal{X}_{\mathrm{p}}$ and $\mathcal{X}_{\mathrm{w}}$ is equivalent to minimizing linear objective function over convex hulls sets, $\mathrm{CH}(\mathcal{X}_{\mathrm{p}})$ and $\mathrm{CH}(\mathcal{X}_{\mathrm{w}})$. Therefore, the Lagrangian lower bound obtained is a global lower bound.
3. The Lagrangian multipliers are computed based on the values of variables obtained from continuous subproblems; see Step 6.

Now, we show that the sequence $\{(\boldsymbol{x}^k, \boldsymbol{y}^k)\}$ generated by Algorithm 1 converges to the global optimal solution of the centralized implementation of MICP OPF and OPS subproblems (9) as $k \to \infty$. This is divided into two parts. Part 1 proves the convergence, while Part 2 verifies the optimality. For brevity and convenience, we adopt the following definitions:

$$
\begin{aligned}
&L_{\rho} := L_{\rho,\mathrm{p}} + L_{\rho,\mathrm{w}},\\
&\varphi^{c}(\boldsymbol{x},\lambda) := \varphi^{c}_{\mathrm{p}}\left(\boldsymbol{x}_{\mathbf{p}},\lambda_{\mathbf{p}}\right) + \varphi^{c}_{\mathrm{w}}\left(\boldsymbol{x}_{\mathbf{w}},\lambda_{\mathbf{w}}\right),\\
&\varphi(\boldsymbol{x},\lambda) := \varphi_{\mathrm{p}}\left(\boldsymbol{x}_{\mathbf{p}},\lambda_{\mathbf{p}}\right) + \varphi_{\mathrm{w}}\left(\boldsymbol{x}_{\mathbf{w}},\lambda_{\mathbf{w}}\right),\\
&\hat{\varphi}(\boldsymbol{x},\boldsymbol{y},\lambda) := \hat{\varphi}_{\mathrm{p}}\left(\boldsymbol{x}_{\mathbf{p}},\boldsymbol{y},\lambda_{\mathbf{p}}\right) + \hat{\varphi}_{\mathrm{w}}\left(\boldsymbol{x}_{\mathbf{w}},\boldsymbol{y},\lambda_{\mathbf{w}}\right),\\
&f(\boldsymbol{x},\boldsymbol{y}) := \text{(9a)}.\\
&\mathcal{X} := \mathcal{X}_{\mathrm{p}} \cup \mathcal{X}_{\mathrm{w}},\\
&\boldsymbol{x}^k := (\boldsymbol{x}^k_{\mathbf{p}}, \boldsymbol{x}^k_{\mathbf{w}})\,\text{(vector concatenation)},\\
&\boldsymbol{\alpha}^k_{\mathbf{a}} := (\boldsymbol{\alpha}^k_{\mathbf{p}}, \boldsymbol{\alpha}^k_{\mathbf{w}}),\\
&(\boldsymbol{x}-\boldsymbol{y}) := \left(\left(\boldsymbol{y}-\boldsymbol{p}^{\mathbf{l}}\right),\left(\boldsymbol{p}^{\mathbf{p}}-\boldsymbol{y}\right)\right).
\end{aligned}
$$

For limit point $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ of the sequence $\{(\boldsymbol{x}^k,\boldsymbol{y}^k)\}$ generated by Algorithm 1, the convergence condition at $\boldsymbol{x} \in \mathcal{X}$ is defined as [42]:

$$L'_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y};s) \geq 0 \quad \text{for all } s \in X - \{\boldsymbol{x}\}, \tag{A.1}$$

where $L'_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y};s) = lim_{\beta\to 0}\frac{L(\boldsymbol{x}+\beta s,\boldsymbol{y})-L(\boldsymbol{x},\boldsymbol{y})}{\beta}$. Furthermore, the Direction Related Assumption is given as follows: for any iteration k, s^k is chosen such that $\boldsymbol{x}^k + s^k \in \mathcal{X}$ and $L'_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y};s) \geq 0$. Note that s^k is a gradient of $\boldsymbol{x}^k$.

Part 1: The sequence $\{(\boldsymbol{x}^k,\boldsymbol{y}^k)\}$ generated by Algorithm 1 always converges to the limit point $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$.

Here, we prove that the limit point $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ of the sequence $\{(\boldsymbol{x}^k,\boldsymbol{y}^k)\}$ of feasible solutions to the problems (12) and (13) satisfies the convergence condition (A.1). According to the Armijo rule [51], we have

$$\frac{L\left(\boldsymbol{x}^k+\beta^k s^k,\boldsymbol{y}^k\right) - L\left(\boldsymbol{x}^k,\boldsymbol{y}^k\right)}{\beta^k} \leq \sigma L'_{\boldsymbol{x}}\left(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k\right) \tag{A.2}$$

for any $\sigma \in (0,1)$. As $L'_{\boldsymbol{x}}(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k) < 0$ according to the Direction Related Assumption (defined above) and $\beta^k \geq 0$, above expression can be rewritten as $L(\boldsymbol{x}^k+\beta^k s^k,\boldsymbol{y}^k) < L(\boldsymbol{x}^k,\boldsymbol{y}^k)$. We also have $L(\boldsymbol{x}^{k+1},\boldsymbol{y}^{k+1}) \leq L(\boldsymbol{x}^k+\beta^k s^k,\boldsymbol{y}^k) < L(\boldsymbol{x}^k,\boldsymbol{y}^k)$ and $L(\boldsymbol{x}^{k+1},\boldsymbol{y}^{k+1}) < L(\boldsymbol{x}^k,\boldsymbol{y}^k)$. Also, L is bounded from below, we have $\lim_{k\to\infty} L(\boldsymbol{x}^k,\boldsymbol{y}^k) = \bar{L} > -\infty$. Hence, we have

$$\lim_{k\to\infty} L\left(\boldsymbol{x}^{k+1},\boldsymbol{y}^{k+1}\right) - L\left(\boldsymbol{x}^k,\boldsymbol{y}^k\right) = 0.$$

Furthermore,

$$\lim_{k\to\infty} L\left(\boldsymbol{x}^k+\beta^k s^k,\boldsymbol{y}^k\right) - L\left(\boldsymbol{x}^k,\boldsymbol{y}^k\right) = 0. \tag{A.3}$$

For the sake of contradiction, we assume that $\lim_{k\to\infty}(\boldsymbol{x}^k,\boldsymbol{y}^k) = (\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ does not satisfy the convergence condition. From the definition of gradient related assumption [52], we have

$$\limsup_{k\to\infty} L'_{\boldsymbol{x}}\left(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k\right) < 0. \tag{A.4}$$

Hence, in conclusion, $\lim_{k\to\infty}\beta^k = 0$. From the Armijo rule, after a certain iteration $k \geq \bar{k}$, we can define $\{\bar{\beta}^k\}$, $\bar{\beta}^k = \beta^k/\gamma$ for some γ, where $\bar{\beta}^k \leq 1$ and we have

$$\sigma L'_{\boldsymbol{x}}\left(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k\right) < \frac{L\left(\boldsymbol{x}^k+\bar{\beta}^k s^k,\boldsymbol{y}^k\right) - L\left(\boldsymbol{x}^k,\boldsymbol{y}^k\right)}{\bar{\beta}^k}. \tag{A.5}$$

If we apply the mean value theorem to the right side of the above expression, for some $\widetilde{\beta}^k \in [0,\bar{\beta}^k]$, we have

$$\sigma L'_{\boldsymbol{x}}\left(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k\right) < L'_{\boldsymbol{x}}\left(\boldsymbol{x}^k+\widetilde{\beta}^k s^k,\boldsymbol{y}^k;s^k\right). \tag{A.6}$$

Moreover, $\limsup_{k\to\infty} L'_{\boldsymbol{x}}(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k) < 0$, and if we take a limit point $\bar{s}$ of $\{s^k\}$ such that $L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}},\bar{s}) < 0$. Also, we have, $\lim_{k\to\infty,k\in\mathcal{K}} L'_{\boldsymbol{x}}(\boldsymbol{x}^k,\boldsymbol{y}^k;s^k) = L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}};\bar{s})$ and $\lim_{k\to\infty,k\in\mathcal{K}} L'_{\boldsymbol{x}}(\boldsymbol{x}^k+\widetilde{\beta}^k s^k,\boldsymbol{y}^k;s^k) = L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}};\bar{s})$. From these two factors, we can infer that $L'_{\boldsymbol{x}}(\boldsymbol{x},\boldsymbol{y};s)$ is continuous. Now, from expression (A.6), we have

$$\sigma L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}};\bar{s}) \leq L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}};\bar{s}) \quad\Longrightarrow\quad 0 \leq (1-\sigma)L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}};\bar{s}).$$

Since $(1-\sigma) > 0$, $L'_{\boldsymbol{x}}(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}};\bar{s}) < 0$ which is a contradiction. Therefore, the limit point $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ of the sequence $\{(\boldsymbol{x}^k,\boldsymbol{y}^k)\}$ i.e., $\lim_{k\to\infty}(\boldsymbol{x}^k,\boldsymbol{y}^k) = (\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ satisfies the convergence condition, which means algorithm 1 always converges.

Part 2: The limit point $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$ of the sequence $\{(\boldsymbol{x}^k,\boldsymbol{y}^k)\}$ generated by Algorithm 1 is a global optimal solution.

From *Part 1*, we have that the algorithm converges to the limit point $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$. In other words, the algorithm produces a solution, $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$. Here, we establish the global optimality of the solution $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}})$. The optimality conditions (KKT conditions) associated with the $(\bar{\boldsymbol{x}},\bar{\boldsymbol{y}}) \in \mathrm{argmin}_{\boldsymbol{x},\boldsymbol{y}}\left\{L_{\rho}(\boldsymbol{x},\boldsymbol{y},\lambda) : \boldsymbol{\alpha}_{\mathbf{a}} \in \boldsymbol{\alpha}^k_{\mathbf{a}}\right\}$ is given as follows:

$$
\begin{aligned}
\boldsymbol{\Phi}_{\boldsymbol{x}} :=& \begin{bmatrix} \nabla f(\bar{\boldsymbol{x}}) + [\lambda + \rho(\bar{\boldsymbol{x}}-\bar{\boldsymbol{y}})]^{\top}\mathbf{1} \end{bmatrix}^{\top}\begin{bmatrix} \boldsymbol{x}-\bar{\boldsymbol{x}} \end{bmatrix}\\
&\geq 0 \quad \text{for all } \boldsymbol{x} \in \mathrm{CH}(\mathcal{X}).
\end{aligned}
$$

The above optimality condition can also be written as:

$$\min_{\boldsymbol{x}}\left\{\boldsymbol{\Phi}_{\boldsymbol{x}}\right\} = 0.$$

The above expression can be re-written in terms of $\varphi(\bar{\boldsymbol{x}},\lambda+\rho(\bar{\boldsymbol{x}}-\bar{\boldsymbol{y}}))$ as follows:

$$\breve{\varphi}(\bar{\boldsymbol{x}},\lambda+\rho(\bar{\boldsymbol{x}}-\bar{\boldsymbol{y}})) = f(\bar{\boldsymbol{x}}) + \lambda^{\top}\bar{\boldsymbol{x}} + \rho\left\|\bar{\boldsymbol{x}}-\bar{\boldsymbol{y}}\right\|_2^2$$

$$= L_\rho(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \lambda) + \frac{\rho}{2} \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2^2 .$$

Note that

$$\varphi(\boldsymbol{x}^k, \lambda) := \min_{\boldsymbol{x}} \left\{ f(\boldsymbol{x}^k) + \nabla_{\boldsymbol{x}} f(\boldsymbol{x}^k)^\top (\boldsymbol{x} - \boldsymbol{x}^k) + \lambda^\top \boldsymbol{x} : \boldsymbol{x} \in \mathcal{X} \right\}.$$

$$\hat{\varphi}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \lambda) := L_\rho(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \lambda) + \frac{\rho}{2} \|\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}\|_2^2 .$$

Hence,

$$\varphi(\lambda + \rho(\bar{\boldsymbol{x}} - \bar{\boldsymbol{y}}), \bar{\boldsymbol{x}}) = \hat{\varphi}(\bar{\boldsymbol{x}}, \bar{\boldsymbol{y}}, \lambda) . \tag{A.7}$$

The expression (A.7) implies that the upper and lower bounds of the Lagrangian function converge as $k \to \infty$. In other words, Algorithm 1 converges to the global optimal solution [53].

References

[1] Frank S, Steponavice I, Rebennack S. Optimal power flow: A bibliographic survey I. Energy Syst 2012;3(3):221–58.

[2] Jowitt PW, Germanopoulos G. Optimal pump scheduling in water-supply networks. J Water Res Plan Manag 1992;118(4):406–22.

[3] McCormick G, Powell R. Optimal pump scheduling in water supply systems with maximum demand charges. J Water Res Plan Manag 2003;129(5):372–9.

[4] de Oliveira GC, Bertone E, Stewart RA. Optimisation modelling tools and solving techniques for integrated precinct-scale energy–water system planning. Appl Energy 2022;318:119190. http://dx.doi.org/10.1016/j.apenergy.2022.119190, URL https://www.sciencedirect.com/science/article/pii/S0306261922005578.

[5] Zhang W, Valencia A, Gu L, Zheng QP, Chang N-B. Integrating emerging and existing renewable energy technologies into a community-scale microgrid in an energy-water nexus for resilience improvement. Appl Energy 2020;279:115716. http://dx.doi.org/10.1016/j.apenergy.2020.115716, URL https://www.sciencedirect.com/science/article/pii/S0306261920312095.

[6] Chen C, Zhang X, Zhang H, Cai Y, Wang S. Managing water-energy-carbon nexus in integrated regional water network planning through graph theory-based bi-level programming. Appl Energy 2022;328:120178. http://dx.doi.org/10.1016/j.apenergy.2022.120178, URL https://www.sciencedirect.com/science/article/pii/S0306261922014350.

[7] Li Q, Yu S, Al-Sumaiti A, Turitsyn K. Modeling and co-optimization of a micro water-energy nexus for smart communities. In: 2018 IEEE PES innovative smart grid technologies conference europe. IEEE; 2018, p. 1–5.

[8] Li Q, Yu S, Al-Sumaiti AS, Turitsyn K. Micro water–energy nexus: Optimal demand-side management and quasi-convex hull relaxation. IEEE Trans Control Netw Syst 2018;6(4):1313–22.

[9] Zamzam AS, Dall'Anese E, Zhao C, Taylor JA, Sidiropoulos ND. Optimal water–power flow-problem: Formulation and distributed optimal solution. IEEE Trans Control Netw Syst 2018;6(1):37–47.

[10] Oikonomou K, Parvania M. Optimal coordinated operation of interdependent power and water distribution systems. IEEE Trans Smart Grid 2020;11(6):4784–94.

[11] Goodarzi M, Li Q. Evaluate the capacity of electricity-driven water facilities in small communities as virtual energy storage. Appl Energy 2022;309:118349.

[12] Nguyen HT, Al-Sumaiti AS, Turitsyn K, Li Q, El Moursi MS. Further optimized scheduling of micro grids via dispatching virtual electricity storage offered by deferrable power-driven demands. IEEE Trans Power Syst 2020;35(5):3494–505.

[13] Menke R, Abraham E, Parpas P, Stoianov I. Demonstrating demand response from water distribution system through pump scheduling. Appl Energy 2016;170:377–87.

[14] Zhang K, Zhang Y, Xi S, Liu J, Li J, Hou S, et al. Multi-objective optimization of energy-water nexus from spatial resource reallocation perspective in China. Appl Energy 2022;314:118919. http://dx.doi.org/10.1016/j.apenergy.2022.118919, URL https://www.sciencedirect.com/science/article/pii/S0306261922003415.

[15] Moazeni F, Khazaei J, Asrari A. Step towards energy-water smart microgrids; Buildings thermal energy and water demand management embedded in economic dispatch. IEEE Trans Smart Grid 2021;12(5):3680–91. http://dx.doi.org/10.1109/TSG.2021.3068053.

[16] Moazeni F, Khazaei J. Dynamic economic dispatch of islanded water-energy microgrids with smart building thermal energy management system. Appl Energy 2020;276:115422. http://dx.doi.org/10.1016/j.apenergy.2020.115422, URL https://www.sciencedirect.com/science/article/pii/S030626192030934X.

[17] Moazeni F, Khazaei J, Pera Mendes JP. Maximizing energy efficiency of islanded micro water-energy nexus using co-optimization of water demand and energy consumption. Appl Energy 2020;266:114863. http://dx.doi.org/10.1016/j.apenergy.2020.114863, URL https://www.sciencedirect.com/science/article/pii/S0306261920303755.

[18] Giudici F, Castelletti A, Garofalo E, Giuliani M, Maier HR. Dynamic, multi-objective optimal design and operation of water-energy systems for small, off-grid islands. Appl Energy 2019;250:605–16. http://dx.doi.org/10.1016/j.apenergy.2019.05.084, URL https://www.sciencedirect.com/science/article/pii/S0306261919309316.

[19] Liu Y, Mauter MS. Assessing the demand response capacity of U.S. drinking water treatment plants. Appl Energy 2020;267:114899. http://dx.doi.org/10.1016/j.apenergy.2020.114899, URL https://www.sciencedirect.com/science/article/pii/S0306261920304116.

[20] Golmohamadi H, Asadi A. A multi-stage stochastic energy management of responsive irrigation pumps in dynamic electricity markets. Appl Energy 2020;265:114804. http://dx.doi.org/10.1016/j.apenergy.2020.114804, URL https://www.sciencedirect.com/science/article/pii/S0306261920303160.

[21] Moazeni F, Khazaei J. Co-optimization of wastewater treatment plants interconnected with smart grids. Appl Energy 2021;298:117150. http://dx.doi.org/10.1016/j.apenergy.2021.117150, URL https://www.sciencedirect.com/science/article/pii/S0306261921005808.

[22] Diaz C, Ruiz F, Patino D. Modeling and control of water booster pressure systems as flexible loads for demand response. Appl Energy 2017;204:106–16.

[23] Elsir M, Al-Sumaiti AS, El Moursi MS, Al-Awami AT. Coordinating the day-ahead operation scheduling for demand response and water desalination plants in smart grid. Appl Energy 2023;335:120770. http://dx.doi.org/10.1016/j.apenergy.2023.120770, URL https://www.sciencedirect.com/science/article/pii/S0306261923001344.

[24] Moazeni F, Khazaei J. Optimal energy management of water-energy networks via optimal placement of pumps-as-turbines and demand response through water storage tanks. Appl Energy 2021;283:116335. http://dx.doi.org/10.1016/j.apenergy.2020.116335, URL https://www.sciencedirect.com/science/article/pii/S0306261920317189.

[25] Rodriguez-Garcia L, Hosseini MM, Mosier TM, Parvania M. Resilience analytics for interdependent power and water distribution systems. IEEE Trans Power Syst 2022;37(6):4244–57. http://dx.doi.org/10.1109/TPWRS.2022.3149463.

[26] Li J, Xu Y, Wang Y, Li M, He J, Liu C-C, et al. Resilience-motivated distribution system restoration considering electricity-water-gas interdependency. IEEE Trans Smart Grid 2021;12(6):4799–812. http://dx.doi.org/10.1109/TSG.2021.3105234.

[27] Zhao P, Li S, Hu PJ-H, Gu C, Cao Z, Xiang Y. Managing water-energy-carbon nexus for urban areas with ambiguous moment information. IEEE Trans Power Syst 2022;1–14. http://dx.doi.org/10.1109/TPWRS.2022.3214189.

[28] Elsir M, Al-Awami AT, Antar MA, Oikonomou K, Parvania M. Risk-based operation coordination of water desalination and renewable-rich power systems. IEEE Trans Power Syst 2022;1. http://dx.doi.org/10.1109/TPWRS.2022.3174565.

[29] Yang Y, Li Z, Mandapaka PV, Lo EY. Risk-averse restoration of coupled power and water systems with small pumped-hydro storage and stochastic rooftop renewables. Appl Energy 2023;339:120953. http://dx.doi.org/10.1016/j.apenergy.2023.120953, URL https://www.sciencedirect.com/science/article/pii/S0306261923003173.

[30] Stuhlmacher A, Mathieu JL. Water distribution networks as flexible loads: A chance-constrained programming approach. Electr Power Syst Res 2020;188:106570.

[31] Li Q, Yang L, Lin S. Coordination strategy for decentralized reactive power optimization based on a probing mechanism. IEEE Trans Power Syst 2014;30(2):555–62.

[32] Boyd S, Parikh N, Chu E. Distributed optimization and statistical learning via the alternating direction method of multipliers. Now Publishers Inc; 2011.

[33] Wei E, Ozdaglar A. Distributed alternating direction method of multipliers. In: 2012 IEEE 51st IEEE conference on decision and control. IEEE; 2012, p. 5445–50.

[34] Zhao P, Gu C, Cao Z, Xie D, Teng F, Li J, et al. A cyber-secured operation for water-energy nexus. IEEE Trans Power Syst 2021;36(4):3105–17. http://dx.doi.org/10.1109/TPWRS.2020.3043757.

[35] Zhao P, Gu C, Cao Z, Ai Q, Xiang Y, Ding T, Lu X, et al. Water-energy Nexus management for power systems. IEEE Trans Power Syst 2021;36(3):2542–54. http://dx.doi.org/10.1109/TPWRS.2020.3038076.

[36] Ayyagari KS, Gatsis N. Optimal pump scheduling in multi-phase distribution networks using benders decomposition. Electr Power Syst Res 2022;212:108584. http://dx.doi.org/10.1016/j.epsr.2022.108584, URL https://www.sciencedirect.com/science/article/pii/S0378779622006691.

[37] Michelena N, Park H, Papalambros PY. Convergence properties of analytical target cascading. AIAA J 2003;41(5):897–905.

[38] Tosserams S, Etman L, Papalambros P, Rooda J. An augmented Lagrangian relaxation for analytical target cascading using the alternating direction method of multipliers. Struct Multidisc Optim 2006;31(3):176–89.

[39] Losi A, Russo M. On the application of the auxiliary problem principle. J Optim Theory Appl 2003;117(2):377–96.

[40] Bürger M, Notarstefano G, Allgöwer F. A polyhedral approximation framework for convex and robust distributed optimization. IEEE Trans Automat Control 2014;59(2):384–95. http://dx.doi.org/10.1109/TAC.2013.2281883.

[41] Houska B, Frasch J, Diehl M. An augmented Lagrangian based algorithm for distributed nonconvex optimization. SIAM J Optim 2016;26(2):1101–27.

[42] Boland N, Christiansen J, Dandurand B, Eberhard A, Oliveira F. A parallelizable augmented Lagrangian method applied to large-scale non-convex-constrained optimization problems. Math Program 2019;175(1):503–36.
[43] Sharma S, Huang Q, Tbaileh A, Li Q. Scenario-based analysis for disaster-resilient restoration of distribution systems. In: 2019 north american power symposium. 2019, p. 1–6. http://dx.doi.org/10.1109/NAPS46351.2019.9000318.
[44] Baran M, Wu FF. Optimal sizing of capacitors placed on a radial distribution system. IEEE Trans Power Deliv 1989;4(1):735–43.
[45] Li Q, Ayyanar R, Vittal V. Convex optimization for DES planning and operation in radial distribution systems with high penetration of photovoltaic resources. IEEE Trans Sustain Energy 2016;7(3):985–95.
[46] Singh MK, Kekatos V. Optimal scheduling of water distribution systems. IEEE Trans Control Netw Syst 2019.
[47] Li Q, Vittal V. Convex hull of the quadratic branch AC power flow equations and its application in radial distribution networks. IEEE Trans Power Syst 2017;33(1):839–50.
[48] Kersting WH. Radial distribution test feeders. IEEE Trans Power Syst 1991;6(3):975–85.
[49] Rossman LA, Clark RM, Grayman WM. Modeling chlorine residuals in drinking-water distribution systems. J Environ Eng 1994;120(4):803–20.
[50] Lubin M, Martin K, Petra CG, Sandıkçı B. On parallelizing dual decomposition in stochastic integer programming. Oper Res Lett 2013;41(3):252–8.
[51] Quarteroni A, Sacco R, Saleri F. Numerical mathematics . Vol. 37. Springer Science & Business Media; 2010.
[52] Bertsekas DP. Nonlinear programming. J Oper Res Soc 1997;48(3):334.
[53] Conn AR, Gould NI, Toint P. A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds. SIAM J Numer Anal 1991;28(2):545–72.