Water Research X 30 (2026) 100479

Contents lists available at ScienceDirect

Water Research X

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

Neural network-informed Optimal Water Flow problem: Modeling, algorithm, and benchmarking

A. Belmondo Bianchi *, H.H.M. Rijnaarts, S. Shariat Torbaghan

Environmental Technology, Wageningen University and Research, Building 118, Bornse Weilanden 9, Wageningen, 6708 WG, Gelderland, The Netherlands

ARTICLE INFO

Keywords:
Water distribution network
Mathematical optimization
Input convex neural network
Water–energy nexus

ABSTRACT

Water distribution networks comprise interconnected components such as pipes, tanks, and pumps, whose hydraulic behavior is inherently nonlinear and nonconvex. Modeling head loss in pipes and pump performance curves is a major challenge for optimization-based planning and operations. These challenges arise, for instance, when solving the Optimal Water Flow (OWF) problem, which aims to determine pump schedules that minimize energy costs while satisfying hydraulic and operational constraints. While various approximation techniques exist, they often lack sufficient accuracy, raising concerns about their reliability in practice. To address this, we propose a hybrid approach that integrates deep learning with mathematical optimization to solve the OWF problem. We design a modified Input Convex Neural Network (ICNN) capable of capturing complex nonlinear relationships, focusing on pipe friction losses and pump hydraulics. To ensure tractable optimization, we introduce a novel regularization that enforces input convexity, enabling neural network inference to be reformulated as a linear program. This convex approximation is embedded into the OWF formulation, enabling end-to-end optimization with standard solvers. Empirical results demonstrate significant improvements in accuracy and scalability over existing OWF approximations, offering a practical tool for cost-effective, energy-efficient water distribution management.

1. Introduction

1.1. Background and motivation

Access to clean drinking water is essential for public health, economic growth, and societal well-being. Global water consumption has increased over fivefold in the past century and is projected to continue rising across multiple sectors (Boretti and Rosa, 2019). Stressors like population growth, urbanization, and climate change-induced droughts intensify competition for water resources, leading to significant economic, environmental, and social impacts (Wada et al., 2016; Kavya et al., 2023). Water utilities face increasing challenges and uncertainties, even in water-rich regions, in ensuring a reliable and safe supply for agriculture, industry, and human consumption due to technological, regulatory, and financial constraints (Karimidastenaei et al., 2022). A key emerging challenge is the rising energy cost of drinking water treatment and distribution, with pumps being a major operational expense (Stuhlmacher and Mathieu, 2020). Furthermore, the growing penetration of renewable energy sources (RES) has increased electricity price volatility. Consequently, there is a growing need to coordinate water utility operations with dynamic electricity pricing. Advanced optimization strategies, such as electricity price-driven scheduling of water network operations, offer significant potential for improving energy efficiency and reducing costs (Singh and Kekatos, 2019). This mathematical optimization problem is known as Optimal Water Flow (OWF) (Zamzam et al., 2018; Guo and Summers, 2020; Ayyagari et al., 2021).

In the OWF problem, the objective is to minimize pumping energy costs by routing water from sources to demands under dynamic electricity prices. Pump hydraulic constraints are crucial as pumps provide pressure to overcome elevation differences and friction losses, thus dictating energy consumption. The OWF problem is critical for cost reduction and efficiency in real-world applications (Vieira et al., 2020). However, solving OWF is computationally challenging due to the nonlinear and nonconvex relationship between flow and pressure. This complexity intensifies with variable speed pumps (VSPs), where the relationship among flow, head, motor speed, and power consumption is also nonlinear and nonconvex (Candelieri et al., 2018). While VSPs offer greater operational flexibility and energy efficiency than fixed-speed pumps (FSPs), they introduce additional

* Corresponding author.
E-mail address: alessio.belmondobianchidilavagna@wur.nl (A. Belmondo Bianchi).

https://doi.org/10.1016/j.wroa.2025.100479
Received 10 October 2025; Received in revised form 29 December 2025; Accepted 29 December 2025
Available online 12 January 2026

computational challenges due to their complex, nonconvex hydraulic characteristics. Different approximation methods have been developed to address various forms of nonconvexities in solving OWF, balancing accuracy and computational efficiency. Nevertheless, further advancements are needed to improve generalization, accuracy, and scalability.

1.2. Research gaps

The optimization of Water Distribution Networks (WDNs) has been a longstanding focus in mathematical programming and operations research, drawing significant attention from researchers and practitioners (D'Ambrosio et al., 2015). One of the key challenges in WDN optimization problems, such as the OWF problem, is the high state-space dimensionality, resulting from the spatial and temporal interdependencies of components like pipes, pumps, and tanks. Moreover, these problems involve nonlinear constraints related to pump hydraulics and friction losses, as well as discrete design and operational variables, including pump on–off statuses and optimal pipeline sizing and placement (Marchi et al., 2014). Consequently, these problems fall into the category of Nonlinear Programming (NLP) or Mixed-Integer Nonlinear Programming (MINLP) and are classified as NP-hard, making it challenging to obtain reliable solutions within practical time frames for real-world applications (Singh and Kekatos, 2020). Solving such problems to global or even local optimality is computationally demanding and can easily become intractable (Nerantzis et al., 2020; Martínez-Bahena et al., 2018).

To mitigate the computational challenges of WDN optimization, various approximation techniques have been developed, including convex relaxation and piecewise linear approximation (PLA) (Bonvin et al., 2017; Menke et al., 2016). Convex relaxation methods, such as those developed in Li et al. (2018) and Fooladivanda et al. (2018) reformulate nonconvex water network optimization problems as convex programs to improve tractability. In particular, the convex hull-based relaxation proposed by Li et al. (2018) addresses the joint scheduling of water pumps and electricity use in a micro water–energy nexus, enabling demand-side management while satisfying basic hydraulic requirements. The second-order cone (SOC) relaxation proposed in Fooladivanda et al. (2018) targets dynamic pump and storage operation in water distribution systems, reducing energy costs under time-varying tariffs while enforcing pressure and flow limits for practical operation. While these methods reduce computational complexity, they often introduce inaccuracies compared to solving the full nonconvex problem (Bianchi et al., 2023). Another alternative is PLA, which approximates nonlinear constraints by discretizing them into linear segments, maintaining higher solution accuracy. However, this approach introduces binary variables, leading to a Mixed-Integer Linear Programming (MILP) formulation, which remains computationally challenging when applied to large-scale problems (Gu and Sioshansi, 2025).

Applications of OWF in short-term scheduling, real-time (RT) control, and uncertainty modeling require accurate and computationally efficient models that can optimize system performance multiple times within constrained time frames (Garzón et al., 2022; Reis et al., 2023). To address these needs, recent research has increasingly focused on Machine Learning (ML) techniques for WDN optimization (Gambella et al., 2021; Bagloee et al., 2018). ML-based models serve as surrogate approximations of computationally challenging hydraulic models, enabling rapid yet accurate evaluations. Unlike traditional mathematical optimization models, ML models leverage large datasets to learn complex relationships between network input parameters, state variables, and outputs using methods like supervised, unsupervised, and reinforcement learning. These approaches have demonstrated significant improvements in computational efficiency (Ahmed et al., 2024).

Deep Learning (DL), a class of Artificial Neural Networks (ANNs) including Feedforward Neural Networks (FNNs), Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), Autoencoders, and Graph Neural Networks (GNNs) (Sit et al., 2020), has been applied to various WDN modeling and optimization tasks. These applications include short-term water demand forecasting (Nasser et al., 2020), nodal pressure estimation (Hajgató et al., 2021; Truong et al., 2024), leak detection (Javadiha et al., 2019), network renovation (Dini and Tabesh, 2019), network design (Sayers et al., 2019), and water quality monitoring (Li et al., 2019). While these methods excel at prediction and extracting complex patterns from large datasets, ANNs alone often fall short in solving complex mathematical optimization problems, especially in spatially and temporally coupled networked systems.

Alongside DL, Deep Reinforcement Learning (DRL) has emerged as a promising technology for optimization, with applications in water network management (Fu et al., 2022). For instance, Hajgató et al. (2020) developed a DRL model for pump control, where pump speed was determined based on system states, nodal pressures, and pump speed ratios. Similarly, Ma et al. (2024) employed a multi-agent DRL system for pump management in water networks, effectively maintaining nodal pressure constraints, highlighting DRL's potential in pump scheduling. Their results demonstrated that DRL could achieve performance comparable to traditional optimization algorithms. While valuable, these approaches fail to generalize across different network topologies and struggle to incorporate the hard physical constraints of water networks, such as flow and pressure limits, which are critical for accurate optimization of water distribution systems (Fu et al., 2022). Additionally, they do not provide any optimality guarantees for the underlying decision problem, and their performance is sensitive to the exploration-exploitation process during training, which can result in unstable or suboptimal decision policies in practice. As a result, relying solely on machine learning models to solve water network optimization problems remains challenging.

In recent years, a hybrid approach combining ANNs with traditional mathematical optimization is emerging (Wu and Wang, 2023). Here, mathematical optimization remains central, while ANNs learn nonlinear relationships between control and system state variables, effectively approximating nonconvex constraints. ANNs can be designed and trained to be convex, allowing their integration into convex optimization problems (Van Hentenryck, 2021). A notable example is the Input Convex Neural Network (ICNN), introduced by Amos et al. (2017). ICNNs are specialized ANN architectures that ensure convexity with respect to their input variables, making them ideal for integration with mathematical optimization frameworks in a process often called optimization-based inference. The advantage of this approach is that it preserves a physics-based optimization structure, in which most hydraulic and operational constraints are modeled explicitly and only selected nonlinear relationships are approximated by neural networks, enabling seamless integration into a convex optimization problem. This presents a research opportunity to develop a hybrid modeling framework that leverages ICNNs' expressive capabilities while maintaining accuracy, computational tractability, and solution robustness. Such a neural network-informed optimization framework could significantly enhance water network optimization efficiency, providing reliable solutions to end-users.

1.3. Scope and contributions

This work introduces a scalable and computationally efficient approach to solving the OWF problem in water distribution networks. We address the inherent nonlinearity and nonconvexity of constraints, particularly those related to friction losses and VSP hydraulics, by proposing a hybrid optimization framework. This framework leverages a novel variant of ICNNs, termed the Input Convex Concave neural Network (IC2N), to approximate these complex constraints. By training these networks to learn the relationship between control and system state variables and enforcing input–output convexity via structured weight constraints and loss regularization, we reformulate the inference problem as a linear program. Incorporating this linear program into the OWF problem yields a computationally efficient formulation that maintains solution accuracy. To the authors' knowledge, this is the first application of a neural network-informed reformulation to the OWF problem. Beyond the algorithmic aspects, these developments are tightly coupled to the hydraulics and operation of water distribution networks. The neural surrogates are trained on hydraulic data and embedded in the OWF formulation to preserve physically meaningful relationships between flow, head, and energy consumption, thereby advancing both optimization methodology and domain-specific modeling. This data-driven technique provides water utilities with a powerful tool for reducing energy costs and ensuring system reliability. The key contributions of this work are:

1. Introduces a hybrid approach that integrates ANNs with mathematical optimization to efficiently solve the OWF problem.
2. Proposes a novel deep learning architecture, termed IC2N, which accurately models nonlinear relationships between control and state variables related to nonconvex functions, using a combination of structured parameter constraints and a monotonicity-enforcing regularization scheme.
3. Ensures input–output convexity of the trained neural network, enabling its inference to be reformulated as a linear program and seamlessly embedded into the OWF problem.
4. Demonstrates significant improvements in both computational scalability and solution accuracy compared to state-of-the-art OWF approximation methods.

1.4. Organization

Section 2 introduces the nonlinear OWF formulation. Section 3 reviews neural network-informed approximations and the proposed IC2N framework. Section 4 presents numerical results, and Section 5 concludes with future research directions.

2. Methodology

2.1. Modeling assumptions

We consider a pressurized WDN represented as a directed graph with nodes $\mathcal{N}$ and arcs $\mathcal{A}$. Nodes include junctions $j \in \mathcal{J}$, reservoirs $r \in \mathcal{R}$, and tanks $s \in \mathcal{S}$, with $\mathcal{N} = \mathcal{J} \cup \mathcal{R} \cup \mathcal{S}$. Junctions connect pipes, pumps, or valves, enabling inflow/outflow to meet demand. Reservoirs are water sources without incoming pipes, thus serving as network starting points. Tanks provide storage with finite capacity, allowing water to be pumped in and later discharged using gravitational energy. Arcs (i, j) denote water transfer and are classified as pipes $(i, j) \in \mathcal{E}$, pumps $(i, j) \in \mathcal{P}$, and valves $(i, j) \in \mathcal{V}$, such that $\mathcal{A} = \mathcal{E} \cup \mathcal{P} \cup \mathcal{V}$. Pipes convey flow subject to frictional head loss. Pumps increase pressure head, and we focus on VSPs for their flexibility compared to FSPs. Pump status (on/off) is assumed predetermined, eliminating binary variables and enabling an NLP instead of a MINLP, which simplifies computation and benchmarking; this assumption can be relaxed without loss of generality. Valves reduce pressure in uncontrolled situations, e.g., steep elevation changes. The optimization horizon is discretized as $t \in \mathcal{T}$ with a one-hour time step, assuming steady-state hydraulic conditions at each interval. This temporal resolution is consistent with day-ahead pump scheduling, where decisions are typically driven by hourly day-ahead electricity price signals. Finer temporal resolutions can also be incorporated if required, without loss of generality.

2.2. Optimal water flow problem

The OWF problem is a mathematical optimization problem that plays an important role in WDN scheduling and control tasks. We first present its complete nonconvex formulation, followed by a description of the different decision variables, parameters and constraints involved. The full NLP formulation of the OWF problem is given as follows:

$$\underset{II}{Min} \quad \sum_{t\in\mathcal{T}} \sum_{(i,j)\in\mathcal{P}} P_t^{el} p_{i,j,t} \tag{1a}$$

$s.t.$

$$\sum_{i\in N_j^{To}} q_{i,j,t} - \sum_{k\in N_j^{Fr}} q_{j,k,t} = d_{j,t} \qquad j \in \mathcal{J}, t \in \mathcal{T} \tag{1b}$$

$$h_{s,t} - h_{s,t-1} = \frac{\tau}{A_s}\left(\sum_{i\in N_s^{To}} q_{i,s,t} - \sum_{j\in N_s^{Fr}} q_{s,j,t}\right) \qquad s \in \mathcal{S}, t \in \mathcal{T} \tag{1c}$$

$$h_{i,t} - h_{j,t} = f_{i,j,t}\frac{8L_{i,j}^E q_{i,j,t}^2}{\pi^2\, g D_{i,j}^5} \qquad (i, j) \in \mathcal{E}, t \in \mathcal{T} \tag{1d}$$

$$f_{i,j,t} = -1.8\log_{10}\left(\left(\frac{\varepsilon_{i,j}}{3.7\, D_{i,j}}\right)^{1.11} + \frac{6.9}{r_{i,j,t}}\right)^{-2} \qquad (i, j) \in \mathcal{E}, t \in \mathcal{T} \tag{1e}$$

$$r_{i,j,t} = \frac{4\rho q_{i,j,t}}{\pi\mu D_{i,j}^2} \qquad (i, j) \in \mathcal{E}, t \in \mathcal{T} \tag{1f}$$

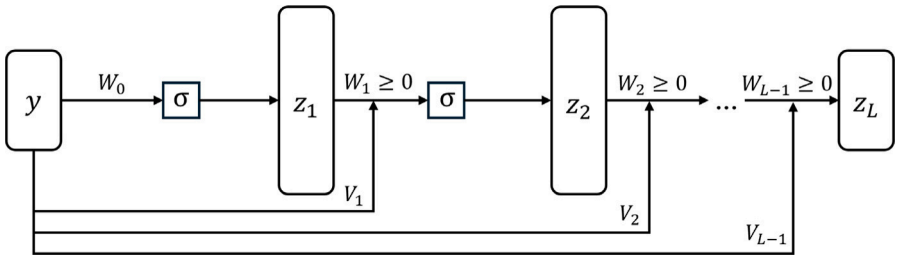

Fig. 1. Structure of an Input Convex Neural Network (ICNN).

$$\underline{\Omega}_{i,j} \leq \omega_{i,j,t} \leq \overline{\Omega}_{i,j} \qquad (i,j) \in \mathcal{P}, t \in \mathcal{T} \quad (1g)$$

$$h_{j,t} - h_{i,t} = B^{(1)}_{i,j} q^2_{i,j,t} + B^{(2)}_{i,j} q_{i,j,t} \omega_{i,j,t} + B^{(3)}_{i,j} \omega^2_{i,j,t} \qquad (i,j) \in \mathcal{P}, t \in \mathcal{T} \quad (1h)$$

$$p_{i,j,t} = C^{(1)}_{i,j} q^3_{i,j,t} + C^{(2)}_{i,j} q^2_{i,j,t} \omega_{i,j,t} + C^{(3)}_{i,j} q_{i,j,t} \omega^2_{i,j,t} + C^{(4)}_{i,j} \omega^3_{i,j,t} \qquad (i,j) \in \mathcal{P}, t \in \mathcal{T} \quad (1i)$$

$$h_{i,t} - h_{j,t} = k_{i,j,t} \qquad (i,j) \in \mathcal{V}, t \in \mathcal{T} \quad (1j)$$

$$\underline{H}_n + H^0_n \leq h_{n,t} \leq \overline{H}_n + H^0_n \qquad n \in \mathcal{N}, t \in \mathcal{T} \quad (1k)$$

$$\underline{Q}_{i,j} \leq q_{i,j,t} \leq \overline{Q}_{i,j} \qquad (i,j) \in \mathcal{A}, t \in \mathcal{T} \quad (1l)$$

The set of decision variables for the optimization problem is defined as follows: $\Pi := \{q_{i,j,t}, h_{n,t}, f_{i,j,t}, r_{i,j,t}, \omega_{i,j,t}, p_{i,j,t}, k_{i,j,t} \mid (i,j) \in \mathcal{E} \cup \mathcal{P} \cup \mathcal{V} \subseteq \mathcal{A}, n \in \mathcal{J} \cup \mathcal{R} \cup \mathcal{S} \subseteq \mathcal{N}, t \in \mathcal{T}\}$. The objective function (1a) minimizes the total energy-related operational cost of the WDN, considering electricity price P^{el}_t and pump power consumption $p_{i,j,t}$. Constraint (1b) enforces nodal flow balance at junction j, where incoming and outgoing flows $(q_{i,j,t}, q_{j,k,t})$ must equal demand $d_{j,t}$. Constraint (1c) governs storage tank dynamics, relating water level $h_{s,t}$ to inflow $q_{i,s,t}$, outflow $q_{s,j,t}$, time constant τ, and tank cross-sectional area A_s.

Constraint (1d) represents the Darcy–Weisbach equation, modeling head loss in pipeline segments based on pressure heads $(h_{i,t}, h_{j,t})$, flow rate $q_{i,j,t}$, pipe friction factor $f_{i,j,t}$, pipe length $L^E_{i,j}$, pipe diameter $D_{i,j}$, and gravitational acceleration g. Constraint (1e) uses the Haaland equation (Haaland, 1983) to determine $f_{i,j,t}$ as an explicit function of Reynolds number $r_{i,j,t}$, pipe diameter $D_{i,j}$, and absolute roughness $\epsilon_{i,j}$. Constraint (1f) defines $r_{i,j,t}$ based on fluid density ρ, dynamic viscosity μ, pipe diameter $D_{i,j}$, and flow rate $q_{i,j,t}$.

Constraints (1g)–(1i) describe pump hydraulic performance and energy consumption empirically. Constraint (1g) enforces operational feasibility for pumps, ensuring their relative speeds $\omega_{i,j,t}$ are within specified minimum $(\underline{\Omega}_{i,j})$ and maximum $(\overline{\Omega}_{i,j})$ limits. Eq. (1h) relates head gain $(h_{j,t} - h_{i,t})$ to flow rate $q_{i,j,t}$ and pump rotational speed $\omega_{i,j,t}$ using quadratic coefficients $B^{(1)}_{i,j}, B^{(2)}_{i,j}, B^{(3)}_{i,j}$. Eq. (1i) defines pump power consumption $p_{i,j,t}$ with a cubic relationship using pump-specific parameters $C^{(1)}_{i,j}, C^{(2)}_{i,j}, C^{(3)}_{i,j}, C^{(4)}_{i,j}$.

Constraint (1j) models PRV operation using a nonnegative pressure reduction variable $k_{i,j,t}$. Constraint (1k) imposes lower and upper bounds on nodal heads $h_{n,t}$, relative to elevation H^0_n. Finally, constraint (1l) ensures flow rates $q_{i,j,t}$ remain within predefined limits. Model (1a)–(1g) is computationally intractable due to the nonlinearity and nonconvexity of constraints (1d), (1e), (1h), and (1i). While nonlinear optimization solvers like IPOPT (Lougee-Heimer, 2003) can address this, they do not guarantee global optimality, and finding even local optima becomes challenging with increasing state-space dimensionality. To overcome these difficulties, we introduce a novel approach to approximate nonconvex constraints using ANNs. The ANNs are designed to encode key hydraulic constraints (i.e., pipe friction losses and pump performance curves) in a way that is consistent with established water distribution practice, while remaining tractable for large-scale optimization.

3. Neural network-based approximation

Traditional approximation methods, such as convex relaxations (Fooladivanda and Taylor, 2017; Li et al., 2018; Bianchi et al., 2024) and PLA (D'Ambrosio et al., 2010; Geißler et al., 2011), can lead to suboptimal solutions or higher computational costs. To address this, we propose an innovative approach leveraging ANNs. Specifically, we explore two ANN architectures: ICNN and a novel variant we term IC2N. ICNNs, introduced by Amos et al. (2017), are well-suited for approximating convex or quasi-convex constraints like Eqs. (1d)–(1e). IC2N extends this concept to handle a broader class of nonlinear functions by incorporating nonconvex components, such as those found in the pump hydraulic model (1h)–(1i).

3.1. Input Convex Neural Network

ICNNs (Fig. 1) are specialized ANN architectures designed to ensure convexity in the mapping from input to output. One can see that ICNNs consist of an input layer, denoted by y, several hidden layers z_l for $l \in [1, L-1]$, and an output layer z_L. A defining characteristic of ICNNs is the presence of two types of weight matrices: W_l for inter-layer connections and V_l for residual connections from the input layer y to each hidden layer and the final output layer. To guarantee that the output z_L remains a convex function of the input y, ICNNs must satisfy two key conditions. First, the weight matrices W_l for $l \in [1, L-1]$ must be non-negative, a constraint typically enforced through techniques such as weight clipping (Gulrajani et al., 2017) or projected gradient methods (Calamai and Moré, 1987). Second, all activation functions σ used in the network must be convex and non-decreasing. Common choices for activation functions include ReLU and Softplus (Bishop and Bishop, 2023).

The architecture of a fully connected ICNN with layers $l \in [0, L]$, where each hidden layer z_l $(l \in [1, L-1])$ receives input from the previous layer z_{l-1} and directly from the input via residual connections V_l, is given by (2a)–(2c). Note that $W_l \geq 0$ for $l \in [1, L-1]$, while V_l and b_l are unconstrained. Eq. (2a) maps the input y to the first hidden layer using a convex activation σ. Eq. (2b) preserves convexity by combining non-negative weighted activations from the previous layer and direct input contributions through V_l. Lastly, Eq. (2c) produces the output as a linear function of the last hidden layer and input, ensuring the overall function remains convex.

$$z_1 = \sigma(yW_0 + b_0) \qquad (2a)$$

$$z_{l+1} = \sigma(z_l W_l + y V_l + b_l) \qquad l \in [1, L-2] \quad (2b)$$

$$z_L = z_{L-1} W_{L-1} + y V_{L-1} + b_{L-1}. \quad (2c)$$

ICNNs were first introduced to accelerate inference tasks, aiming to find the input (e.g., y) that minimizes the network's scalar output, with applications in prediction, natural language processing, image recognition, and classification (Amos et al., 2017). However, their convex structure also enables embedding into larger optimization models to approximate nonlinear constraints. Note that here, inference refers to reproducing the forward propagation of a trained network: given fixed weights and biases, the network computes the output z_L for a given input y. Two optimization-based inference approaches exist: (1) MILP, modeling activations with binaries (Zhang et al., 2020), and (2) LP, approximating activations without binaries (Wu and Wang, 2023). While MILP is exact, it scales poorly with binary growth. To improve speed, we avoid MILP and use ReLU activations, $\sigma(z_l W_l + y V_l + b_l) = \max(z_l W_l + y V_l + b_l, 0)$. Since affine transformations and ReLU are convex, their composition is convex (Amos et al., 2017), allowing ICNN inference to be reformulated as an LP minimizing the output z_L, as shown in (3a)–(3d).

$$\underset{z_l}{\text{Min}} \quad z_L \quad (3a)$$

$$s.t.$$

$$z_1 \geq y W_0 + b_0 \quad (3b)$$

$$z_{l+1} \geq z_l W_l + y V_l + b_l \qquad l \in [1, L-1] \quad (3c)$$

$$z_l \geq 0 \qquad l \in [1, L-1] \quad (3d)$$

The objective function (3a) minimizes the output of the ICNN, denoted by z_L. Constraint (3b) represents the first-layer activation, ensuring that the first hidden layer z_1 is larger or equal the linear transformation of the input (yW_0) plus the bias term b_0. Constraints (3c) define activations for subsequent layers, requiring each z_{l+1} to be greater than or equal to the sum of the previous layer's output $z_l W_l$, the direct input connection yV_l, and the bias b_l. Here, z is the decision variable, while W, V, and b are fixed, trained parameters. The non-negativity constraint (3d) ensures all hidden activations remain non-negative, preserving the ICNN's convexity. These constraints together reformulate the trained ICNN inference as a linear program.

Note that the LP formulation is exact only when representing an ICNN trained for a single unit (e.g., a VSP). When the LP-based inference is embedded into a larger WDN optimization, this exactness is not guaranteed because the LP captures only the network propagation, while shared variables with the rest of the model can introduce discrepancies if the surrounding constraints or objective are not aligned with the ICNN's convex propagation rules. In particular, if external constraints on the ICNN output (z_L) become binding, the optimizer may adjust intermediate activations (z_l, $l \in [1, L-1]$) instead of the input y, so that the LP no longer reproduces the original nonlinear input–output mapping. We refer to the shared variables as the complicating variables and the external constraints on the ICNN output as the complicating constraint, following standard decomposition terminology (Conejo et al., 2006).

A complicating constraint is any constraint that links otherwise separable subproblems and, by doing so, can prevent a decomposition or relaxation from remaining tight. In the present context, complicating constraints are those that couple the ICNN output with other parts of the optimization in a way that can prevent the LP relaxation of the network inference from becoming tight. Here, LP-based inference represents the forward propagation of the trained network through inequality constraints on the hidden-layer activations; when the OWF objective and feasibility region do not impose additional complicating constraints on the ICNN output, these inequalities become binding at optimality and the LP reproduces the same mapping as a standard forward pass. If complicating constraints are added, the problem remains feasible, but there is no guarantee that the LP relaxation will stay tight, and an additional approximation error may arise beyond the intrinsic training error of the surrogate model. In our case, we later show that the operating domain of the OWF problem remains within the region where the ICNN approximation is exact, but this interaction should be examined carefully in other applications.

3.2. Input convex concave neural network

As noted, ICNNs are well-suited for learning convex or quasi-convex functions, such as the friction head loss model in (1d)–(1f). Here, an ICNN takes flow rate and pipe diameter as inputs (y) to predict friction head loss (z_L), serving as a surrogate for the Darcy–Weisbach equation and implicitly capturing friction factor variations via the Haaland equation. Note that, although we use the Haaland equation for training, the Colebrook–White equation could be used without loss of generality. Beyond pipelines, the hydraulic and energy equations of VSPs (1h) and (1i) are inherently nonconvex. Since standard ICNNs cannot model nonconvex functions, we introduce IC2N, a modified ICNN architecture for efficiently learning nonconvex constraints.

Inspired by Sankaranarayanan and Rengaswamy (2022), we relax the non-negativity of the final layer's weight matrix W_{L-1}, allowing IC2N to approximate nonconvex constraints via a difference-of-convex decomposition (Wen et al., 2018). For input $y \in \mathbb{R}^n$, IC2N computes $f(y) = g(y) - h(y)$, with g convex and h concave. This relaxation captures nonconvexities but introduces two challenges for LP reformulation. First, minimizing z_L becomes ill-posed since negative weights in W_{L-1} make the objective unbounded. Second, non-monotonic input–output relationships may yield suboptimal or invalid solutions in dynamic OWF optimization, where the objective depends on the product of time-varying variables and parameters. These issues must be addressed to ensure reliable LP reformulation. The first issue is tackled by leveraging the IC2N structure in the following inference problem:

$$\underset{z_l}{\text{Min}} \quad \|z_{L-1} W_{L-1}\|_1 \quad (4a)$$

$$s.t.$$

$$z_1 \geq y W_0 + b_0 \quad (4b)$$

$$z_{l+1} \geq z_l W_l + b_l \qquad l \in [1, L-2] \quad (4c)$$

$$z_L = z_{L-1} W_{L-1} + b_{L-1} \quad (4d)$$

$$z_l \geq 0 \qquad l \in [1, L-1] \quad (4e)$$

The objective (4a) minimizes the ℓ_1 norm of the final layer's linear output (excluding bias), i.e., $\|z_{L-1}W_{L-1}\|_1$, serving as a convex proxy in a difference-of-convex decomposition. Although convex, the ℓ_1 norm is applied to a linear combination with mixed-sign weights, yielding a piecewise structure that can switch between convex and concave parts, and thus a nonconvex overall mapping. Eqs. (4b)–(4d) define the activations up to z_L, while (4e) enforces non-negativity to layer $L-1$. Unlike ICNNs, IC2N omits residual terms yV_l: although such connections may improve stability, prior work shows they are unnecessary and that removing them simplifies the architecture while enhancing nonconvex expressiveness (Sankaranarayanan and Rengaswamy, 2022). Note that we specifically adopt the ℓ_1 norm because it yields a convex, piecewise-linear objective that treats positive and negative contributions of $z_{L-1}W_{L-1}$ symmetrically. Other convex norms (e.g., ℓ_2, ℓ_p, ℓ_∞, or the Frobenius norm) produce smooth, nonlinear, or element-selective results, which do not preserve the desired difference-of-convex structure in the objective function. The proposition and proof that follow rely on this property.

Next, we tackle the second issue concerning monotonicity. While IC2N inference is exact when used independently, its non-monotonic behavior complicates embedding within broader optimization problems such as OWF. We address this by introducing a monotonicity regularization term during training, which ensures exactness of IC2N LP-based inference when embedded into the broader optimization. Here, exactness means that the LP in (4a)–(4e) produces the same z_L as a standard feedforward evaluation of the trained network. Embedding within a broader optimization refers to integrating this LP into the OWF model while still reproducing the original IC2N output despite additional surrounding constraints (i.e., hydraulic constraints not represented in the IC2N inference) and a specific objective, such as minimizing pumping costs under time-varying electricity prices. Under the conditions formalized in Proposition 1, the LP inference remains exact and can reliably approximate pump hydraulics and energy consumption constraints within the OWF problem.

Proposition 1. *The LP inference defined by (4a)–(4e) remains exact and can reliably approximate pump hydraulics and energy consumption constraints within the OWF problem under the following conditions:*

1. *Activations $\sigma : \mathbb{R} \to \mathbb{R}_{\geq 0}$ are convex and non-decreasing,*
2. *Hidden layer weights satisfy $W_l \geq 0$ for $l \in [1, L-2]$,*
3. *No complicating constraints are imposed on z_L,*
4. *The objective $\|z_{L-1}W_{L-1}\|_1$ is monotonic in z_L.*

Under these conditions, the LP inference defined by (4a)–(4e) yields the same output (z_L) as the standard forward evaluation of the trained IC2N for any feasible input, so that the LP inference is exact when embedded in the OWF problem. Detailed explanations and the complete proof are provided in Section S1 of the supplementary materials.

In practice, the ReLU activation function satisfies condition 1 of the proposition, as it is both convex and non-decreasing. Condition 2 can be enforced during training through weight clipping or projected gradient methods to ensure non-negative weights in the hidden layers. Condition 3 is satisfied by avoiding additional constraints on z_L beyond those implied by the LP inference itself, so that the surrounding OWF objective and feasibility region do not interfere with the tightness of the LP relaxation. Satisfying condition 4 (monotonicity between z_L and $\|z_{L-1}W_{L-1}\|_1$) requires explicit regularization. To this end, we introduce a monotonicity regularization term, denoted by $\mathcal{L}^M$, which penalizes negative covariance between z_L and $\|z_{L-1}W_{L-1}\|_1$ across a batch of N training samples. The term is defined as:

$$\mathcal{L}^M = \frac{1}{N}\sum_{i=1}^{N}\left(\|z_{L-1}^{(i)}W_{L-1}\|_1 - \bar{z}_{L-1}\right)\left(z_L^{(i)} - \bar{z}_L\right) \tag{5}$$

where $\bar{z}_{L-1} = \frac{1}{N}\sum_{j=1}^{N}\|z_{L-1}^{(j)}W_{L-1}\|_1$ and $\bar{z}_L = \frac{1}{N}\sum_{j=1}^{N} z_L^{(j)}$ are the batch means.

Eq. (5) computes the empirical covariance between $\|z_{L-1}W_{L-1}\|_1$ (i.e., the norm of the penultimate layer linear transformation excluding bias) and z_L (i.e., the output layer), encouraging a positive correlation and thereby promoting the desired monotonicity. The total training loss function is then defined in Eq. (6) as follows:

$$\mathcal{L}^{\mathrm{TOT}} = \frac{1}{N}\sum_{i=1}^{N}(z_L^{(i)} - x^{(i)})^2 + \zeta\mathcal{L}^M \tag{6}$$

The first term in Eq. (6) is the Mean Squared Error (MSE) between predicted outputs $z_L^{(i)}$ and ground truth targets $x^{(i)}$. The hyperparameter ζ controls the strength of the monotonicity regularization and should be tuned via cross-validation to balance predictive accuracy and monotonic behavior. With this regularization, IC2N can approximate nonconvex relationships, such as the energy consumption of VSPs, using flow rate and pressure head as inputs and predicting power consumption. In this role, the trained IC2N acts as a physics-informed surrogate model, implicitly capturing hydraulic constraints within the prediction task. Note that Eq. (6) does not require modification across applications: as long as the training data originate from a stationary operating profile that can be represented via a difference-of-convex decomposition, the methodology and proof remain valid. The only application-dependent adjustment is the choice of the regularization weight ζ, which promotes the desired monotonicity required by Condition 4.

3.3. Neural network-informed OWF

The final OWF formulation integrates ICNN-based approximations for friction head losses and an IC2N-based model for pump hydraulics. Before presenting the mathematical formulation, we introduce the variables used in the ICNN and IC2N inference procedures for the OWF problem. The neural network inputs are collected in the vectors $y_{i,j,t}$, which contain the relevant physical hydraulic variables for each edge (i,j) at time step t: flow and diameter for the ICNN, and flow and head gain for the IC2N. Latent activations of each hidden layer are represented by the vectors $z_{i,j,t,l}$, with the final-layer activation $z_{i,j,t,L}$ corresponding to the network output: friction loss for the ICNN and pump energy consumption for the IC2N. The forward propagation from input to output is implemented as linear constraints in (8a)–(9f), thereby replacing the original nonlinear hydraulic constraints. From this point onward, the symbol $\breve{\cdot}$ denotes decision variables and parameters specifically associated with the ICNN-based approximation of friction head losses, whereas variables associated with the IC2N retain the standard notation.

The objective minimizes total electricity costs for water pumping, with the IC2N providing estimates of pump energy consumption. As defined in Constraints (4a)–(4e), the objective of the IC2N serves as a convex proxy for true pump power consumption, formulated as a function of the final layer's linear transformation and scaled by time-varying electricity prices. It is formulated as follows:

$$\text{Min} \sum_{t\in\mathcal{T}} P_t^{el} \left(\sum_{(i,j)\in\mathcal{P}} \sum_{\gamma\in\Gamma^{L-1}} \sum_{d\in\Gamma^L} |W_{L-1,\gamma,d}| z_{i,j,t,L-1,\gamma} \right) \tag{7a}$$

Eq. (7a) defines the IC2N-assisted OWF objective. For each time step t, the electricity price P_t^{el} multiplies the convex proxy of pump power, computed from the penultimate layer activations $z_{i,j,t,L-1,\gamma}$ and the trained weights $W_{L-1,\gamma,d}$. The use of absolute weights ensures convexity, and since $z_{i,j,t,L-1,\gamma} \geq 0$, we have $|W_{L-1,\gamma,d}| z_{i,j,t,L-1,\gamma} = |W_{L-1,\gamma,d} z_{i,j,t,L-1,\gamma}|$. Bias terms are excluded, as they are fixed and do not affect optimization. This formulation ensures tight IC2N inference constraints at optimality, thereby enabling the approximation of true nonlinear pump power consumption. Although the LP objective does not directly yield energy cost (unlike the NLP formulation), the actual pump power consumption can be recovered post-optimization via the IC2N output $z_{i,j,t,L,d}$.

Constraints (8a)–(8f) define the ICNN inference procedure used to approximate friction head losses. These constraints replace constraints (1d)–(1f) from the original NLP formulation, thereby providing a linear ICNN-based approximation.

$$\acute{y}_{i,j,t,0} = q_{i,j,t} \qquad (i,j)\in\mathcal{E}, t\in\mathcal{T} \tag{8a}$$

$$\acute{y}_{i,j,t,1} = D_{i,j} \qquad (i,j)\in\mathcal{E}, t\in\mathcal{T} \tag{8b}$$

$$\acute{z}_{i,j,t,1,d} \geq \sum_{\gamma\in\Gamma^y} \acute{W}_{0,\gamma,d} \acute{y}_{i,j,t,\gamma} + \acute{b}_{0,d} \qquad (i,j)\in\mathcal{E}, t\in\mathcal{T}, d\in\Gamma^1 \tag{8c}$$

$$\acute{z}_{i,j,t,l+1,d} \geq \sum_{\gamma\in\Gamma^l} \acute{W}_{l,\gamma,d} \acute{z}_{i,j,t,l,\gamma} + \sum_{\gamma\in\Gamma^y} \acute{V}_{l,\gamma,d} \acute{y}_{i,j,t,\gamma} + \acute{b}_{l,d} \qquad (i,j)\in\mathcal{E}, t\in\mathcal{T}, l\in[1,L-1], d\in\Gamma^{l+1} \tag{8d}$$

$$\acute{z}_{i,j,t,l,d} \geq 0 \qquad (i,j)\in\mathcal{E}, t\in\mathcal{T}, l\in[1,L-1], d\in\Gamma^1 \tag{8e}$$

$$\acute{z}_{i,j,t,L,0}\ L_{i,j}^E = h_{i,t} - h_{j,t} \qquad (i,j)\in\mathcal{E}, t\in\mathcal{T} \tag{8f}$$

Constraint (8a) sets the first input neuron to the flow rate $q_{i,j,t}$, while constraint (8b) assigns the pipe diameter $D_{i,j}$ to the second input neuron, embedding the key hydraulic variables. Constraints (8c) and (8d) define the affine transformations in the first and deeper hidden layers, respectively, where neuron outputs depend on weighted sums of inputs and previous layer outputs plus biases. Constraint (8e) enforces nonnegativity of all neuron outputs, ensuring the ICNN's convexity. Finally, constraint (8f) links the ICNN's output to the nodal head difference $h_{i,t} - h_{j,t}$. The intermediate variables $\acute{z}_{i,j,t,l,d}$ form a latent-space representation of the frictional forces in the pipe, which depend on flow and diameter as well as other hydraulic parameters (e.g., roughness) internally captured by the network during training, while the final-layer activation $\acute{z}_{i,j,t,L,0}$ directly approximates the resulting friction head loss used in the OWF constraints. Unlike the general ICNN inference problem (3a)–(3d), minimizing the final ICNN output explicitly in the objective is unnecessary here. Since friction losses determine pump head gain, which the IC2N model minimizes indirectly by reducing pump energy consumption, the ICNN output $\acute{z}_L$ is implicitly minimized as part of the overall optimization.

Constraints (9a)–(9f) define the IC2N inference procedure used to approximate pump hydraulics and energy consumption. These constraints replace the nonlinear constraints (1h)–(1i) from the original NLP formulation, thereby providing a linear IC2N-based approximation of the pump model.

$$y_{i,j,t,0} = q_{i,j,t} \qquad (i,j)\in\mathcal{P}, t\in\mathcal{T} \tag{9a}$$

$$y_{i,j,t,1} = h_{j,t} - h_{i,t} \qquad (i,j)\in\mathcal{P}, t\in\mathcal{T} \tag{9b}$$

$$z_{i,j,t,1,d} \geq \sum_{\gamma\in\Gamma^y} W_{0,\gamma,d} y_{i,j,t,\gamma} + b_{0,d} \qquad (i,j)\in\mathcal{P}, t\in\mathcal{T}, d\in\Gamma^1 \tag{9c}$$

$$z_{i,j,t,l+1,d} \geq \sum_{\gamma\in\Gamma^l} W_{l,\gamma,d} z_{i,j,t,l,\gamma} + b_{l,d} \qquad (i,j)\in\mathcal{P}, t\in\mathcal{T}, l\in[1,L-2], d\in\Gamma^{l+1} \tag{9d}$$

$$z_{i,j,t,L,d} = \sum_{\gamma\in\Gamma^{L-1}} W_{L-1,\gamma,d} z_{i,j,t,L-1,\gamma} + b_{L-1,d} \qquad (i,j)\in\mathcal{P}, t\in\mathcal{T}, d\in\Gamma^L \tag{9e}$$

$$z_{i,j,t,l,d} \geq 0 \qquad (i,j)\in\mathcal{P}, t\in\mathcal{T}, l\in[1,L-1], d\in\Gamma^1 \tag{9f}$$

Constraint (9a) assigns the first input neuron $y_{i,j,t,0}$ to the pump flow $q_{i,j,t}$, while (9b) sets the second input neuron $y_{i,j,t,1}$ to the hydraulic head gain $h_{j,t} - h_{i,t}$. Constraint (9c) defines the first hidden layer's activation via an affine transformation of inputs, and constraint (9d) propagates this linear structure through the intermediate layers. Constraint (9e) computes the final output as a linear function of the last hidden layer, while constraint (9f) enforces nonnegativity on hidden neuron outputs, maintaining convexity. The intermediate variables $z_{i,j,t,l,d}$ capture the internal nonlinear behavior of the pump hydraulics as a latent-space representation, whereas the final-layer activation $z_{i,j,t,L,0}$ provides the IC2N-based approximation of pump energy consumption, effectively replacing the original nonlinear pump power curve and linking directly to the physical quantity $p_{i,j,t}$ in the OWF problem.

The complete formulation of the proposed neural network-assisted OWF problem is summarized in (10). The objective function and constraints together define a linear program that efficiently approximates the original OWF problem and can be solved using standard off-the-shelf optimization solvers.

$$\underset{\Xi}{\text{Min}} \quad (7a), \quad s.t. \quad \begin{cases} (8a)-(8f), \\ (9a)-(9f), \\ (1b), (1c), (1j)-(1l). \end{cases} \tag{10}$$

The set of decision variables for the optimization problem is defined as follows: $\Xi := \{q_{i,j,t},\ h_{n,t},\ p_{i,j,t},\ k_{i,j,t},\ \acute{z}_{i,j,t,l,d},\ z_{i,j,t,l,d},\ \acute{y}_{i,j,t,\xi},\ y_{i,j,t,\xi},\ |\ (i,j)\in\mathcal{E}\cup\mathcal{P}\cup\mathcal{V}\subseteq\mathcal{A},\ n\in\mathcal{J}\cup\mathcal{R}\cup\mathcal{S}\subseteq\mathcal{N}, t\in\mathcal{T}, l\in\mathcal{L},\ d\in\Gamma^l, \xi\in\Gamma^y\}$. Constraints (8a)–(8f) describe the ICNN approximation of friction head losses.

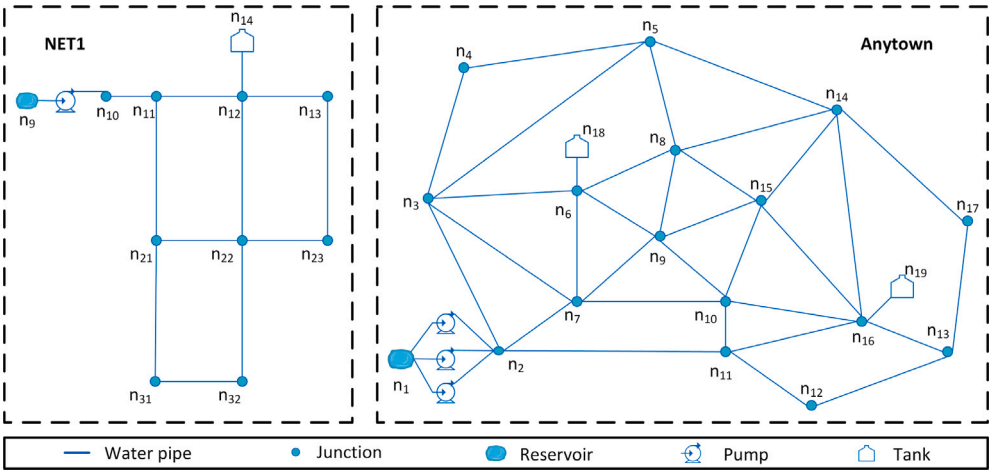

Fig. 2. EPANET water distribution network test cases used in this study: NET1 (left), and Anytown (right).

Constraints (9a)–(9f) represent the IC2N approximation of pump hydraulics and power consumption. Finally, constraints (1b), (1c), and (1j)–(1l) include all original linear network constraints from the NLP model, such as network flow conservation, tank dynamics, and physical bounds on system variables, which remain unchanged.

4. Results and discussion

4.1. Case study

We demonstrate our approach on two adapted EPANET test cases: NET1 and Anytown, whose layouts are shown in Fig. 2. NET1 (Rossman, 2000) is a simplified WDN with 11 nodes (9 junctions, 1 reservoir, 1 elevated tank), 12 pipes (19.3 km, 250–450 mm), and one pump. Demand varies from 40–95 l/s across six nodes, supplied by the reservoir and tank (259 m elevation, 15.4 m diameter, 30.5–45.7 m operating range). To assess scalability, we also use Anytown (Walski et al., 1987), a benchmark urban WDN with 19 nodes (16 junctions, 1 reservoir, 2 tanks), 34 pipes (69.6 km, 200–400 mm), and three parallel pumps. Demand ranges 200–550 l/s across 16 nodes. Tanks (65.5 m elevation, 16 and 17 m diameter) operate between 3–10.7 m.

The ICNN predicts pipe friction losses from synthetic data generated with the Darcy–Weisbach equation (1d), with the friction factor computed via the Haaland equation (1e). We sample 150,000 scenarios of flow and diameter (pipe length fixed at 1 m), yielding head loss per unit length (later scaled in optimization, Eq. (8f)). The ICNN maps flow and diameter to head loss with three hidden layers (10, 5, 2 neurons), enforcing convexity by clipping $W_l \geq 0$ for $l \in [1, L-1]$. It is trained for 20,000 epochs and tested on unseen data.

The IC2N predicts VSP power using data from the empirical model (1h)–(1i), incorporating efficiency via the reference h–q curve and speed adjustment. We sample 250,000 scenarios of flow and head gain (min relative speed 0.6). The IC2N maps inputs to power with two hidden layers (24, 48 neurons), enforcing $W_l \geq 0$ for $l \in [1, L-2]$. It is trained for 25,000 epochs with a monotonicity regularization term ($\zeta = 10^{-4}$, Eqs. (5)–(6)). For both models, inputs and outputs are MinMax-scaled, and data is split into training (70%), validation (20%), and test (10%) sets. ReLU activations and the Adam optimizer are used with a mean squared error loss. The datasets used in this study, including all network input data and the training datasets for the ICNN and IC2N models, are available in the online repository (Bianchi, 2025).

The proposed neural network-informed OWF problem is solved using Gurobi and compared against the exact NLP formulation solved with IPOPT. We also test two state-of-the-art approximations: piecewise linear approximation (MILP) and convex relaxation (CVX), both solved with Gurobi. The MILP replaces nonlinearities with piecewise linear segments using an SOS2-based PLA (Beale and Tomlin, 1970) for the univariate constraint (1e) and a triangular PLA (Geißler et al., 2011) for the bivariate constraints (1h)–(1i). The CVX method applies the quasi-convex hull relaxation (Li et al., 2018; Bianchi et al., 2023) to reformulate nonconvexities into convex constraints. These comparisons assess IC2N computational efficiency, solution accuracy, and trade-offs with alternative methods. We refer the reader to Sections S2 and S3 of the supplementary materials for a detailed description of the MILP and CVX model formulations. Simulations are implemented in Python v3.12.3 with Pyomo v6.8.2 on a laptop with 32 GB RAM and an Intel(R) Xeon(R) 6-core 2.80 GHz processor.

4.2. Numerical results

Table 1 compares the results of the proposed neural network-informed OWF formulation (IC2N) with those of the nonlinear formulation (NLP and NLP-INI), the mixed-integer linear formulation (MILP), and the convex relaxation (CVX), applied to the NET1 and Anytown test cases. NET1 represents a small-scale network, while Anytown serves as a larger-scale benchmark, helping to assess the scalability of the different approaches. Note that the NLP-INI model initializes the NLP model with the solution of the CVX problem to provide a warm start, thereby enhancing solver convergence and stability. We refer the reader to Section S4 of the supplementary materials for a detailed description of the initialization procedure. The evaluation criteria include the objective function value, computational time, and optimality gap, which indicates how closely each solution approaches the reference provided by NLP-INI. Note that, in the IC2N model, the optimization objective acts as a convex surrogate. Therefore, the true objective value (i.e., the total electricity cost) reported here is calculated a posteriori from the network output and the electricity price profile.

Across the two networks, the results highlight clear differences in how each formulation balances computational effort and solution quality. For NET1, where all methods remain close to the NLP-INI reference, the primary distinction lies in efficiency. The NLP formulation attains the

Table 1
Comparison of various OWF problem formulations based on objective function value, solution time, and optimality gap relative to the NLP solution for NET1 (small) and Anytown (large) networks.

Network	Formulation	Objective (€)	Time (s)	Optimality gap[a] (%)
NET1	NLP	257.41	63.26	0.00
	MILP	257.61	192.48	0.08
	CVX	254.77	0.04	1.03
	NLP-INI[c]	257.41	1.20	0.00
	IC2N	258.26	0.06	0.33
Anytown	NLP	N/A[b]	N/A[b]	N/A[b]
	MILP	1898.81	511.57	0.67
	CVX	1848.49	0.14	3.36
	NLP-INI[c]	1911.59	3.84	0.00
	IC2N	1906.18	0.17	0.28

[a] NLP-INI solution serves as a reference for optimality gap calculation.
[b] Model converged to a locally infeasible solution.
[c] NLP model initialized with solution of CVX problem.

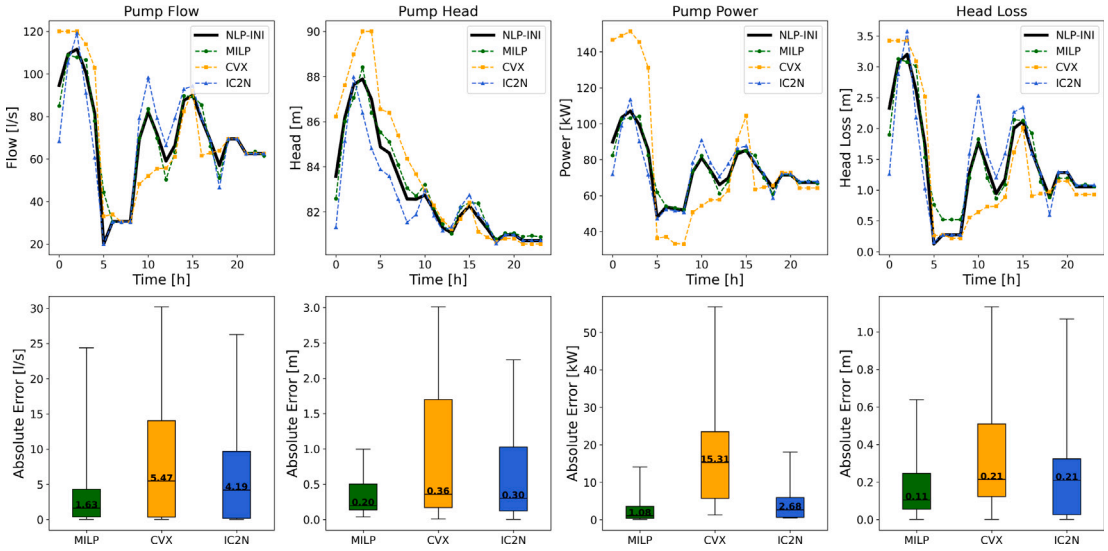

Fig. 3. NET1 results comparison across NLP-INI, MILP, CVX, and IC2N formulations showing time series (top) and absolute errors (bottom) for pump flow, pump head gain, pump power, and friction head loss in pipe (10,11). Absolute error is computed using the NLP-INI solution as the reference. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

reference solution (257.41 €) but requires substantially longer runtimes, whereas NLP-INI achieves the same solution with an order-of-magnitude speed-up, confirming the effectiveness of warm-starting. MILP maintains high accuracy but exhibits considerably higher computational cost, even at this small scale. In contrast, CVX and IC2N demonstrate significantly lower runtimes. CVX does so at the expense of a larger optimality gap, while IC2N retains a tight approximation of the reference solution, indicating that the neural network–informed linearization successfully preserves problem structure while remaining computationally lightweight.

The differences become more pronounced in the Anytown benchmark. The NLP formulation fails to converge, illustrating the difficulty of applying nonconvex continuous optimization methods to larger hydraulic networks. MILP continues to produce accurate solutions but with a substantial increase in solution time, suggesting scalability limitations. CVX remains extremely fast but exhibits the largest deviation from the reference, mirroring its looser relaxation. IC2N, however, attains a near-reference solution while keeping computational time comparable to CVX, demonstrating strong scalability and robustness. Overall, the results indicate that the IC2N formulation offers an effective compromise between accuracy and computational efficiency, particularly in contexts where classical NLP approaches become unreliable or impractical.

In addition to computational performance, we also assess whether the proposed IC2N formulation preserves the expected physical behavior. In the following, we highlight how the IC2N model respects realistic hydraulic relationships and translates into operationally meaningful scheduling decisions. To gain further insight into the model's hydraulic behavior, Fig. 3 compares the performance of the IC2N model with the NLP-INI reference, as well as the MILP and CVX approximations, on the NET1 (small) network. The analysis focuses on four key decision variables: pump flow, pump head gain, pump power, and friction head loss in a selected pipe. The top row of the figure presents time series plots of the NLP solution alongside the approximations over the 24-hour simulation horizon. The bottom row displays the absolute error distributions of the approximations relative to the NLP solution, where the horizontal black line indicates the median, the box represents the interquartile range (IQR), and the whiskers span the full range of errors.

Visual inspection of the time-series plots in the upper panel shows that all three approximation models generally follow the NLP trend (thick black line), although their accuracy varies. The MILP formulation (green dashed line) consistently demonstrates the closest match to the NLP reference. IC2N (blue dashed line) also provides a good approximation, while CVX (yellow dashed line) exhibits the most pronounced deviations from the NLP solution. The lower panel, comprising box plots of absolute errors, quantitatively substantiates these observations.

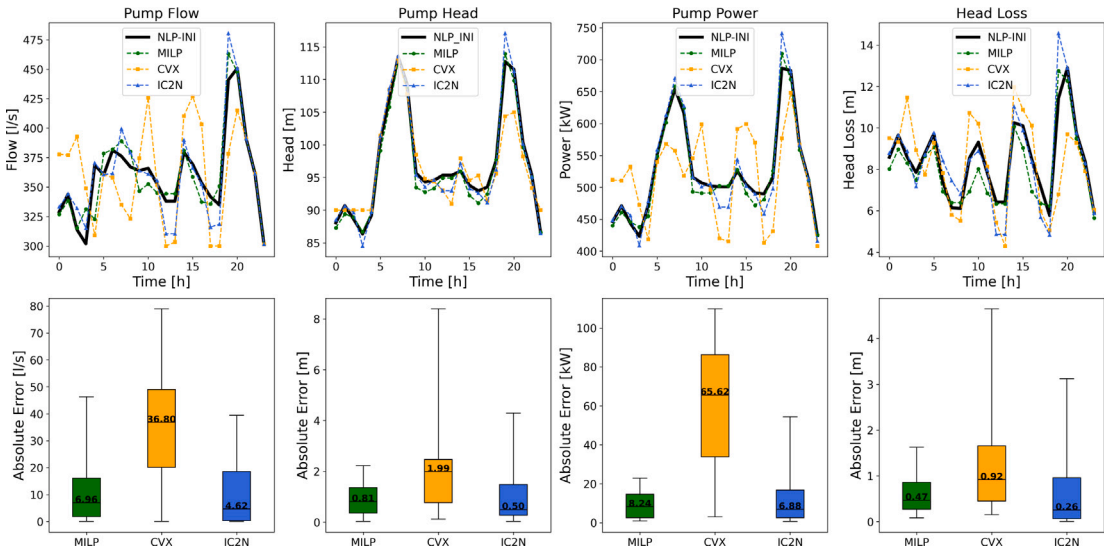

Fig. 4. Anytown results comparison across NLP-INI, MILP, CVX, and IC2N formulations showing time series (top) and absolute errors (bottom) for pump flow, pump head gain, pump power, and friction head loss in pipe (2,3). Absolute error is computed using the NLP-INI solution as the reference.

MILP achieves the highest accuracy, with the smallest median errors and tightest IQRs across all variables. For pump flow, MILP exhibits a median absolute error of approximately 1.63 l/s, with an IQR from 0.34 l/s to 4.28 l/s and a maximum error of 24.38 l/s. For pump head, the median error is about 0.2 m, with an IQR from 0.14 m to 0.5 m and a maximum of 1 m. In the case of pump power, the median error is approximately 1.08 kW, with an IQR from 0.41 kW to 3.54 kW and a maximum of 14.1 kW. For head loss, the median error is about 0.11 m, with an IQR from 0.06 m to 0.25 m and a maximum of 0.64 m.

IC2N delivers moderately higher errors than MILP but remains substantially more accurate than CVX. Its median errors are roughly 2.6 times larger for pump flow (4.19 l/s vs. 1.63 l/s), 1.5 times larger for pump head (0.3 m vs. 0.2 m), 2.5 times larger for pump power (2.68 kW vs. 1.08 kW), and 1.9 times larger for head loss (0.21 m vs. 0.11 m) compared to MILP. Compared to CVX, IC2N reduces median errors by 23% for pump flow, 17% for pump head, 82% for pump power, and 0% for head loss. Moreover, its IQRs are consistently narrower, indicating better robustness and more consistent performance.

CVX exhibits the largest median errors and widest IQRs, reflecting substantial inaccuracies and high variability. For pump flow, the median error is 5.47 l/s (IQR 0.32–14.05 l/s, max 30.2 l/s), over 3 times larger than MILP and 30% larger than IC2N. Pump head shows a median of 0.36 m (IQR 0.17–1.70 m, max 3.01 m), roughly 1.8 times larger than IC2N and 1.8 times larger than MILP. Pump power displays the most pronounced discrepancies, with a median error of 15.31 kW (IQR 5.7–23.5 kW, max >56 kW), more than 14 times larger than MILP and over 5 times larger than IC2N. For head loss, the median is 0.21 m (IQR 0.12–0.51 m, max 1.13 m), about double MILP and equal to IC2N.

Fig. 4 presents the same comparison for the larger and more complex Anytown network. This comparison evaluates the performance of each approach for a more challenging system with greater spatial and hydraulic complexity than the smaller NET1 case, thus providing valuable insights into the scalability and robustness of the proposed approach.

The time-series plots for Anytown show that MILP continues to provide the closest approximation to the NLP trend, while IC2N maintains a relatively good fit and CVX exhibits more substantial deviations and less consistent tracking of the reference. The lower panel, which presents box plots of absolute errors, quantitatively reinforces these findings. For pump flow, MILP achieves a median absolute error of approximately 6.96 l/s, with an IQR from 1.84 l/s to 16.14 l/s and a maximum error of 46.29 l/s. IC2N performs slightly better in terms of central tendency, with a median error of 4.62 l/s, an IQR from 0.32 l/s to 18.53 l/s, and a maximum error of 39.48 l/s. This corresponds to an improvement of about 34% in the median error compared to MILP. However, MILP still shows tighter variability, indicating greater consistency.

For pump head, MILP has a median absolute error of 0.81 m, whereas IC2N achieves a lower median error of 0.5 m, indicating a 38% reduction. However, the IQR for IC2N (0.27 m to 1.48 m) is wider than that of MILP (0.36 m to 1.36 m), indicating that IC2N may occasionally produce larger errors across time steps. In the case of pump power, IC2N achieves a median absolute error of 6.88 kW, while MILP's median error is 8.24 kW. This represents a reduction of approximately 16.5% in median error. However, the upper tail for IC2N is heavier, with a maximum error of 54.38 kW compared to MILP's 22.89 kW.

For friction head loss, IC2N also shows lower central errors, with a median of 0.26 m compared to MILP's 0.47 m, equivalent to a 45% reduction. Nevertheless, IC2N has a broader IQR (0.07 m to 0.96 m) and maximum error of 3.12 m, whereas MILP has a tighter IQR (0.27 m to 0.86 m) and a maximum of 1.63 m. This again highlights the trade-off between average-case accuracy and worst-case deviations. Compared to CVX, IC2N clearly offers superior performance across all variables. For instance, CVX's median absolute error for pump flow is 36.86 l/s, nearly eight times higher than IC2N's. In pump power, CVX shows a high median error of 65.62 kW, nearly 10 times larger than IC2N's 6.88 kW. Similar gaps are observed in pump head (1.99 m for CVX vs. 0.5 m for IC2N) and head loss (0.92 m vs. 0.26 m).

4.3. Discussion

The comparative evaluation underscores the strengths of the IC2N approach, particularly in terms of computational efficiency and solution quality. On the NET1 network, IC2N achieves a 0.33% optimality gap relative to the NLP baseline, slightly less accurate than MILP's 0.08%, but solves in only 0.06 s, over 3200 times faster than MILP (192.48 s) and more than 1000 times faster than NLP (63.26 s). The NLP-INI variant reduces

time to 1.20 s but remains about 20 times slower than IC2N. For the more complex Anytown network, NLP fails to find a feasible solution, while IC2N attains a 0.28% gap, outperforming MILP (0.67%) and CVX (3.36%) in accuracy, solving in 0.17 s, approximately 3000 times faster than MILP and over 20 times faster than NLP-INI. These results suggest that IC2N offers a favorable trade-off for large-scale systems, balancing accuracy and speed. Initializing NLP with the solution of CVX improves convergence, but adds modeling overhead and may not scale well. In larger networks with multiple technologies and nonlinear profiles, the quality of the CVX starting point can degrade, limiting the reliability of NLP-INI.

Time-series and error analyses show MILP yields the lowest median absolute errors on NET1. IC2N's errors are roughly 1.6 to 2.5 times higher but still substantially better than CVX, reducing median pump power error by over 80% and pump flow error by 23% relative to CVX. On Anytown, IC2N outperforms MILP by lowering pump flow and head errors by about 34% and 38%, and pump power and head loss errors by 16.5% and 45%, respectively. However, IC2N exhibits higher variability with broader interquartile ranges and larger maximum errors, while MILP maintains tighter error distributions. CVX shows the poorest performance across all variables, with median errors up to ten times higher than IC2N on Anytown.

Overall, IC2N balances speed and accuracy effectively. While MILP is marginally more consistent on small networks, IC2N achieves near-MILP accuracy with orders-of-magnitude faster solve times, making it well suited for a wide range of applications. It is particularly promising for Monte Carlo-based scenario optimization, where the problem must be solved repeatedly, or for network expansion problems that already involve substantial integer complexity. In such cases, introducing additional integer variables through MILP or MINLP formulations would likely be computationally prohibitive. It is worth noting that the ICNN and IC2N models used in this study are relatively small. In practice, neural networks are designed to be only as complex as necessary to capture system dynamics while avoiding overfitting. The chosen model sizes were sufficient for the hydraulic complexity of the WDNs considered here. For more complex networks or operating conditions, larger or deeper architectures may be required.

Although the results demonstrate good predictive performance, potential underfitting, overfitting, and limitations in generalizability are inherent challenges of surrogate modeling. Such issues were observed during training and were mitigated through careful data selection and hyperparameter tuning. A detailed quantitative analysis is beyond the scope of the present work, but we note these aspects to emphasize that data quality and model design remain essential when applying ICNN/IC2N surrogates in other contexts. Further, neural networks' sensitivity to noisy inputs and lack of interpretability pose additional challenges, and may require retraining, integration of domain knowledge, and exploring other hybrid data-driven and physics-based approaches to improve reliability.

5. Conclusion

This work developed a hybrid optimization framework integrating the proposed IC2N neural network architecture into water distribution network optimization, enabling accurate approximation of nonlinear hydraulic relationships within a convex optimization setting. The proposed method achieves near-MILP solution quality with computational speeds several orders of magnitude faster, demonstrating scalability and robustness across network sizes. On benchmark networks, it solved the OWF problem over 3000 times faster than traditional MILP, while maintaining an optimality gap below 0.33% and reducing errors in critical operational variables by up to 80% compared to state-of-the-art convex relaxations. These findings suggest IC2N as a practical alternative for large-scale and complex water system optimization problems where traditional methods are computationally prohibitive. The approach offers stakeholders a practical, data-driven tool for supporting decision-making in water distribution planning and operations. While the current study focuses on relatively small benchmark networks to enable direct comparisons with nonlinear and mixed-integer optimization methods, future work will extend the framework to established large-scale benchmark networks, such as the one presented in Marchi et al. (2014), as well as real-world cases. This will allow a more thorough assessment of the method's practical applicability and scalability in realistic operational settings. Additionally, future research will incorporate emerging technologies and address uncertainty to further enhance real-time control and planning applications.

CRediT authorship contribution statement

A. Belmondo Bianchi: Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **H.H.M. Rijnaarts:** Writing – review & editing, Supervision, Resources, Project administration, Funding acquisition. **S. Shariat Torbaghan:** Writing – review & editing, Visualization, Supervision, Software, Resources, Project administration, Methodology, Funding acquisition, Conceptualization.

Declaration of Generative AI and AI-assisted technologies in the writing process

During the preparation of this work the author(s) used ChatGPT in order to improve readability and language. After using this tool/service, the author(s) reviewed and edited the content as needed and take(s) full responsibility for the content of the publication.

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.

Acknowledgments

This research was performed within the framework of the research program AquaConnect, funded by the Dutch Research Council (NWO, grant-ID P19-45) and public and private partners of the AquaConnect consortium and coordinated by Wageningen University and Research.

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.wroa.2025.100479.

Data availability

Data will be made available on request.

References

Ahmed, A.A., Sayed, S., Abdoulhalik, A., Moutari, S., Oyedele, L., 2024. Applications of machine learning to water resources management: A review of present status and future opportunities. J. Clean. Prod. 441, 140715.

Amos, B., Xu, L., Kolter, J.Z., 2017. Input convex neural networks. In: International Conference on Machine Learning. PMLR, pp. 146–155.

Ayyagari, K.S., Wang, S., Gatsis, N., Taha, A.F., Giacomoni, M., 2021. Energy-efficient optimal water flow considering pump efficiency. In: 2021 IEEE Madrid PowerTech. IEEE, pp. 1–6.

Bagloee, S.A., Asadi, M., Patriksson, M., 2018. Minimization of water pumps' electricity usage: A hybrid approach of regression models with optimization. Expert Syst. Appl. 107, 222–242.

Beale, E.M.L., Tomlin, J.A., 1970. Special facilities in a general mathematical programming system for non-convex problems using ordered sets of variables. OR 69 (447–454), 99.

Bianchi, A.B., 2025. Neural network-informed optimal water flow problem: Modeling, algorithm, and benchmarking. http://dx.doi.org/10.17632/84hjf8xgff.1, URL https://doi.org/10.17632/84hjf8xgff.1.

Bianchi, A.B., Pardo, M., Alskaif, T., Rijnaarts, H., Torbaghan, S.S., 2024. Novel market-based mechanism for flexibility provision by water distribution networks to power systems. IEEE Access.

Bianchi, A.B., Willet, J., Rijnaarts, H., Torbaghan, S.S., 2023. Distribution robust water-based demand side management in power transmission networks. Sustain. Energy Grids Netw. 36, 101232.

Bishop, C.M., Bishop, H., 2023. Deep Learning: foundations and Concepts. Springer Nature.

Bonvin, G., Demassey, S., Le Pape, C., Maïzi, N., Mazauric, V., Samperio, A., 2017. A convex mathematical program for pump scheduling in a class of branched water networks. Appl. Energy 185, 1702–1711.

Boretti, A., Rosa, L., 2019. Reassessing the projections of the world water development report. NPJ Clean Water 2 (1), 15.

Calamai, P.H., Moré, J.J., 1987. Projected gradient methods for linearly constrained problems. Math. Program. 39 (1), 93–116.

Candelieri, A., Perego, R., Archetti, F., 2018. Bayesian optimization of pump operations in water distribution systems. J. Global Optim. 71, 213–235.

Conejo, A.J., Castillo, E., Minguez, R., Garcia-Bertrand, R., 2006. Decomposition Techniques in Mathematical Programming: Engineering and Science Applications. Springer Science & Business Media.

D'Ambrosio, C., Lodi, A., Martello, S., 2010. Piecewise linear approximation of functions of two variables in milp models. Oper. Res. Lett. 38 (1), 39–46.

D'Ambrosio, C., Lodi, A., Wiese, S., Bragalli, C., 2015. Mathematical programming techniques in water network optimization. European J. Oper. Res. 243 (3), 774–788.

Dini, M., Tabesh, M., 2019. Optimal renovation planning of water distribution networks considering hydraulic and quality reliability indices. Urban Water J. 16 (4), 249–258.

Fooladivanda, D., Domínguez-García, A.D., Sauer, P.W., 2018. Utilization of water supply networks for harvesting renewable energy. IEEE Trans. Control. Netw. Syst. 6 (2), 763–774.

Fooladivanda, D., Taylor, J.A., 2017. Energy-optimal pump scheduling and water flow. IEEE Trans. Control. Netw. Syst. 5 (3), 1016–1026.

Fu, G., Jin, Y., Sun, S., Yuan, Z., Butler, D., 2022. The role of deep learning in urban water management: A critical review. Water Res. 223, 118973.

Gambella, C., Ghaddar, B., Naoum-Sawaya, J., 2021. Optimization problems for machine learning: A survey. European J. Oper. Res. 290 (3), 807–828.

Garzón, A., Kapelan, Z., Langeveld, J., Taormina, R., 2022. Machine learning-based surrogate modeling for urban water networks: review and future research directions. Water Resour. Res. 58 (5), e2021WR031808.

Geißler, B., Martin, A., Morsi, A., Schewe, L., 2011. Using piecewise linear functions for solving MINLPs. In: Mixed Integer Nonlinear Programming. Springer, pp. 287–314.

Gu, W., Sioshansi, R., 2025. Operational modeling of the nexus between water-distribution and electricity systems. Curr. Sustainable/Renewable Energy Rep. 12 (1), 1–16.

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., Courville, A.C., 2017. Improved training of wasserstein gans. Adv. Neural Inf. Process. Syst. 30.

Guo, Y., Summers, T.H., 2020. Distributionally robust stochastic optimal water flow and risk management. IFAC-PapersOnLine 53 (2), 16617–16623.

Haaland, S.E., 1983. Simple and explicit formulas for the friction factor in turbulent pipe flow. J. Fluids Eng. 105 (1), 89–90.

Hajgató, G., Gyires-Tóth, B., Paál, G., 2021. Reconstructing nodal pressures in water distribution systems with graph neural networks. arXiv preprint arXiv:2104.13619.

Hajgató, G., Paál, G., Gyires-Tóth, B., 2020. Deep reinforcement learning for real-time optimization of pumps in water distribution systems. J. Water Resour. Plan. Manag. 146 (11), 04020079.

Javadiha, M., Blesa, J., Soldevila, A., Puig, V., 2019. Leak localization in water distribution networks using deep learning. In: 2019 6th International Conference on Control, Decision and Information Technologies (CoDIT). IEEE, pp. 1426–1431.

Karimidastenaei, Z., Avellán, T., Sadegh, M., Kløve, B., Haghighi, A.T., 2022. Unconventional water resources: Global opportunities and challenges. Sci. Total Environ. 827, 154429.

Kavya, M., Mathew, A., Shekar, P.R., Sarwesh, P., 2023. Short term water demand forecast modelling using artificial intelligence for smart water management. Sustain. Cities Soc. 95, 104610.

Li, D., Chen, D., Jin, B., Shi, L., Goh, J., Ng, S.-K., 2019. MAD-gan: Multivariate anomaly detection for time series data with generative adversarial networks. In: International Conference on Artificial Neural Networks. Springer, pp. 703–716.

Li, Q., Yu, S., Al-Sumaiti, A.S., Turitsyn, K., 2018. Micro water–energy nexus: Optimal demand-side management and quasi-convex hull relaxation. IEEE Trans. Control. Netw. Syst. 6 (4), 1313–1322.

Lougee-Heimer, R., 2003. The common optimization interface for operations research: Promoting open-source software in the operations research community. IBM J. Res. Dev. 47 (1), 57–66.

Ma, H., Wang, X., Wang, D., 2024. Pump scheduling optimization in urban water supply stations: A physics-informed multiagent deep reinforcement learning approach. Int. J. Energy Res. 2024 (1), 9557596.

Marchi, A., Salomons, E., Ostfeld, A., Kapelan, Z., Simpson, A.R., Zecchin, A.C., Maier, H.R., Wu, Z.Y., Elsayed, S.M., Song, Y., et al., 2014. Battle of the water networks II. J. Water Resour. Plan. Manag. 140 (7), 04014009.

Martínez-Bahena, B., Cruz-Chávez, M.A., Ávila-Melgar, E.Y., Cruz-Rosales, M.H., Rivera-Lopez, R., 2018. Using a genetic algorithm with a mathematical programming solver to optimize a real water distribution system. Water 10 (10), 1318.

Menke, R., Abraham, E., Parpas, P., Stoianov, I., 2016. Exploring optimal pump scheduling in water distribution networks with branch and bound methods. Water Resour. Manag. 30, 5333–5349.

Nasser, A.A., Rashad, M.Z., Hussein, S.E., 2020. A two-layer water demand prediction system in urban areas based on micro-services and LSTM neural networks. IEEE Access 8, 147647–147661.

Nerantzis, D., Pecci, F., Stoianov, I., 2020. Optimal control of water distribution networks without storage. European J. Oper. Res. 284 (1), 345–354.

Reis, A.L., Lopes, M.A., Andrade-Campos, A., Antunes, C.H., 2023. A review of operational control strategies in water supply systems for energy and cost efficiency. Renew. Sustain. Energy Rev. 175, 113140.

Rossman, L., 2000. EPANET 2 users manual. US environmental protection agency, washington, DC. Technical Report, EPA/600/R-00/057.

Sankaranarayanan, P., Rengaswamy, R., 2022. CDiNN–convex difference neural networks. Neurocomputing 495, 153–168.

Sayers, W., Savic, D., Kapelan, Z., 2019. Performance of LEMMO with artificial neural networks for water systems optimisation. Urban Water J. 16 (1), 21–32.

Singh, M.K., Kekatos, V., 2019. Optimal scheduling of water distribution systems. IEEE Trans. Control. Netw. Syst. 7 (2), 711–723.

Singh, M.K., Kekatos, V., 2020. On the flow problem in water distribution networks: Uniqueness and solvers. IEEE Trans. Control. Netw. Syst. 8 (1), 462–474.

Sit, M., Demiray, B.Z., Xiang, Z., Ewing, G.J., Sermet, Y., Demir, I., 2020. A comprehensive review of deep learning applications in hydrology and water resources. Water Sci. Technol. 82 (12), 2635–2670.

Stuhlmacher, A., Mathieu, J.L., 2020. Chance-constrained water pumping to manage water and power demand uncertainty in distribution networks. Proc. IEEE 108 (9), 1640–1655.

Truong, H., Tello, A., Lazovik, A., Degeler, V., 2024. Graph neural networks for pressure estimation in water distribution systems. Water Resour. Res. 60 (7), e2023WR036741.

Van Hentenryck, P., 2021. Machine learning for optimal power flows. Tutorials Oper. Res.: Emerg. Optim. Methods Model. Tech. Appl. 62–82.

Vieira, B.S., Mayerle, S.F., Campos, L.M., Coelho, L.C., 2020. Optimizing drinking water distribution system operations. European J. Oper. Res. 280 (3), 1035–1050.
Wada, Y., Flörke, M., Hanasaki, N., Eisner, S., Fischer, G., Tramberend, S., Satoh, Y., Van Vliet, M., Yillia, P., Ringler, C., et al., 2016. Modeling global water use for the 21st century: The water futures and solutions (wfas) initiative and its approaches. Geosci. Model. Dev. 9 (1), 175–222.
Walski, T.M., Brill Jr., E.D., Gessler, J., Goulter, I.C., Jeppson, R.M., Lansey, K., Lee, H.-L., Liebman, J.C., Mays, L., Morgan, D.R., et al., 1987. Battle of the network models: Epilogue. J. Water Resour. Plan. Manag. 113 (2), 191–203.
Wen, B., Chen, X., Pong, T.K., 2018. A proximal difference-of-convex algorithm with extrapolation. Comput. Optim. Appl. 69, 297–324.
Wu, T., Wang, J., 2023. Transient stability-constrained unit commitment using input convex neural network. IEEE Trans. Neural Netw. Learn. Syst..
Zamzam, A.S., Dall'Anese, E., Zhao, C., Taylor, J.A., Sidiropoulos, N.D., 2018. Optimal water–power flow-problem: Formulation and distributed optimal solution. IEEE Trans. Control. Netw. Syst. 6 (1), 37–47.
Zhang, Y., Chen, C., Liu, G., Hong, T., Qiu, F., 2020. Approximating trajectory constraints with machine learning–microgrid islanding with frequency constraints. IEEE Trans. Power Syst. 36 (2), 1239–1249.