Bounds and Convex Heuristics for Bi-objective Sensor Placement in Water Networks

Filippo Pecci, Ivan Stoianov

InfraSense Labs, Department of Civil and Environmental Engineering, Imperial College London, London, SW7 2BB, UK

July 2022

Abstract

Optimal sensor placement in water networks has been traditionally formulated to maximize either hydraulic model accuracy or spatial coverage. Because a unique sensor configuration that optimizes both objectives may not exist, these approaches inevitably result in sub-optimal configurations with respect to one of the objectives. This paper presents a new bi-objective optimization problem formulation to investigate the trade-offs between these conflicting objectives. We develop a convex heuristic to approximate the Pareto front, and compute guaranteed bounds to discard portions of the criterion space that do not contain non-dominated solutions. Our method relies on a Chebyshev scalarization scheme and convex optimization. We implement the proposed methods for optimal sensor placement in an operational water network from the UK. For this case study, the convex heuristic computes near-optimal solutions for the individual objective minimization problems, and tight bounds on the true Pareto front of the considered bi-objective optimization problem.

1 Introduction

Hydraulic models play a critical role in supporting water utilities achieving their operational objectives. These include optimal pressure control [WAPS15], detection and localization of water leaks [BPS21, SDG+22], optimal network design [BDL+15, UPS21], improvement of energy efficiency [JLCK15, MAPS16]. Monitoring hydraulic pressure and flows with high temporal and spatial resolution is critical to build and maintain accurate hydraulic models [ZXX+18, WPS20]. In addition, monitoring devices are used to provide confidence in the implementation of advanced control solutions.

The problem of optimal sensor placement in water networks has traditionally been formulated with the objective of minimizing the uncertainty associated with the hydraulic model, estimated using model sensitivities with respect to pipe roughness parameters [BU98].
However, since the most sensitive locations can be close to each other, such approach may result in unsatisfactory locations for wider monitoring purposes, with sensors clustered around few network nodes - see also the numerical experiments reported in this manuscript. Hence, researchers have considered the problem of optimal sensor placement in water networks to maximize spatial coverage, using metrics based on network topology [DWS00, FGGV15].

Because a single sensor configuration that is optimal for both sensitivity and topology based metrics may not exist, we propose a novel bi-objective problem formulation for optimal sensor placement in water networks. As common in optimal experiment design literature [JB09, KGW12, YA21, AZLSW21], we consider D-optimality as sensitivity-based metric to evaluate the performance of a given sensor configuration. Such convex non-linear objective function aims to minimize the volume of the confidence ellipsoid around the estimated hydraulic parameters. In comparison, the topology metric used by [FGGV15] aims to generate a spatially even distribution of sensors, by minimizing the sum of the distances between network nodes and their closest sensors. We formulate a bi-objective problem to optimize D-optimality and the topology-based metric. Furthermore, the placement of sensors at network nodes is modeled by binary decision variables, yielding a convex bi-objective mixed integer non-linear optimization problem.

In bi-objective optimization, the best compromises between the two objective functions are characterized by the notion of Pareto optimality. The Pareto optimal set is made of those feasible solutions such that no other solution exist that reduces one objective function value without increasing the other. The image of the Pareto optimal set through the objective functions is called Pareto front, and it is included in the criterion space, i.e. the space of the objective function values. Due to the presence of binary decision variables, the Pareto front of the considered problem may be disconnected and composed of isolated points. This complicates the generation of the complete Pareto front, and exact solution methods rely on computationally intensive techniques such as branch and bound [DENR20, DGP20]. As a result, the considered bi-objective optimal sensor placement problems can become computationally impractical to solve exactly when large water networks are considered. Here, we focus on techniques to approximate the Pareto front, discarding portions of criterion space that do not contain any point of the true Pareto front.

We develop a Chebyshev scalarization scheme, which converts the original bi-objective optimization problem into a series of parameterized, single-objective, mixed integer programs (MIPs). These MIPs are mixed-integer semidefinite programs whose size grows with $n^2$, where $n$ is the number of nodes in the network. Hence, the computation of certified globally optimal solutions can be impractical when large water networks are considered. We implement a heuristic based on convex optimization to generate feasible solutions with lower bounds on the globally optimal values. Furthermore, we investigate the use of feasible solutions and their associate lower bounds to derive an approximation of the Pareto Front with guaranteed bounds. In particular, we prove that the lower bounds of the Chebyshev scalarized MIPs define a subset of the criterion space that does not contain the image of any feasible solution through the objective functions. We evaluate the proposed methods
2 Background and problem formulation

We consider a water distribution network with \( n_p \) links and \( n_n \) nodes, operating over a time interval \( \{1, \ldots, n_t\} \). Let \( h_{0k} \in \mathbb{R}^{n_n} \) be a known vector of hydraulic heads at water inlets, \( d_k \in \mathbb{R}^{n_n} \) be a known vector of nodal demands, and \( \eta_k \in \mathbb{R}^{n_r} \) be a known vector of head losses at pressure control valves, for all \( k \in \{1, \ldots, n_t\} \). Let \( \theta \in \mathbb{R}^{n_r} \) be a vector of roughness parameters, where each index \( r \in \{1, \ldots, n_r\} \) corresponds to a group of pipes that share the same roughness parameter \( \theta_r \). Network hydraulic heads \( h_k \in \mathbb{R}^{n_n} \) and flows \( q_k \in \mathbb{R}^{n_p} \) are computed by solving the following non-linear equations:

\[
A_{11}(q_k, \theta)q_k + A_{12}h_k + A_{10}h_{0k} + A_{13}\eta_k = 0 \\
A_{12}^T q_k = d_k.
\] (1)

The vector \( A_{11}(q_k, \theta)q_k \) corresponds to the head losses within network links due to friction, and the diagonal matrix \( A_{11}(q_k, \theta) \) is defined by:

\[
A_{11}(q_k, \theta)_{ll} = \rho_l(q_k, \theta_r)|q_k|^{n_l-1}, \quad l \in \{1, \ldots, n_p\},
\] (2)

where \( r_l \in \{1, \ldots, n_r\} \) is the index of the group corresponding to link \( l \in \{1, \ldots, n_p\} \), and \( \rho_l \) and \( n_l \) are resistance coefficient function and exponent associated with link \( l \), respectively, whose formulations depend on the head loss formula used \[LJW99\] Section 2.2. In the case of Hazen-Williams friction model, we have \( n_l = 1.852 \) and

\[
\rho_l(\theta_r) = \frac{10.67 L_l}{\theta_r^{1.852} D_l^{1.871}}.
\] (3)

When the Darcy-Weisbach formula is used, we have \( n_l = 2 \) and \( \rho_l(q_k, \theta_r) \) is a continuously differentiable function whose piece-wise formulation can be found in \[WPS20\].

In the following, we denote by \( \phi_k : \mathbb{R}^{n_r} \to \mathbb{R}^{n_p+n_n} \) the non-linear function such that \( \phi_k(\theta) = [q_k^T h_k^T]^T \) solves (1). Let \( S \subset \{1, \ldots, n_n\} \) and \( S_0 \subset \{1, \ldots, n_p\} \) be index sets corresponding to hydraulic heads and flow measuring devices, respectively. Let measurement \( y_{ik} \) correspond to either a measured flow or a measured hydraulic head at time step \( k \), for each \( i \in S \cup S_0 \).

As done in previous literature, we consider pipe roughness coefficients \( \theta \in \mathbb{R}^{n_r} \) as parameters to be estimated. It is possible to estimate the vector of parameters \( \theta \in \mathbb{R}^{n_r} \) by solving a non-linear inverse problem:

\[
\text{minimize} \quad \frac{1}{n_t|S \cup S_0|} \sum_{k=1}^{n_t} \sum_{i \in S \cup S_0} L(\phi_{ik}(\theta), y_{ik}) + g(\theta) \\
\text{subject to} \quad \theta \geq 0
\] (4)
where $L(\cdot, y_{ik})$ and $g$ are suitable convex loss and regularization functions, respectively. For examples on parameter estimation in water networks see \cite{LB91} and \cite{WPS20}.

The quality of the parameter estimates obtained solving (5) heavily depends on the choice of the measuring locations $S$ and $S_0$. The problem of selecting the best sensor locations for parameter estimation has been formulated within the framework of optimal experiment design \cite{Puk06}. We assume that the vector of parameters $\theta \in \mathbb{R}^{n_r}$ and the measurements are related through the following Gaussian additive non-linear model:

$$y_{ik} = \phi_{ik}(\theta) + v_{ik}, \quad i \in S \cup S_0$$  \hspace{1cm} (5)

with $v_{ik} \sim N(0, \sigma^2)$ independent identically distributed random variables. The goal of optimal experiment design is to select sensor locations so that the estimation error is minimized.

In the case of (5), the error covariance matrix is computed using the sensitivity vectors:

$$\Sigma = \sigma^2 \left( \sum_{j \in S_0} A_j A_j^T + \sum_{j \in S} A_j A_j^T \right)^{-1}, \hspace{1cm} (6)$$

with $A_j = [\nabla \phi_{j,1}(\bar{\theta}) \ldots \nabla \phi_{j,n_t}(\bar{\theta})] \in \mathbb{R}^{n_r \times n_t}$, given an initial estimate of the parameter vector denoted by $\bar{\theta} \in \mathbb{R}^{n_r}$. In this study, we exploit the saddle-point structure of Equations (1) and derive a computationally efficient algorithm to compute these sensitivity vectors - see Appendix A1 for a detailed description. We assume that $n_r < |S \cup S_0|$ and vectors $\nabla \phi_{jk}(\bar{\theta})$ span $\mathbb{R}^{n_r}$. If this is not the case, a prior should be imposed on the parameters $\theta \in \mathbb{R}^{n_r}$. If $\theta_l \sim N(0, \lambda)$, we obtain a covariance matrix $\tilde{\Sigma} = \left( \frac{1}{\sigma^2} \sum_{j \in S \cup S_0} A_j A_j^T + \frac{1}{\lambda} I_r \right)^{-1}$, where $I_r \in \mathbb{R}^{n_r \times n_r}$ is the identity matrix. Formulations and methods discussed in the remainder of this manuscript are then applied to $\tilde{\Sigma}$. In addition, we assume that the locations of flow measuring devices $S_0 \subset \{1, \ldots, n_p\}$ are fixed, and we want to optimize the choice of $S \subset \{1, \ldots, n_n\}$. Let binary variables $z \in \{0, 1\}^{n_n}$ be used to model the placement of sensors, with $z_j = 1$ if a sensor is placed on node $j$ or $z_j = 0$ otherwise. Moreover, we set $X_0 = \sum_{j \in S_0} A_j A_j^T$.

Several metrics have been proposed to measure the estimation error associated with the covariance matrix $\Sigma$. A popular choice is D-optimality \cite{BU98, JB09, KGW12, YA21, AZLSW21}, which aims to minimize the volume of the confidence ellipsoid around the estimated parameters:

$$f_D(z) = -\log \det \left( X_0 + \sum_{j=1}^{n_n} z_j A_j A_j^T \right). \hspace{1cm} (7)$$

Metrics used in optimal experiment design rely on sensitivity information to evaluate the estimation error. However, they do not explicitly consider network topological properties. Since the most sensitive locations can be very close to each other, sensitivity-based metrics may result in unsatisfactory sensor configurations. For example, when the number of installed sensors is greater than $2n_r$, it is possible to place sensors at both inlet and outlet of a link for each group $r \in \{1, \ldots, n_r\}$. In this way, the head loss across the link would be measured, reducing the uncertainty associated with the estimated roughness parameter.
for the corresponding group. For these reasons, researchers have also considered topology-based methods, which do not require an initial estimate of the parameters $\hat{\theta}$, and take into account network connectivity to generate an even spread of sensors [DWS00]. In particular, [FGGV15] proposed a facility location problem to minimize:

$$f_T(z) = \sum_{i=1}^{n_n} \min_{j \in N_i \cap I(z)} C_{ij},$$

(8)

where $N_i = \{1, \ldots, n_n\} \setminus \{i\}$ and $I(z) = \{j \in \{1, \ldots, n_n\} \mid z_j = 1\}$. Coefficients $C_{ij}$ are defined as the length of the shortest-path between node $i$ and node $j$, where each link $l$ has been assigned a weight $L_l$, with $l \in \{1, \ldots, n_p\}$. Function $f_T$ can be interpreted as the sum of the distances between network nodes and their closest sensors. The following Lemma shows how to reformulate $f_T$ using linear programming.

**Lemma 2.1.** Let $\hat{z} \in \{0, 1\}^{n_n}$, then $f_T(\hat{z})$ is the optimal value of the following linear program:

$$\begin{align*}
\text{minimize} & \quad \sum_{i=1}^{n_n} \sum_{j=1}^{n_n} C_{ij} u_{ij} \\
\text{subject to} & \quad u_{ij} \leq \hat{z}_j, \quad i \in \{1, \ldots, n_n\}, \; j \in \{1, \ldots, n_n\} \\
& \quad \sum_{j=1}^{n_n} u_{ij} = 1, \quad i \in \{1, \ldots, n_n\} \\
& \quad u_{ii} = 0, \quad i \in \{1, \ldots, n_n\} \\
& \quad u_{ij} \geq 0 \quad i \in \{1, \ldots, n_n\}, \; j \in \{1, \ldots, n_n\}
\end{align*}$$

(9)

**Proof.** See Appendix [A2] \hfill \square

The above lemma implies that the problem of minimizing $f_T$ can be formulated as a mixed integer linear program (MILP) and solved using integer programming techniques [FGGV15]. Because the size of these MILPs grows rapidly with the number of nodes, previous literature has also considered both specialized heuristics [Ree06, UV21], and model reduction approaches for specific network topologies [CW20].

As shown by the numerical experiments reported in Section 5, a single sensor configuration that simultaneously minimizes $f_D$ and $f_T$ may not exist. Therefore, in the next Section, we introduce a new bi-objective problem formulation, and a scalarization method to generate an approximation of the associated Pareto front.
3 Bi-objective sensor placement in water networks

To investigate the trade-offs between sensitivity and topology based metrics, we propose a new bi-objective problem formulation:

\[
\begin{align*}
\text{minimize} & \quad (f_D(z), f_T(z)) \\
\text{subject to} & \quad \sum_{j=1}^{n_n} z_j = m \\
& \quad Gz \leq b \\
& \quad z \in \{0, 1\}^{n_n},
\end{align*}
\]

(10)

where \( m \geq 2 \) corresponds to the number of sensors to be installed. We have included constraints to prevent two sensors from being placed on nodes that are connected by a link, avoiding impractical sensor configurations. To achieve this, we define matrix \( G \in \mathbb{R}^{n_p \times n_n} \) as:

\[
G(l, j) = \begin{cases} 
1 & \text{if link } l \text{ is connected to node } j \\
0 & \text{otherwise}, 
\end{cases}
\]

(11)

and vector \( b = [1, \ldots, 1]^T \). Solving Problem (10) aims to compute the set of Pareto optimal solutions, also known as the efficient set. In fact, the most efficient compromises between \( f_D \) and \( f_T \) are characterized by the notion of Pareto optimality.

\textbf{Definition 3.1.} A sensor configuration \( z^* \) is said to be Pareto optimal for Problem (10) if there is no \( z' \in \{0, 1\}^{n_n} \) feasible for Problem (10), such that \( f_U(z') \leq f_U(z^*) \) for all \( U \in \{D, T\} \), and \( f_Y(z') < f_Y(z^*) \) for some \( Y \in \{D, T\} \).

Let \( \Theta \subset \{0, 1\}^{n_n} \) be the set of Pareto optimal solutions for Problem (10). Due to the presence of binary decision variables, \( \Theta \) is disconnected and composed of isolated points. Because of such complicated structure, researchers have focused on approximating the Pareto front, which is the image of \( \Theta \) through the objective functions:

\[
\begin{align*}
F(\Theta) = \left\{ (f_D(z^*), f_T(z^*)) \mid z^* \in \Theta \right\}.
\end{align*}
\]

(12)

To the best of the authors’ knowledge, previous studies have solved optimal sensor placement problems in water distribution networks using genetic algorithms [KSW03, KSW05, BKSA09]. While being practical approaches for highly non-linear, discrete, or black-box optimization problems, these methods do not provide theoretical guarantee of global non-dominance, and they require a significant computational effort to generate good quality solutions. In comparison, we investigate the application of mathematical optimization methods to approximate the Pareto front with guaranteed bounds. To complete our numerical investigation, Section 5.4 reports a comparison with the state-of-the-art genetic algorithm NSGA-II [DPAM02].
Classical algorithms to approximate the Pareto front (known as criterion space algorithms) rely on scalarization methods, which convert a bi-objective problem into a series of parameterized single-objective optimization problems. In this study, we investigate a Chebyshev scalarization method [Mic98, Section 3.4]. As initial step we consider the individual minimization problems:

\begin{align}
\text{minimize} & \quad f_Y(z) \\
\text{subject to} & \quad \sum_{j=1}^{n} z_j = m \\
& \quad Gz \leq b \\
& \quad z \in \{0,1\}^{n},
\end{align}

for all $Y \in \{D,T\}$. Here, we assume that we have implemented a method to compute a feasible solution $\hat{z}_Y$ and a lower bound to the optimal value of Problem (13), denoted by $f_Y^*$, for all $Y \in \{D,T\}$. Section 4 presents a suitable heuristic based on convex optimization to achieve this. Given $\beta \in (0,1)$, consider the following optimization problem:

\begin{align}
\text{minimize} & \quad f_\beta(z) \\
\text{subject to} & \quad \sum_{j=1}^{n} z_j = m \\
& \quad Gz \leq b \\
& \quad z \in \{0,1\}^{n},
\end{align}

where

\begin{align}
f_\beta(z) = \max \left( w^\beta_D(f_D(z) - f^*_D), w^\beta_T(f_T(z) - f^*_T) \right),
\end{align}

with

\begin{align}
w^\beta_D = \frac{\beta}{f_D(\hat{z}_T) - f^*_D}, \quad w^\beta_T = \frac{1 - \beta}{f_T(\hat{z}_D) - f^*_T}.
\end{align}

While other scalarization approaches (e.g. weighted sum) are limited to the generation of the convex portions of the Pareto front, any Pareto optimal solution can be found solving Problem (10) for an opportunely selected weight. In fact, the following results holds.

**Theorem 3.1.** Let $\hat{z}_\beta$ be a globally optimal solution of Problem (14). Then, $\hat{z}_\beta$ is weakly Pareto optimal.

**Proof.** See Appendix A2

**Theorem 3.2.** Given $z^* \in \Theta$, there exist $\beta^* \in (0,1)$ such that $z^*$ solves Problem (14) for $\beta = \beta^*$.

**Proof.** See Appendix A2
Problem (14) results in mixed-integer semidefinite program (SDP) with exponential cone constraints, whose size grows with $n^2$ - see A5. As a result, obtaining certified globally optimal solutions for Problem (14) can be impractical when large water networks are considered. For this reason, we only assume that we are able to compute a feasible solution $\hat{z}_\beta$ and a lower bound to the optimal value of Problem (14), denoted by $f^*_\beta$ - Section 4 presents a suitable convex heuristic. Consider a sequence $\beta_1, \ldots, \beta_N$ and let $\hat{z}_{\beta_1}, \ldots, \hat{z}_{\beta_N}$ be the corresponding feasible solutions. Furthermore, we include the individual minimizers by defining $\hat{z}_{\beta_0} = \hat{z}_D$ and $\hat{z}_{\beta_{N+1}} = \hat{z}_T$. An approximation of the Pareto front is given by:

$$\mathcal{P} = \left\{ (f_D(\hat{z}_{\beta_k}), f_T(\hat{z}_{\beta_k})) \mid k = 0, \ldots, N + 1 \right\}. \quad (17)$$

**Theorem 3.3.** Define the following set:

$$\mathcal{W} = \left\{ (p_1 + u, p_2 + v) \mid (p_1, p_2) \in \mathcal{P}, u \geq 0, v \geq 0, u + v > 0 \right\}. \quad (18)$$

Then, we have:

$$F(\Theta) \cap \mathcal{W} = \emptyset \quad (19)$$

**Proof.** Let $z^* \in \Theta$ be such that $(f_D(z^*), f_T(z^*)) \in \mathcal{W}$. Without loss of generality, we assume that $f_D(z^*) = f_D(\hat{z}_{\beta_k}) + u$ and $f_T(z^*) = f_T(\hat{z}_{\beta_k}) + v$, for some $k \in \{0, \ldots, N+1\}$ and $u \geq 0, v > 0$. Hence, $f_D(z^*) \geq f_D(\hat{z}_{\beta_k})$ and $f_T(z^*) > f_T(\hat{z}_{\beta_k})$. This contradicts the assumption that $z^* \in \Theta$. Therefore, we must have that $(f_D(z^*), f_T(z^*)) \notin \mathcal{W}$. \hfill \Box

It is also possible to bound the Pareto front from below as follows. Assume to have obtained lower bounds to the optimal value of Problem (14), for different values of $\beta$, denoted by $f^*_\beta_1, \ldots, f^*_\beta_N$. Define the following sets:

$$\mathcal{L} = \left\{ \left( \frac{f^*_\beta_k}{w^*_{\beta_k}}, \frac{f^*_\beta_k}{w^*_{\beta_k}} \right) \mid k = 1, \ldots, N \right\}$$

$$\mathcal{R} = \left\{ (p_1, p_2) \in \mathbb{R}^2 \mid p_1 < f^*_D \text{ or } p_2 < f^*_T \right\} \quad (20)$$

$$\mathcal{Q} = \left\{ (p_1 - u, p_2 - v) \mid (p_1, p_2) \in \mathcal{L}, u > 0, v > 0 \right\}.$$

**Theorem 3.4.** The set $\mathcal{R} \cup \mathcal{Q}$ does not contain the image of any feasible point for Problem (10) through the objective functions $(f_D, f_T)$. In particular:

$$F(\Theta) \cap (\mathcal{R} \cup \mathcal{Q}) = \emptyset \quad (21)$$

**Proof.** Let $\tilde{z}$ be feasible for Problem (10) and assume that $(f_D(\tilde{z}), f_T(\tilde{z})) \in \mathcal{R}$. This implies that either $f_D(\tilde{z}) < f^*_D$ or $f_T(\tilde{z}) < f^*_T$. But this contradicts the assumption that $f^*_D$ and $f^*_T$ are lower bounds to the optimal values of the individual minimization problems for $f_D$ and
$f_T$, respectively. Hence, it must be that $\tilde{z} \not\in \mathcal{R}$. Now assume that $(f_D(\tilde{z}), f_T(\tilde{z})) \in \mathcal{Q}$. This implies that there exist $k \in \{1, \ldots, N\}$, $u > 0$, and $v > 0$ such that

$$f_D(\tilde{z}) = \frac{f^*_k}{w^*_D} + f^*_D - u, \quad f_T(\tilde{z}) = \frac{f^*_k}{w^*_T} + f^*_T - v$$

(22)

Hence, we have:

$$f_D(\tilde{z}) < \frac{f^*_k}{w^*_D} + f^*_D, \quad f_T(\tilde{z}) < \frac{f^*_k}{w^*_T} + f^*_T,$$

(23)

which is equivalent to:

$$w^*_D (f_D(\tilde{z}) - f^*_D) < f^*_k, \quad w^*_T (f_T(\tilde{z}) - f^*_T) < f^*_k.$$

(24)

This yields $f^*_k(\tilde{z}) < f^*_k$, which contradicts the assumption that $f^*_{k}$ is a lower bound to the optimal value of Problem (14) with $\beta = \beta_k$. Therefore, we conclude that $(f_D(\tilde{z}), f_T(\tilde{z})) \not\in \mathcal{Q}$. 

Figure 1 shows examples of sets $\mathcal{P}$, $\mathcal{W}$, $\mathcal{R}$, and $\mathcal{Q}$, with different values of $N$. We observe that, as we increase the number of weights, the proposed method can provide a tight outer-approximation of the true Pareto front, which is included in the white area between sets $\mathcal{W}$ and $\mathcal{R} \cup \mathcal{Q}$.

![Figure 1: Examples of Pareto front approximations computed by Algorithm 1](image)

The proposed Chebyshev scalarization method is summarized in Algorithm 1. The method returns a Pareto front approximation $\mathcal{P}$ and sets $\mathcal{W}$, $\mathcal{R}$, and $\mathcal{Q}$, which bound the region containing the true Pareto front.
Algorithm 1 Chebyshev scalarization with global bounds.

**Input:** Matrices $A_1, \ldots, A_n, C, N$.

**Output:** Sets $\mathcal{P}, \mathcal{W}, \mathcal{R}, \mathcal{Q}$.

Apply a suitable solver to Problem (13) to minimize $f_D$ and obtain a feasible solution $\hat{z}_D$ and lower bound $f_D^*$. 

Apply a suitable solver to Problem (13) to minimize $f_T$ and obtain a feasible solution $\hat{z}_T$ and lower bound $f_T^*$. 

Define $P \leftarrow \{f_D(\hat{z}_D), f_T(\hat{z}_D)\} \cup \{f_D(\hat{z}_T), f_T(\hat{z}_T)\}$ and $L \leftarrow \emptyset$. 

Generate a sequence of weights $\beta_1, \ldots, \beta_N$ for $k = 1, \ldots, N$ do

Apply a suitable solver to Problem (14) with $\beta = \beta_k$ and obtain a feasible solution $\hat{z}_{\beta_k}$ and lower bound $f_{\beta_k}^*$. 

$P \leftarrow P \cup \{(f_D(\hat{z}_{\beta_k}), f_T(\hat{z}_{\beta_k}))\}$. 

$L \leftarrow L \cup \left\{(\frac{f_D}{w_{\beta_k}^D} + f_D^*, \frac{f_T}{w_{\beta_k}^T} + f_T^*)\right\}$. 

end for

Define sets $\mathcal{W}, \mathcal{R}, \mathcal{Q}$ as in (18) and (20).

4 Solution of single-objective sensor placement problems

Assuming that either $f(z) = f_D(z)$, $f(z) = f_T(z)$, or $f(z) = f_{\beta}(z)$, Problems (13) and (14) correspond to the following mixed integer optimization problem:

$$
\begin{align*}
\text{minimize} & \quad f(z) \\
\text{subject to} & \quad \sum_{j=1}^{n} z_j = m \\
& \quad Gz \leq b \\
& \quad z \in \{0, 1\}^n.
\end{align*}
$$ (25)

When $f(z) = f_T(z)$, Lemma 2.1 implies that Problem (25) is re-written as a mixed integer linear program - see also A5. In comparison, when $f(z) = f_D(z)$ and $f(z) = f_{\beta}(z)$, Problem (25) results in a mixed-integer semidefinite program with exponential cone constraints. In particular, when $f(z) = f_{\beta}(z)$, the size of these mixed-integer problems grows with $n^2$. Hence, the implementation of exact methods based on branch and bound techniques can become impractical. In this section, we propose a heuristic based on convex optimization to solve Problem (25). We develop a convex heuristic based on two steps.

1. Relax the binary constraints and solve the following continuous optimization problem to obtain a vector of fractional values $z^* \in [0, 1]$ and a lower bound $f^* = f(z^*)$ to the
optimal value: \[
\minimize f(z)
\]
subject to \[
\sum_{j=1}^{n_n} z_j = m \\
Gz \leq b \\
z \in [0,1]^{n_n}.
\] (26)

2. Implement a heuristic algorithm to generate a good quality feasible solution \( \hat{z} \in \{0,1\}^{n_n} \).

Observe that the proposed convex heuristic results in a bound on the level of sub-optimality of \( \hat{z} \), given by:
\[
\text{Gap} = \frac{f(\hat{z}) - f^*}{f^*}
\] (27)

When \( f(z) = f_T(z) \), Lemma (2,1) implies that Problem (26) is an equivalently re-written as a linear program. Hence, it can be efficiently solved by state-of-the-art linear solvers e.g. GUROBI [Gur21]. In the cases of \( f_D \) and \( f_\beta \), we reformulate the convex non-linear Problem (26) as a SDP with exponential cone constraints, and solve it using the convex optimization solver MOSEK [MOS22]. Since the size of the considered SDPs grows with \( n_n^2 \), it might be necessary to implement tailored decomposition approaches [GCC21] to derive scalable solution methods for problems formulated on large water networks e.g. with \( n_n > 10^4 \). These SDP reformulations are needed because the direct implementation of state-of-the-art non-linear programming solver IPOPT [WB06] resulted in a challenging computational performance, as discussed in Section 5.3. Details on these reformulations are given in A5.

In the remainder of this Section, we present the heuristic algorithm.

Let \( z^* \in [0,1] \) be a solution of Problem (26), and \( f^* = f(z^*) \) be the corresponding lower bound to the optimal value of Problem (25). We investigate a heuristic method to compute a feasible solution for Problem (25), when either \( f(z) = f_D(z) \), \( f(z) = f_T(z) \), or \( f(z) = f_\beta(z) \). The proposed method builds on a Fedorov exchange algorithm by [JB09], adding the handling of constraints on allowed sensor locations and different objective functions in addition to \( f_D \). We refer to the implemented algorithm as Round-and-Swap, see Appendix A3 for a pseudo-code description.

As first step, the fractional values in \( z^* \) are rounded to obtain a vector of binary values \( \hat{z} \in \{0,1\}^{n_n} \) that satisfies the problem’s constraints. Since we expect the largest elements in \( z^* \) correspond to good quality sensor configurations, we round the largest \( m \) elements in \( z^* \) to 1, and to 0 the others. We also check that each selected location does not result in violation of the linear constraints, in which case we select the next location according to the descending order of \( z^* \)-values.

Next, we investigate alternative sensor configurations by considering those that are obtained swapping one of the \( m \) locations identified by \( \hat{z} \) for one of the \( n_n - m \) locations that were not selected. When we find a feasible sensor configuration that reduces the objective function value, we update \( \hat{z} \in \{0,1\}^{n_n} \). If we find that no swap reduces the objective function
compared to the current solution, the algorithm terminates. In comparison to the algorithm in [JB09], at each swapping iteration, we do not only evaluate the objective function value of the new sensor configuration, but we also check that it satisfies the linear constraints. In this way, the algorithm is guaranteed to generate feasible solutions.

While the swapping process is guaranteed to terminate after a finite number of iterations, it can still require a large number of function evaluations. Hence, we define a maximum number of iterations $\text{MaxIter}$, which is set to $10^5$ in our simulations. The largest entries in $z^*$ are expected to include the globally optimal sensor locations, and this can help to reduce the number of iterations needed to converge. In particular, we consider for swapping only locations corresponding to entries in $z^*$ with values between 0.1 and 0.9. We identify locations to be removed from the set of selected sensor locations in ascending order based on their $z^*$-values. Analogously, the sensor locations to be added to the set of selected locations are chosen in descending order based on their values in $z^*$.

5 Numerical experiments

All experiments reported in this Section are carried out in Julia 1.7.2 [BEKS17], where optimization solvers are called through JuMP 1.0.0 [DHL17]. All computations are performed on a laptop with Intel(R) Core(TM) i9-9980HK CPU @ 2.40GHz and 32 GB of RAM. Linear programs (A20) are solved by GUROBI [Gur21, version 9.1]. Semidefinite programs (A22) and (A25) are solved using MOSEK [MOS22, version 9.3].

We implement the developed methods for optimizing the locations of pressure sensors to be installed in BWFLnet, the hydraulic model of an operational water network from the UK - see Figure 2a. The network has $n_0 = 2$ inlets, $n_n = 2221$ nodes, and $n_p = 2281$ links, including 3 pressure control valves (PCVs) which are optimally controlled to minimize Average Zone Pressure [WAPS15]. The EPANET model for BWFLnet is available at the Mendeley data repository in [PSO21]. In addition, BWFLnet is operated with a dynamically adaptive topology, where two dynamic boundary valves (DBVs) are open during diurnal operation and closed at night for leakage management purposes. The network also includes a number of throttle control valves (TCVs), that are assumed to be fully open. We consider all valves’ coefficients as known and we do not perform parameter estimation on these links. The hydraulic model has 13 different pipe groups, which were determined by the network operator. Two of these groups, namely groups 8 and 9, consists of leaf links whose roughness parameters can be estimated only by installing sensors at both ends of the pipe. Hence, for the purpose of optimal sensor placement, these groups are discarded. As a result, we consider $n_\theta = 11$ groups in BWFLnet, as showed in Figure 2b.

Pressure sensors are installed at inlets and outlets of PCVs and DBVs, the top water inlet, and at the two critical points, defined as the nodes with the lowest pressure. This results in 13 pressure sensors whose locations are fixed. Other 24 pressure sensors have been placed according to expert engineering judgment - see Figure 2a. In this study, we solve optimal sensor placement problems in BWFLnet, and compare the obtained solutions with the expert choice. Within the problem formulation, we ignore the expert choice, and we incorporate
5.1 Single-objective optimization

We first implement the developed convex heuristic to individually minimize $f_D$ and $f_T$ in BWFLnet, i.e. solve Problem (13). The number of sensors to be added to the fixed 13 measuring devices ranges from 2 to 40. Hence, we consider the installation of 15 to 53 pressure sensors. We formulate Problem (13) for each value of $m \in \{15, \ldots, 53\}$, and implement the developed convex heuristic. As a first step, we solve a convex relaxation (26) of the resulting optimization problem. Then, we implement the Round-and-Swap algorithm to compute a good quality integer solution. The implemented method returns both lower and upper bounds to the optimal value of the original mixed-integer optimization problem.

The results obtained for $f_D$ are reported in Figure 3 and we observe that the gap between
lower and upper bounds is small for most instances. Therefore, with the exceptions of problem instances with two and three additional sensors, all computed feasible solutions are proven to be globally optimal or near-optimal. While the results for other problem instances suggest that the solutions for \( m \in \{15, 16\} \) are near-optimal, the lower bounds for these instances are not as tight. Similarly, in the case of \( f_T \), the implemented convex heuristic successfully computes near-optimal feasible solutions for the majority of problem instances - see Figure 4. Figure 4b shows that computed lower and upper bounds have larger differences only for few problem instances, suggesting that the Round-and-Swap algorithm has converged to sub-optimal solutions in these cases.

Figures 3 and 4 enable a cost-benefit analysis of different instrumentation levels, as well as compare the existing sensor configuration against optimal solutions. In particular, these results show that the computed solutions with the same number of sensors as the expert choice (i.e. \( m = 37 \)) dominate the existing sensor configuration. In fact, in the case \( f_D \), the convex heuristic results in a bound on the optimality gap equal to 0.64%, while the bound on the gap of the expert solution is 58%. In the case of \( f_T \), the heuristic has converged to a solution that is within 0.36% of optimality, while the expert solution has a bound on the optimality gap equal to 66%.

Although the considered sensor placement problem is solved off-line, and there are no constraints on computational speed and time, the analysis of the computational effort provides insights into the implemented problem formulations and algorithms. In the case of D-optimality, the convex Problem (26) is rewritten as a semidefinite program - see Problem (A22) in A5. Even though the problem includes thousands of variables, the convex solver MOSEK has successfully converged in all tested instances with limited computational effort - the maximum computational time was 0.55 seconds. Analogously, the Round-and-Swap algorithm converged in less than 300 hundreds iterations (i.e. function evaluations) for all problems. As a result, the longest computational time was 0.07 seconds. In the case of

### Table: Upper and Lower Bounds

| Number of Sensors | Upper Bound | Lower Bound | Expert |
|-------------------|-------------|-------------|--------|
| 15                | 30          | 20          |        |
| 20                | 25          | 15          |        |
| 25                | 20          | 10          |        |
| 30                | 15          | 5            |        |
| 35                | 10          | 0            |        |
| 40                | 5           |              |        |
| 50                | 0           |              |        |

### Figure 3: Results obtained by the convex heuristic (CVX) when implemented to install 15 to 53 pressure sensors in BWFLnet for minimizing \( f_D \).
When minimizing \( f_T \), we report a more challenging computational performance. In fact, the formulation of Problem (A20) results in a linear program with 4,935,062 variables. The state-of-the-art solver GUROBI converged in all problem instances, with a maximum computational time of 250 seconds. In comparison, the Round-and-Swap algorithm took less 500 iterations in the majority of tested instances, but some problem formulations required up to 15,164 iterations (i.e. function evaluations). As a consequence, the maximum computational time was 190 seconds.

In Figure 5, we report the sensor configurations obtained when minimizing \( f_D \) and \( f_T \), respectively, when the number of sensors to install is \( m = 37 \). As expected, when \( f_D \) is minimized, we do not obtain an even spread of sensors, which appear to be grouped around certain network links to minimize the uncertainty around the estimated parameters. However, pressure measuring devices are commonly used for additional tasks, for example leak localization [BPS21]. In such cases, a more uniform sensor distribution is preferred. This is obtained by minimizing \( f_T \), as shown in Figure 5b.

### 5.2 Bi-objective optimization

An ideal sensor configuration corresponds to a compromise between accuracy of hydraulic model calibration, and uniform spatial distribution. In order to investigate sensor configurations representing trade-offs between \( f_D \) and \( f_T \), we formulate the bi-objective Problem (10) to optimize the locations of \( m = 37 \) measuring devices, including the 13 fixed sensors. We implement the Chebyshev scalarization scheme in Algorithm 1 with \( N = 20 \), where single-objective optimization problems are solved by a heuristic to compute feasible solutions and lower bounds on the optimal values - see Section 4. The algorithm generates a set of potentially non-dominated points \( \mathcal{P} \), which corresponds to 22 feasible solutions. This set is an approximation of the true Pareto front, computed by the convex heuristic. In addition, Algorithm 1 generates sets \( \mathcal{R} \) and \( \mathcal{Q} \) such that no point of the true Pareto front is included in
Minimization of $f_D$. Minimization of $f_T$.

Figure 5: Optimized valve configurations for individual objectives with $m = 37$.

$\mathcal{R} \cup \mathcal{Q}$ - see Theorem 3.4. As a result, $\partial(\mathcal{R} \cup \mathcal{Q})$ is a bound on the Pareto front, as it limits the space that does not contain non-dominated points (i.e. points on the Pareto front). This is presented in Figure 6 which reports the results computed by Algorithm 1. The scalarization method has generated an even spread of potentially non-dominated points, which represents the trade-offs between $f_D$ and $f_T$. Algorithm 1 requires the implementation of the convex heuristic to solve 20 instances of Problem (14), resulting in a total computational time of roughly 4.5 hours. The computational effort is dominated by the solution of the convex relaxation, which is rewritten as Problem (A25). This results in a semidefinite program with 4,935,073 scalar variables, one matrix variable and 11 exponential cone constraints. The average computational time for MOSEK to converge to a solution was 812 seconds. In comparison, the largest number of iterations in the Round-and-Swap algorithm was 1522, which resulted in a maximum computational time of approximately 16 seconds.

We observe that, for the considered case study, sets $\mathcal{W}$ and $\mathcal{R} \cup \mathcal{Q}$ in Theorems 3.3 and 3.4, respectively, results in tight bounds on the Pareto front. This suggests that set $\mathcal{P}$ computed by Algorithm 1 is close to the true Pareto front of Problem (10). Figure 7 illustrates the trade-offs between $f_D$ and $f_T$. The highlighted point on the approximated Pareto front in Figure 7a represents a compromise between sensitivity and topology based optimality definitions. As shown in Figure 7b, this results in a sensor configuration with a better spatial coverage compared to Figure 5a. Therefore, such a solution meets other operational objectives in addition to hydraulic model calibration, including detection and localization of water leaks.

Finally, we have tested the proposed methods on other network models and no significant difference was observed. Appendix A4 reports the results obtained for a different case study, the water network model L-TOWN-A [VET+20]. Similarly to BWFLnet, Algorithm 1 is able to approximate the Pareto fronts with tight bounds. However, we decided to not include these results in the main text as they do not provide any additional insight.
Figure 6: Approximation of the Pareto front obtained by Algorithm 1 with $N = 20$, implemented to solve the bi-objective sensor location problem of installing 37 sensors in BWFLnet.

Figure 7: Point on the Pareto front with corresponding sensor configuration.
5.3 Numerical experiments with IPOPT

We have investigated the implementation of the state-of-the-art non-linear programming solver IPOPT \cite{WB06} to solve the considered convex problems without the need for semidefinite programming reformulations, i.e. directly considering Problems \cite{A21} and \cite{A23}. We consider the same case study BWFLnet as in Section 5. In the implementation of IPOPT, we have passed exact Jacobian and Hessian matrices through the low-level C-interface provided by Ipopt.jl \cite{https://github.com/jump-dev/Ipopt.jl}. In the case of the individual minimization of $f_D$ (i.e. Problem \cite{A21}), IPOPT required an average computational time of 238 seconds, compared to the average computational time associated with MOSEK, which was 0.35 seconds. In the case of $f_\beta$ (i.e. Problem \cite{A23}), the implementation of IPOPT resulted in an out-of-memory error. This challenging computational performance can be explained by the structure of Problems \cite{A21} and \cite{A23}, which result in large and dense Jacobian and Hessian matrices. As example, in the case of $f_\beta$, the number of non-zero elements within the Jacobian matrix is equal to 19,704,782. These experiments supports the choice of reformulating these optimization problems as SDPs and solving them using MOSEK. Not only such reformulations do not require the analytical derivation of Jacobian and Hessian matrices, but they also result in faster solution times. We also expect off-the-shelf convex mixed-integer non-linear programming solvers like BONMIN \cite{BBC+08} to have a challenging computational performance when applied to the considered MIPs, because they rely on IPOPT as non-linear programming solver.

5.4 Application of NSGA-II

We apply the Non-Dominated Sorting Genetic Algorithm-II (NSGA-II) \cite{DPAM02} to solve the considered bi-objective mixed integer optimization problem to minimize $f_D$ and $f_T$. The solver is implemented through Platypus \cite{https://github.com/Project-Platypus/Platypus}. Because Algorithm \cite{1} required a computational time of 4.5 hours, we set a time limit of 6 hours. As shown in Figure 8, the solutions computed by NSGA-II are dominated by those obtained with the convex heuristic. Moreover, Algorithm \cite{1} results in additional information compared to the genetic algorithm. In fact, not only it computes a set of potentially non-dominated solutions, but also theoretical bounds on the true Pareto front.

6 Conclusions

We have proposed a new bi-objective problem formulation for optimal placement of pressure sensors in water networks. The considered objectives are the minimization of uncertainty in hydraulic model parameters and the maximization of spatial coverage. The resulting bi-objective optimization problem belongs to the class of convex mixed-integer non-linear programming. We have considered a Chebyshev scalarization method, which approximates the Pareto front solving a sequence of convex mixed integer non-linear programs. Computing exact solutions for these optimization problems can be impractical when the number of integer variables is large. Hence, we have implemented a convex heuristic to generate
feasible solutions with lower bounds on the globally optimal values. Furthermore, we have demonstrated that such lower bounds can be used to define a region containing the true Pareto front of the bi-objective optimization problem. These results are not restricted to the framework of optimal sensor placement and they can inform researchers working on bi-objective optimization in other application areas. We evaluate the proposed methods using an operational water network from the UK with over 2000 nodes. The developed convex heuristic computes near-optimal solutions for the individual objective minimization problems, and tight bounds on the true Pareto front of the bi-objective optimization problem. The overall computational time required to generate 20 points on the (approximated) Pareto front was less than 5 hours, the vast majority of which was spent solving convex relaxations. This suggests that the convex heuristic can be used to solve bi-objective sensor placement problems arising in practical applications. Hence, the presented methodology is a stepping stone in the development of computational tools to support the upgrade of existing infrastructure with the implementation of advanced monitoring and control solutions. Future work should extend this framework to take into account the different sources of uncertainty inherent in hydraulic models, including the status of partially closed valves, pipes’ lengths and diameters, and estimated customer demands.

A1 Efficient computation of sensitivity vectors

We develop a computationally efficient method to calculate the vectors of sensitivities $\nabla \phi_{ik}(\bar{\theta})$ for all $i \in \{1, \ldots, n_p + n_n\}$, $k \in \{1, \ldots, n_t\}$. We have that

$$\nabla \phi_{ik}(\bar{\theta}) = J_k^T e_i,$$  \hspace{1cm} (A1)
where \( J_k \) is the Jacobian matrix of function \( \phi_k \) evaluated at \( \bar{\theta} \), and \( e_i \) is a vector of the canonical basis of \( \mathbb{R}^{n_p+n_n} \). Recall that \( \phi_k(\bar{\theta}) = [\bar{q}_k^T \bar{h}_k^T]^T \) is such that

\[
A_{11}(\bar{q}_k, \bar{\theta})\bar{q}_k + A_{12}\bar{h}_k + A_{10}h_{0k} + A_{13}\eta_k = 0
\]

\[
A_{12}^T\bar{q}_k = d_k
\]  

(A2)

Hence, by the implicit function theorem, we have that the Jacobian of function \( \phi_k \) is given by:

\[
J_k = -\left[N A_{11}^{kT} A_{12} 0 \right]^{-1} \left[B^k \right]
\]  

(A3)

where \( N = \text{diag}(n_1, \ldots, n_{p_r}) \), \( A_{11}^k = A_{11}(\bar{q}_k, \bar{\theta}) \), and \( B^k \in \mathbb{R}^{n_p \times n_r} \) is defined by:

\[
B_{ij}^k = \frac{\partial \rho_i(\bar{\theta})}{\partial \theta_j}|_{\bar{q}_k} , \quad l \in \{1, \ldots, n_p\}, j \in \{1, \ldots, n_r\}
\]  

(A4)

Hence, we have:

\[
\nabla \phi_{ik}(\bar{\theta}) = -[(B^k)^T 0] \left[N A_{11}^{kT} A_{12} 0 \right]^{-1} e_i
\]  

(A5)

The sensitivity vector is given by:

\[
\nabla \phi_{ik}(\bar{\theta}) = -(B^k)^T \delta_q^{(ik)},
\]  

(A6)

with vectors \( \delta_q^{(ik)} \in \mathbb{R}^{n_p} \) and \( \delta_h^{(ik)} \in \mathbb{R}^{n_n} \) solutions of the linear system:

\[
\left[\begin{array}{cc} NA_{11} & A_{12} \\ A_{12}^T & 0 \end{array}\right] \begin{bmatrix} \delta_q^{(i,k)} \\ \delta_h^{(i,k)} \end{bmatrix} = \begin{bmatrix} e_{1i} \\ e_{2i} \end{bmatrix}
\]  

(A7)

where \( e_i = [e_{1i}^T e_{2i}^T]^T \). Hence, to compute the sensitivity vectors, we need to solve \( n_t(n_p+n_n) \) linear systems like (A7). This would require computing \( n_t \) different factorizations for the matrix in (A7), which has order \( n_p + n_n \). When large water networks are considered (i.e. \( n_p > 1000 \)), this results in significant computational effort. However, system (A7) has a saddle point structure \[BGL05\]. Similar matrices are also encountered when solving (A2) using the Newton-Raphson method. Previous work has proposed a null-space algorithm to solve saddle point systems analogous to (A7) at each Newton-Raphson iteration - see \[AS16\]. Here, we take a similar approach, and we derive a null-space method to solve (A7).

The incidence matrix \( A_{12} \) has full column rank in the case of water distribution networks \[ESD+14\]. Moreover, we assume \( \text{Ker}(A_{11}^k) \cap \text{Ker}(A_{12}^T) = \{0\} \), which is equivalent to requiring that the water network does not include loops where all flows are equal to zeros. Let \( Z \in \mathbb{R}^{n_p \times (n_p-n_n)} \) be null-space basis of \( A_{12}^T \). Then, any vector \( \delta_q^{(ik)} \) satisfying (A7) is written as \( \delta_q^{(ik)} = A_{12}w + Zv \), for \( w \in \mathbb{R}^{n_n} \) such that \( A_{12}^TA_{12}w = e_{2i} \) and \( v \in \mathbb{R}^{n_p-n_n} \). By substituting for \( \delta_q^{(ik)} \) into the first block of equations in (A7), and pre-multiplying by \( Z^T \), we obtain:

\[
Z^TNA_{11}^kZv = Z^T(e_{1i} - NA_{11}^kA_{12}w),
\]  

(A8)
where we used the fact that \( Z^T A_{12} = 0 \). Instead of the linear system (A7), we now have to solve the smaller system of linear equations (A8). The proposed method requires computing factorizations of \( A_{12}^T A_{12} \) and \( Z^T N A_{11}^k Z \), which are symmetric positive definite matrices (SPD) of orders \( n_n \) and \( n_p - n_n \), respectively. In fact, since \( A_{12} \) has full rank, \( A_{12}^T A_{12} \) is SPD. Moreover, \( \text{Ker}(A_{11}^k) \cap \text{Ker}(A_{12}^T) = \{0\} \) implies that \( Z^T N A_{11}^k Z \) is SPD. The overall algorithm to compute sensitivity vectors \( \nabla \phi_{ik}(\theta) \) for all \( i \in \{1, \ldots, n_p + n_n\} \) and \( k \in \{1, \ldots, n_t\} \) is outlined in Algorithm 2. It is necessary to compute a single Cholesky factorization of \( A_{12}^T A_{12} \), and \( n_t \) different Cholesky factorizations for \( Z^T N A_{11}^k Z \). All linear systems are then solved by forward and back substitutions.

**Algorithm 2** Null-space method for computing sensitivity vectors

**Input:** matrices \( A_{12}, A_{11}^1, \ldots, A_{11}^{n_t}, N \)

**Output:** \( \nabla \phi_{ik}(\theta) \) for all \( i \in \{1, \ldots, n_p + n_n\}, k \in \{1, \ldots, n_t\} \).

Compute a null-space basis of \( A_{12}^T \) denoted by \( Z \in \mathbb{R}^{n_p \times (n_p - n_n)} \).

Compute a Cholesky factorization \( A_{12}^T A_{12} = LL^T \).

For \( i \in \{1, \ldots, n_p + n_n\} \) do

- Let \( e_i = [e_{1i}^T, e_{2i}^T]^T \) be the \( i \)-th vector of the canonical basis of \( \mathbb{R}^{n_p + n_n} \).
- Let \( w_i \in \mathbb{R}^{n_n} \) be the solution of \( LL^T w_i = e_{2i} \).

End for

For \( k = 1, \ldots, n_t \) do

- Compute a Cholesky factorization \( Z^T N A_{11}^k Z = P^k(P^k)^T \).

For \( i = 1, \ldots, n_p + n_n \) do

- Solve \( P^k(P^k)^T v_i^k = Z^T (e_{1i} - N A_{11}^k A_{12} w_i) \).
- Set \( \delta_q^{(ik)} = A_{12} w_i + Z v_i^k \).
- \( \nabla \phi_{ik}(\theta) = -(B^k)^T \delta_q^{(ik)} \).

End for

End for
A2 Technical proofs

Proof of Lemma 2.1
To show this, let \( u \in \mathbb{R}^{n \times n} \) be a feasible for Problem (9) in the main manuscript. We have:

\[
\sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} u_{ij} = \sum_{i=1}^{n} \sum_{j \in N_i \cap I(\hat{z})} C_{ij} u_{ij} \geq \sum_{i=1}^{n} \min_{j \in N_i \cap I(\hat{z})} C_{ij} \left( \sum_{j \in N_i \cap I(\hat{z})} u_{ij} \right) = \sum_{i=1}^{n} \min_{j \in N_i \cap I(\hat{z})} C_{ij} \left( \sum_{j=1}^{n} u_{ij} \right) = f_T(\hat{z}) \quad (A9)
\]

Let \( j^* \in \arg\min \{ C_{ij} \mid j \in N_i \cap I(\hat{z}) \} \). Then, define \( \hat{u} \in \mathbb{R}^{n \times n} \) by setting:

\[
\hat{u}_{ij} = \begin{cases} 
1 & \text{if } j = j^* \\
0 & \text{otherwise}
\end{cases}, \quad \forall j \in \{1, \ldots, n\}, \quad (A10)
\]

for each \( i \in \{1, \ldots, n\} \). By construction, \( \hat{u} \) is feasible for Problem (9), and

\[
\sum_{i=1}^{n} \sum_{j=1}^{n} C_{ij} \hat{u}_{ij} = \sum_{i=1}^{n} C_{ij^*} = \sum_{i=1}^{n} \min_{j \in N_i \cap I(\hat{z})} C_{ij} = f_T(\hat{z}) \quad (A11)
\]

Hence, \( f_T(\hat{z}) \) is the optimal value of Problem (9).

Proof of Theorem 3.1
Assume that \( \hat{z}_\beta \) is not weakly Pareto optimal. In this case, there exists \( z' \) feasible for Problem (10) in the main manuscript such that \( f_D(z') < f_D(\hat{z}_\beta) \) and \( f_T(z') < f_T(\hat{z}_\beta) \). Since \( w^\beta_D > 0 \), \( w^\beta_T > 0 \), and \( f_D^* \) and \( f_T^* \) are lower bounds for the individual objective functions, we have:

\[
\begin{align*}
w^\beta_D(f_D(z') - f_D^*) &< w^\beta_D(f_D(\hat{z}_\beta) - f_D^*) \\
w^\beta_T(f_T(z') - f_T^*) &< w^\beta_T(f_T(\hat{z}_\beta) - f_T^*).
\end{align*} \quad (A12)
\]

Hence, \( f_\beta(z') < f_\beta(\hat{z}_\beta) \). This contradicts the assumption that \( \hat{z}_\beta \) is a globally optimal solution for Problem (14). Hence, \( \hat{z}_\beta \) must be a weakly Pareto optimal solution.
Proof of Theorem 3.2

Define:
\[ \beta^* = \frac{(f_D(\hat{z}_D) - f_D^*)(f_T(z^*) - f_T^*)}{(f_D(\hat{z}_D) - f_D^*)(f_T(z^*) - f_T^*) + (f_T(\hat{z}_T) - f_T^*)(f_D(z^*) - f_D^*)} \]  \hspace{1cm} (A13)

Then, we have:
\[ w_D^{\beta^*} = \frac{M}{f_D(z^*) - f_D^*}, \quad w_T^{\beta^*} = \frac{M}{f_T(z^*) - f_T^*}, \]  \hspace{1cm} (A14)

with
\[ M = \frac{(f_D(z^*) - f_D^*)(f_T(z^*) - f_T^*)}{(f_D(\hat{z}_D) - f_D^*)(f_T(z^*) - f_T^*) + (f_T(\hat{z}_T) - f_T^*)(f_D(z^*) - f_D^*)}. \]  \hspace{1cm} (A15)

Assume that \( z^* \in \Theta \) is not a solution of Problem (14) for \( \beta = \beta^* \). Then, there exists \( \hat{z}_{\beta^*} \) such that
\[ f_{\beta^*}(\hat{z}_{\beta^*}) < f_{\beta^*}(z^*) \]
\[ = \max \left( w_D^{\beta^*}(f_D(z^*) - f_D^*), w_T^{\beta^*}(f_T(z^*) - f_T^*) \right) \]  \hspace{1cm} (A16)
\[ = M \]

Therefore, we have:
\[ \max \left( w_D^{\beta^*}(f_D(\hat{z}_{\beta^*}) - f_D^*), w_T^{\beta^*}(f_T(\hat{z}_{\beta^*}) - f_T^*) \right) < M. \]  \hspace{1cm} (A17)

Which yields:
\[ \frac{M}{f_D(z^*) - f_D^*}(f_D(\hat{z}_{\beta^*}) - f_D^*) < M \]  \hspace{1cm} \hspace{1cm} \hspace{1cm} (A18)
\[ \frac{M}{f_T(z^*) - f_T^*}(f_T(\hat{z}_{\beta^*}) - f_T^*) < M. \]

Hence:
\[ f_D(\hat{z}_{\beta^*}) < f_D(z^*) \]
\[ f_T(\hat{z}_{\beta^*}) < f_T(z^*). \]  \hspace{1cm} (A19)

The above inequalities contradict the assumption that \( z^* \) is a Pareto optimal solution for Problem (10). Hence, \( z^* \) must be an optimal solution of Problem (14) with \( \beta = \beta^* \). \( \square \)
A3 Pseudo-code algorithm description

Algorithm 3 Round-and-Swap Algorithm

Input: \( z^*, A_1, \ldots, A_n, G = [g_1, \ldots, g_{nn}], e \in \mathbb{R}^{np}, m, \text{MaxIter} \).

Let \( I = \{i_1, \ldots, i_{nn}\} \) be such that \( z_i^* \geq \ldots \geq z_{nn}^* \).

Define \( I_r = \text{reverse}(I) = \{i_{nn}, \ldots, i_1\} \).

Step 1: Rounding: initialize \( \hat{z} = 0 \) and \( k = 1 \).

while \( \sum_{i=1}^{nn} \hat{z}_i < m \) do

if \( \max(G\hat{z} + g_{ik} - b) \leq 0 \) then \( \hat{z}_{ik} = 1 \).

end if

\( k = k + 1 \).

end while

Step 2: Swapping: initialize \( \text{count} = 0 \).

while \( \text{count} \leq \text{MaxIter} \) do

Set \( \text{flag} = 1 \) and \( S = \{i \in I_r \mid \hat{z}_i = 1 \text{ and } 0.1 \leq z_i^* \leq 0.9\} \).

\( nS = \{i \in I \mid \hat{z}_i = 0 \text{ and } 0.1 \leq z_i^* \leq 0.9\} \).

for \( i \in S \) do

for \( j \in nS \) do

\( \text{count} = \text{count} + 1 \).

\( z_{\text{test}} = \hat{z}, z_{\text{test}}^i = 0, z_{\text{test}}^j = 1 \).

if \( \max(Gz_{\text{test}} - b) \leq 0 \) and \( f(z_{\text{test}}) \leq f(\hat{z}) \) then

\( \text{flag} = 0 \) and \( \hat{z} = z_{\text{test}} \).

end if

end for

end for

if \( \text{flag} = 1 \) then break. \# convergence reached

end if

end while

A4 L-TOWN-A

The water network model L-Town was provided by the organizers of the BattLeDIM [VET+20], and it represents a hypothetical town with approximately 10,000 inhabitants, who receive water from two reservoirs. The original network model consists of three distinct hydraulic areas, namely A, B, and C. In this work, we focus on the main network area, referred to as L-TOWN-A and shown in Figure A1a. The remaining smaller areas B and C are fed from L-TOWN-A through a pressure control valve and a pumping station, respectively. In the model of L-TOWN-A, we opportunistically modify the demands at boundary nodes with areas B and C to represent the water use in these sectors. L-TOWN-A includes 768 links, 661 nodes, and two pressure control valves installed downstream of the two reservoirs. We have grouped pipes in
Figure A1: L-TOWN-A: (a) shows locations of expert choice of existing pressure sensors and pressure control valves (PCVs); (b) pipe group.

L-TOWN-A in 6 groups based on their lengths and diameters - the spatial distribution of the groups is shown in Figure [A1b]. Similarly to BWFLnet, our formulation considers three different demand conditions, corresponding to morning peak, afternoon dip, and evening peak. The organizers have selected 29 locations for the installation of pressure sensors in L-TOWN-A, which are presented in Figure [A1a]. In addition, flow sensors are installed at the two pressure control valves. Pressure and flow sensors have been used in BattLeDIM to detect and localize leaks in L-TOWN-A. As first step, we have formulated and solved two single-objective optimization problems in L-TOWN-A to minimize $f_D$ and $f_T$, respectively. Analogously to BWFLnet, the convex heuristic has resulted in certified globally optimal or near-optimal solutions with tight lower bounds, see Figure [A2]. Then, the Chebyshev scalarization algorithm was implemented to approximate the Pareto front of the bi-objective optimization problems to jointly minimize $f_D$ and $f_T$ in L-TOWN-A, with number of sensors to be install equal to 29. Figure [A3] shows that the algorithm has successfully generated 20 potentially non-dominated solutions with a tight bound on the true Pareto front. These solutions dominate the expert choice, and represent the best compromises between sensitivity and topology based metrics.
Figure A2: Results obtained by the convex heuristic implemented to install 2 to 40 pressure sensors in L-TOWN-A.

Figure A3: Approximation of the Pareto front obtained by the Chebyshev algorithm with $N = 20$, implemented to solve the bi-objective sensor location problem of installing 29 sensors in L-TOWN-A.
A5 Reformulation of convex relaxations

First, as observed in Lemma 2.1, when \( f(z) = f_T(z) \), Problem (26) is equivalent to the following linear program with \( n_n^2 + n_n \) scalar variables:

\[
\text{minimize} \quad \sum_{i=1}^{n_n} \sum_{j=1}^{n_n} C_{ij} u_{ij}
\]

subject to

\[
\sum_{j=1}^{n_n} z_j = m \\
Gz \leq b
\]

\[
u_{ij} \leq z_j, \quad i \in \{1, \ldots, n_n\}, \quad j \in \{1, \ldots, n_n\}
\]

\[
\sum_{j=1}^{n_n} u_{ij} = 1, \quad i \in \{1, \ldots, n_n\}
\]

\[
u_{ii} = 0, \quad i \in \{1, \ldots, n_n\}
\]

\[
u_{ij} \geq 0, \quad i \in \{1, \ldots, n_n\}, \quad j \in \{1, \ldots, n_n\}
\]

\[
z \in [0, 1]^{n_n}.
\]

(A20)

In comparison, when \( f(z) = f_D(z) \), Problem (26) yields:

\[
\text{minimize} \quad - \log \det \left( X_0 + \sum_{j=1}^{n_n} z_j A_j A_j^T \right)
\]

subject to

\[
\sum_{j=1}^{n_n} z_j = m \\
Gz \leq b
\]

\[
z \in [0, 1]^{n_n},
\]

(A21)

which is equivalently re-written as a semidefinite program:

\[
\text{minimize} \quad - \sum_{k=1}^{n_r} y_k
\]

subject to

\[
(y_k, 1, W_{kk}) \in \mathcal{E}, \quad k \in \{1, \ldots, n_r\}
\]

\[
\left[ X_0 + \sum_{j=1}^{n_n} z_j A_j A_j^T \quad W \right]^{\text{diag}(W)} \succeq 0
\]

\(W\) is lower triangular

\[
\sum_{j=1}^{n_n} z_j = m \\
Gz \leq b
\]

\[
z \in [0, 1]^{n_n},
\]

(A22)
where $\text{diag}(W) \in \mathbb{R}^{n_r \times n_r}$ is the diagonal matrix with the same diagonal elements as $W$, and $\mathcal{E}$ is the exponential cone. Problem (A22) is a semidefinite program with $n_r + n_n$ scalar variables, one matrix variable, and $n_r$ exponential cone constraints. Finally, when $f(z) = f_\beta(z)$, Problem (26) is:

$$
\begin{align*}
\text{minimize} & \quad t \\
\text{subject to} & \quad f_D(z) \leq \frac{t}{w_D^\beta} + f_D^* \\
 & \quad f_T(z) \leq \frac{t}{w_T^\beta} + f_T^* \\
 & \quad \sum_{j=1}^{n_n} z_j = m \\
 & \quad Gz \leq b \\
 & \quad z \in [0, 1]^{n_n}.
\end{align*}
$$

By Lemma 2.1, Problem (A23) is equivalent to:

$$
\begin{align*}
\text{minimize} & \quad t \\
\text{subject to} & \quad - \log \det \left( X_0 + \sum_{j=1}^{n_n} z_j A_j A_j^T \right) \leq \frac{t}{w_D^\beta} + f_D^* \\
 & \quad \sum_{i=1}^{n_n} \sum_{j=1}^{n_n} C_{ij} u_{ij} \leq \frac{t}{w_T^\beta} + f_T^* \\
 & \quad u_{ij} \leq z_j, \quad i \in \{1, \ldots, n_n\}, \; j \in \{1, \ldots, n_n\} \\
 & \quad \sum_{j=1}^{n_n} u_{ij} = 1, \quad i \in \{1, \ldots, n_n\} \\
 & \quad u_{ii} = 0, \quad i \in \{1, \ldots, n_n\} \\
 & \quad u_{ij} \geq 0 \quad i \in \{1, \ldots, n_n\}, \; j \in \{1, \ldots, n_n\} \\
 & \quad \sum_{j=1}^{n_n} z_j = m \\
 & \quad Gz \leq b \\
 & \quad z \in [0, 1]^{n_n}.
\end{align*}
$$

28
which is re-written into the following semidefinite program:

\[
\begin{align*}
& \text{minimize} \quad t \\
& \text{subject to} \quad -\sum_{k=1}^{n_r} y_k \leq \frac{t}{w_D} + f_D^* \\
& \quad \quad \quad \sum_{i=1}^{n_n} \sum_{j=1}^{n_n} C_{ij} u_{ij} \leq \frac{t}{w_T} + f_T^* \\
& \quad \quad \quad (y_k, 1, W_{kk}) \in \mathcal{E}, \quad k \in \{1, \ldots, n_r\} \\
& \quad \quad \quad \begin{bmatrix} X_0 + \sum_{j=1}^{n_n} z_j A_j A_j^T & W \\ W^T & \text{diag}(W) \end{bmatrix} \succeq 0 \\
& \quad \quad \quad W \text{ is lower triangular} \\
& \quad \quad \quad u_{ij} \leq z_j, \quad i \in \{1, \ldots, n_n\}, \; j \in \{1, \ldots, n_n\} \\
& \quad \quad \quad \sum_{j=1}^{n_n} u_{ij} = 1, \quad i \in \{1, \ldots, n_n\} \\
& \quad \quad \quad u_{ii} = 0, \quad i \in \{1, \ldots, n_n\} \\
& \quad \quad \quad u_{ij} \geq 0 \quad i \in \{1, \ldots, n_n\}, \; j \in \{1, \ldots, n_n\} \\
& \quad \quad \quad \sum_{j=1}^{n_n} z_j = m \\
& \quad \quad \quad Gz \leq b \\
& \quad \quad \quad z \in [0, 1]^{n_n}.
\end{align*}
\]

**Acknowledgements**

This work was supported by EPSRC (EP/P004229/1, Dynamically Adaptive and Resilient Water Supply Networks for a Sustainable Future).

**References**

[AS16] Edo Abraham and Ivan Stoianov. Sparse Null Space Algothims for Hydraulic analysis of large scale water supply networks. *Journal of Hydraulic Engineering*, 142(3), 2016.

[AZLSW21] Zeyuan Allen-Zhu, Yuanzhi Li, Aarti Singh, and Yining Wang. Near-optimal discrete optimization for experimental design: a regret minimization approach. *Mathematical Programming*, 186(1-2):439–478, 2021.

[BBC+08] Pierre Bonami, Lorenz T. Biegler, Andrew R. Conn, Gérard Cornuéjols, I. E. Grossmann, Carl D. Laird, Jon Lee, Andrea Lodi, François Margot, Nicolas
Sawaya, and Andreas Wächter. An algorithmic framework for convex mixed integer nonlinear programs. Discrete Optimization, 5(2):186–204, 2008.

[BDL+15] Cristiana Bragalli, Claudia D’Ambrosio, Jon Lee, Andrea Lodi, and Paolo Toth. Optimizing the Design of Water Distribution Networks Using Mathematical Optimization. In Case Studies in Operations Research, chapter 9, pages 183–198. Springer New York, 2015.

[BEKS17] Jeff Bezanson, Alan Edelman, Stefan Karpinski, and Viral B Shah. Julia: A fresh approach to numerical computing. SIAM Review, 59(1):65–98, 2017.

[BGL05] Michele Benzi, Gene H. Golub, and Jörg Liesen. Numerical solution of saddle point problems. Acta Numerica, 14, 2005.

[BKSA09] Kourosh Behzadian, Zoran Kapelan, Dragan Savic, and Abdollah Ardeshir. Stochastic sampling design using a multi-objective genetic algorithm and adaptive neural networks. Environmental Modelling and Software, 24(4):530–541, 2009.

[BPS21] Caroline Blocher, Filippo Pecci, and Ivan Stoianov. Prior Assumptions for Leak Localisation in Water Distribution Networks with Uncertainties. Water Resources Management, (0123456789), 2021.

[BU98] Cheryl A. Bush and James G. Uber. Sampling Design Methods for Water Distribution Model Calibration. Journal of Water Resources Planning and Management, 124(6):334–344, 1998.

[CW20] Richard L. Church and Shaohua Wang. Solving the p-median problem on regular and lattice networks. Computers and Operations Research, 123:105057, 2020.

[DENR20] Marianna De Santis, Gabriele Eichfelder, Julia Niebling, and Stefan Rocktaschel. Solving Multiobjective Mixed Integer Convex Optimization Problems. Siam Journal on Optimization, 30(4):3122–3145, 2020.

[DGP20] Marianna De Santis, Giorgio Grani, and Laura Palagi. Branching with hyperplanes in the criterion space: The frontier partitioner algorithm for biobjective integer programming. European Journal of Operational Research, 283(1):57–69, 2020.

[DHL17] Iain Dunning, Joey Huchette, and Miles Lubin. Jump: A modeling language for mathematical optimization. SIAM Review, 59(2):295–320, 2017.

[DPAM02] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan. A fast and elitist multiobjective genetic algorithm: Nsga-ii. IEEE Transactions on Evolutionary Computation, 6(2):182–197, 2002.
[DWS00] W. B. F. De Schaetzen, G. A. Walters, and Dragan Savic. Optimal sampling design for model calibration using shortest path, genetic and entropy algorithms. Urban Water, 2(2):141–152, 2000.

[ESD+14] Sylvan Elhay, Angus R Simpson, Jochen W. Deuerlein, Bradley Alexander, and Wil Schilders. A Reformulated Co-tree Flows Method competitive with the global Gradient Algorithm for solving the water distribution system equations. Journal of Water Resources Planning and Management, 140(12):04014040, 2014.

[FGGV15] Nicola Fontana, Maurizio Giugni, Stefano Gliozzi, and Marcello Vitaletti. Shortest path criterion for sampling design of water distribution networks. Urban Water Journal, 12(2):154–164, 2015.

[GCG21] Michael Garstka, Mark Cannon, and Paul Goulart. COSMO: A Conic Operator Splitting Method for Convex Conic Problems. Journal of Optimization Theory and Applications, 190(3):779–810, 2021.

[Gur21] Gurobi Optimization. Gurobi Optimizer 9.1 Reference Manual, 2021.

[JB09] Siddharth Joshi and Stephen Boyd. Sensor selection via convex optimization. IEEE Transactions on Signal Processing, 57(2):451–462, 2009.

[JLCK15] Donghwi Jung, Kevin Lansey, Younghwan Choi, and Joong Hoon Kim. Robustness-based optimal pump design and scheduling for water distribution systems. Journal of Hydroinformatics, page jh2015091, 2015.

[KGW12] Vassilis Kekatos, Georgios B. Giannakis, and Bruce Wollenberg. Optimal placement of phasor measurement units via convex relaxation. IEEE Transactions on Power Systems, 27(3):1521–1530, 2012.

[KSW03] Zoran Kapelan, Dragan Savic, and Godfrey A. Walters. Multiobjective Sampling Design for Water Distribution Model Calibration. Journal of Water Resources Planning and Management, 129(6):466–479, 2003.

[KSW05] Zoran Kapelan, Dragan Savic, and Godfrey A. Walters. Optimal Sampling Design Methodologies for Water Distribution Model Calibration. Journal of Hydraulic Engineering, 131(3):190–200, 2005.

[LB91] Kevin Lansey and Chuda Basnet. Parameter estimation for water distribution networks. Journal of Water Resources Planning and Management, 117(1):126–144, 1991.

[LJW99] Bruce E Larock, Roland W Jeppson, and Gary Z Watters. Hydraulics of Pipeline Systems. CRC press, 1999.
[MAPS16] Ruben Menke, Edo Abraham, Panos Parpas, and Ivan Stoianov. Demonstrating demand response from water distribution system through pump scheduling. *Applied Energy*, 170:377–387, 2016.

[MATL02] Keshaw N. Mallick, Iftekhar Ahmed, Kevin S. Tickle, and Kevin Lansey. Determining Pipe Groupings for Water Distribution Networks. *Journal of Water Resources Planning and Management*, 128(April):130–139, 2002.

[Mie98] Kaisa Miettinen. *Nonlinear Multiobjective Optimization*, volume 12. Springer US, 1998.

[MOS22] MOSEK ApS. The mosek optimization api. version 9.3.20, 2022.

[PSO21] Filippo Pecci, Ivan Stoianov, and Avi Ostfeld. Case Studies for Optimal Placement and Control of Valves and Chlorine Boosters in Water Networks. *Mendeley Data*, 2021.

[Puk06] Friedrich Pukelsheim. *Optimal Design of Experiments*. SIAM, 2006.

[Ree06] J Reese. Solution methods for the p-median problem An annotated bibliography. *Networks*, 48(3):125–142, 2006.

[SDG+22] David B. Steffelbauer, Jochen Deuerlein, Denis Gilbert, Edo Abraham, and Olivier Piller. Pressure-Leak Duality for Leak Detection and Localization in Water Distribution Systems. *Journal of Water Resources Planning and Management*, 148(3):1–13, 2022.

[UPS21] Aly-joy Ulusoy, Filippo Pecci, and Ivan Stoianov. Bi-objective design-for-control of water distribution networks with global bounds. *Optimization and Engineering*, 2021.

[UV21] Anton V. Ushakov and Igor Vasilyev. Near-optimal large-scale k-medoids clustering. *Information Sciences*, 545:344–362, 2021.

[VET+20] Stelios G. Vrachimis, Demetrios G. Eliades, Riccardo Taormina, Avi Ostfeld, Zoran Kapelan, Shuming Liu, Marios S. Kyriakou, Pavlos Pavlou, Mengning Qiu, and Marios Polycarpou. Dataset of BattLeDIM: Battle of the Leakage Detection and Isolation Methods, sep 2020.

[Wal83] Thomas M Walski. Technique for Calibrating Network Models. *Journal of Water Resources Planning and Management*, 109(4):360–372, 1983.

[WAPS15] Robert Wright, Edo Abraham, Panos Parpas, and Ivan Stoianov. Control of water distribution networks with dynamic DMA topology using strictly feasible sequential convex programming. *Water Resources Research*, 51(12):9925–9941, 2015.
[WB06] Andreas Waechter and Lorenz T. Biegler. On the Implementation of a Primal-Dual Interior Point Filter Line Search Algorithm for Large-Scale Nonlinear Programming. *Mathematical Programming*, 106(1):25–57, 2006.

[WPS20] Alexander Waldron, Filippo Pecci, and Ivan Stoianov. Regularisation of an inverse problem for parameter estimation in water distribution networks. *Journal of Water Resources Planning and Management*, 146(9):04020076, 2020.

[WPS21] A. Waldron, F. Pecci, and I. Stoianov. Closure to “Regularization of an Inverse Problem for Parameter Estimation in Water Distribution Systems” by Alexander Waldron, Filippo Pecci, and Ivan Stoianov. *Journal of Water Resources Planning and Management*, 147(12), 2021.

[YA21] Jing Yu and Mihai Anitescu. Multidimensional sum-up rounding for integer programming in optimal experimental design. *Mathematical Programming*, 185(1-2):37–76, 2021.

[ZXX+18] Xiao Zhou, Weirong Xu, Kunlun Xin, Hexiang Yan, and Tao Tao. Self-Adaptive Calibration of Real-Time Demand and Roughness of Water Distribution Systems. *Water Resources Research*, 54(8):5536–5550, 2018.