Highlights

Information-Theoretic Sensor Placement for Large Sewer Networks

George Crowley, Simon Tait, George Panoutsos, Vanessa Speight, Iñaki Esnaola

- A computationally scalable sensor placement algorithm using mutual information as the performance measure is proposed.
- The proposed algorithm is validated by estimating states at unmonitored locations from simulated hydrodynamic data.
- Mutual information is an appropriate performance measure to design sensor placement procedures for state estimation.
- The proposed sensor placement and estimation framework is viable for the design and monitoring of large networks.

Information-Theoretic Sensor Placement for Large Sewer Networks

George Crowley[a,*], Simon Tait[b], George Panoutsos[a], Vanessa Speight[b], Iñaki Esnaola[a,c]

[a] *Department of Automatic Control and Systems Engineering, The University of Sheffield, England*
[b] *Department of Civil and Structural Engineering, The University of Sheffield, England*
[c] *Department of Electrical Engineering, Princeton University, USA*

Abstract

Utility operators face a challenging task in managing wastewater networks to proactively enhance network monitoring. To address this issue, this paper develops a framework for optimized placing of sensors in sewer networks with the aim of maximizing the information obtained about the state of the network. To that end, mutual information is proposed as a measure of the evidence acquired about the state of the network by the placed sensors. The problem formulation leverages a stochastic description of the network states to analytically characterize the mutual information in the system and pose the sensor placement problem. To circumvent the combinatorial problem that arises in the placement configurations, we propose a new algorithm coined the one-step modified greedy algorithm, which employs the greedy heuristic for all possible initial sensor placements. This algorithm enables further exploration of solutions outside the initial greedy solution within a computationally tractable time. The algorithm is applied to two real sewer networks, the first is a sewer network in the South of England with 479 nodes and 567 links, and the second is the sewer network in Bellinge, a village in Denmark that contains 1020 nodes and 1015 links. Sensor placements from the modified greedy algorithm are validated by comparing their performance in estimating unmonitored locations against other heuristic placements using linear and neural network models. Results show the one-step modified greedy placements outperform others in most cases and tend to cluster sensors for efficiently monitoring parts of the network. The proposed framework and modified greedy algorithm provide wastewater utility operators with a sensor placement method that enables them to design the data acquisition and monitoring infrastructure for large networks.

Keywords: Sensor placement, Sensor selection, Mutual information, Sewer flow monitoring, Sewer level monitoring, Network hydraulic performance

1. Introduction

Within the last decade, installations of sewer level monitors (SLM) and volumetric sewer flow monitors (SFM) have been increasingly deployed into sewer networks for the purpose of monitoring network hydraulic performance and also interpreting flow patterns to locate defects, such as blockages. Currently, the locations chosen for SLM and SFM are often selected by an employee of the managing water utility with expert knowledge according to the purpose of installation, i.e. flood risk estimation or blockage detection. Little literature currently exists for objective sensor placement for SLM and/or SFM for estimation/prediction and forecasting purposes such as sewer flooding in large networks. In the age of smart technologies, the approach taken by wastewater utilities has been to deploy ever-increasing numbers of sensors into their networks, however, poor sensor placements result in an insufficient acquisition of useful information about the state of the network. This work is the first to pose the problem of sensor placement in sewer networks for the task of network monitoring as a mutual information maximization problem.

The outbreak of Industry 4.0 has resulted in significant investment and innovation from industry in general for the use of sensor systems and tech-

*Corresponding author at: Department of Automatic Control and Systems Engineering, The University of Sheffield S1 3JD, England.
Email address: gcrowley1@sheffield.ac.uk (George Crowley)

nologies (Javaid et al., 2021). Having access to an integrated wireless sensor network (WSN) that seamlessly migrates real-time data to the cloud can benefit wastewater network operators and local authorities by being able to monitor their networks in near real-time for purposes such as proactive blockage and other anomaly detection (Faris et al., 2024; Rosin et al., 2022; Sumer Derya et al., 2007), sewer flooding (internal and external), pollutants and disease monitoring (Nourinejad et al., 2021; Banik et al., 2015) and general long-term network performance (Ashley and Hopkinson, 2002). The recent mass scale of sensor installations in wastewater networks in the UK seems to have partially stemmed from the 2019 price review conducted by OFWAT (the UK economic regulator for the UK's water and sewerage operators) for asset management plan (AMP) 7 (01 April 2020 - 31 March 2025). In the reporting guidance for internal sewer flooding under key principles (OFWAT, 2017), it states: *'There is an assumption that there will be continued improvement by all companies in the short and medium term through innovation, new technology, data quality improvements'*, and as such the cost of installing and maintaining WSNs is a continuing expenditure.

A summary of sensor placement problems considered in wastewater networks is presented below, followed by typical methods for formulating and solving sensor placement problems. Sensor placement literature in wastewater networks is originally concerned with the calibration of deterministic hydrodynamic models and water quality monitoring. Calibrating hydrodynamical models depends on data generated by sensors placed in the network. The process of deciding sensible locations to minimize the number of sensors needed for model calibration is first considered in (Clemens, 2002) and further studied in (Vonach et al., 2018). The approach in (Clemens, 2002) computes the singular value decompositions (SVD) for the Jacobian matrix of errors between model values and measuring data with respect to the model parameters for each possible fixed combination of sensor placements. The Jacobian matrix with the largest minimum singular value among the sensor placements is selected, and the resulting locations of sensor placement are used for model calibration. Moreover, this method uses an exhaustive SVD search of all possible placement combinations, which is computationally intractable as the number of locations and amount of sensors selected for placement increases. An iterative approach to computing the calibrated model is explored in (Vonach et al., 2018), which uses the Nash-Sutcliffe-Efficiency (NSE) statistic (Nash and Sutcliffe, 1970) based on water level times series data for systematically sampled sensor placements. Therein, the authors point out that not considering different initial conditions for the model is a problem and that any solution found for model calibration will only be locally optimal. The aforementioned articles highlight the importance of sensor placement procedures for the appropriate calibration of hydrodynamical models.

In water quality monitoring, the introduction of the SARS-CoV-2 virus has received attention from researchers interested in locating areas where the virus is currently present in wastewater networks for epidemiological modeling and is posed as a sensor placement problem. The *'SARS-CoV-2 sewage surveillance'* problem is considered in (Nourinejad et al., 2021), (Larson et al., 2020), and (Calle et al., 2021) where sensor placement problems are formed as locations to conduct genetic-remnant tests. The work conducted in (Nourinejad et al., 2021) is an extension of (Larson et al., 2020), where they pose the sensor placement problem as a mixed integer non-linear programming problem solely based on the network topology, which seeks to minimize the number of sampled manholes required to find the source manhole of the SARS-CoV-2 virus. The work in (Calle et al., 2021) follows similar methods to the above studies, but the number of sensors considered is small (< 10), and therefore, it is difficult to validate the contributions for larger network systems. Another widely considered problem in wastewater monitoring is that of detecting dangerous non-point pollutant sources, e.g. herbicides and fertilizers, pathogens, road salt, and sediment from run-off, which can all impact the performance of the sewer network and the sewage treatment works. In (Banik et al., 2015, 2017b) and (Kang et al., 2013), data simulated from the Storm Water Management Model (SWMM) (Rossman, 2010) tool is used to determine sensor placements for the purpose of water quality monitoring. (Banik et al., 2015) considers a multiobjective optimization problem of maximizing joint entropy and minimizing the total correlation between the different network nodes, which uses concentration data generated from different contamination scenarios. The multi-objective optimization problem obtains a Pareto front of so-

lutions by using the NSGA-II algorithm (K. Deb et al., 2002) but the authors note that any placement chosen from the Pareto front needs further investigation and is not straightforward to assess its validity. (Banik et al., 2017b) extends his previous work by comparing different problem formulations for sensor placements in sewer systems for contaminant detection with the method proposed in (Banik et al., 2015). The authors compare formulations from a mixture of multiobjective information theory measures, detection time–reliability procedures, and single objective-based versions of all objectives considered in the multiobjective case. The NSGA-II algorithm is used to solve the multiobjective optimization problems, whilst a greedy method is used to solve the single optimization problems. Using their performance index, they showed that the solutions obtained using the greedy algorithm in the single objective cases were similar to those obtained in the Pareto front of the best-performing multiobjective optimization problem solved by the NSGA-II algorithm. The work in (Kang et al., 2013) uses analysis of variance (ANOVA) on different contaminant scenarios to choose a node for sensor placement but validates the proposed methodology on a small network of 10 nodes and 9 links. (Yazdi, 2018) also uses the SWMM tool for designing sensor placements in sewer networks for water quality purposes. The authors propose the single objective methodology of maximizing the entropy of water quality time series data, using Differential Evolution (Storn and Price, 1997), a genetic algorithm to solve the optimization problem. (Wang et al., 2023) considers the sensor placement problem by taking into account water quality and hydraulic properties of drainage networks for routine monitoring by developing a re-clustering methodology. A K-means clustering algorithm is implemented using water level and pollutant concentration data of all nodes, generated by a SWMM model. The average closeness centrality and average node inflow indexes are introduced as measures that incorporate decision-maker preferences into the sensor placement design. The placements obtained by the K-means and reindexing are then validated using the information methods used in (Banik et al., 2015). We have established that the sensor placement problem is also important in the detection of various contamination scenarios or sampling for the SARS-CoV-2 virus but the current state of the art is fragmented and techniques tend to be bespoke to the specific settings. In more recent years, sensor placement problems in sewer networks have gained some traction for sewer flooding detection. (Fattoruso et al., 2015) considers the problem of the optimal sampling design for sensor placement, which follows a similar methodology to model calibration. Therein, a multiobjective problem that maximizes the hydrological model calibration accuracy and fixes the number of sensors within some fixed interval is posed. In (Li, 2021) the sensor placement problem for flood forecasting under uncertainty from future rainfall events is considered. The authors simulate historical and future periods of continuous rainfall-runoff data using the SWMM tool and then combines the unsupervised machine learning technique of agglomerative clustering and analysis of variance (ANOVA). The clustering technique specifies the number of clusters in which a sensor is placed, then the ANOVA technique is used to determine which node within the cluster the sensor should be placed at. Both studies present different ideas for sensor placement for sewer flooding, but both studies showcase case studies with only simple and relatively small networks.

The sensor placement problem is typically formulated based on having access to a combination of graph data (such as GIS shape files) and simulated data. Based on the problem formulation, an operationally meaningful cost function is selected to optimize. An example of a purely graph-based problem formulation in addition to those already mentioned includes (Simone et al., 2023), in which a graph-theoretic back-tracking procedure is developed for sensor placement, with the end purpose of contaminant and pathogen detection in wastewater networks. The authors in (Ogie et al., 2017) propose a sensor placement problem formulation purely in terms of geospatial features, in which a multiobjective function based on infrastructure density (hydrological infrastructure components based up and downstream of each node), number of upstream nodes, classification of waterways and geographical distance between nodes is formulated. The authors use the NSGA-II algorithm to solve the multiobjective problem, and apply their methodology to Jakarta, where they consider placing 4 water level sensors, and then consider placing an additional 10 sensors.

Approaches that use simulated data often include cost function definitions based on information-theoretic measures such as entropy (Banik et al., 2015), total correlation (Banik et al., 2017a),

and mutual information (Krause et al., 2008b). Information-theoretic measures were first introduced in the seminal work by Shannon (C. E. Shannon, 1948) that establishes the foundations of information theory. The field of information theory is concerned with the study of information processes with application to a wide range of scientific domains. The main workhorse to aid in the analysis of information processes are the information measures, defined as functionals of the probability distributions of the underlying stochastic processes. In particular, mutual information provides a quantitative description of the amount of information that is shared by two stochastic processes. Solving sensor placement problems for complex networks is generally very difficult, due to the combinatorial nature of the problem. For large networks consisting of $n \in \mathbb{N}$ nodes and selecting $k < n,\ k \in \mathbb{N}$ locations for sensor placement, the number of possible combinations grows exponentially as n grows, and is typically NP-hard for non-linear costs with one example being shown in (Ko et al., 1995). The largest network known to the authors, to have considered the sensor placement problem comes from (Krause et al., 2008a) where they considered the contamination detection problem on a *water distribution system* from an actual metropolitan area network which has 21,000 nodes and simulated placing 30 sensors. The works surveyed showcase a variety of different problem formulations and proposed solutions to find sensor placement procedures. It is worth noting that most of the contributions in this area tend to consider a small number of sensors to be placed in the network, despite some notable exceptions such as (Krause et al., 2008a). However, the total possible sensor placements for 30 sensors in a network of 21,000 nodes has the approximately same number of combinations as a network of 1000 nodes placing 60 sensors, and so scalability in selecting the number of sensors should be factored into proposed solutions especially when considering large networks ($>$ 1000 nodes). Water companies are facing a dilemma about requiring more information about the performance of their networks (CSO spills, flooding incidents), whilst on the opposite front, facing economic and workforce issues as a result of government and regulator policy that is starting to restrict the deployment of large numbers of even 'low-cost' sensors.

1.1. Outline of the Paper

In Section 2, we develop the sensor placement method by introducing our system and sensing model. It further introduces our performance measure and problem formulation and finishes with our proposed solution to solving the sensor placement problem. We proceed in Section 3 by outlining the methods we will use to validate the proposed sensor placement algorithm. This includes Section 3.1 which presents our estimation framework, and Section 3.2 which explains our loss function of choice for the estimation framework. Section 3.3 describes other sensor placement heuristics we use to compare the estimation results obtained using Section 3.1. Sections 4.1 and 4.2 contain real case studies that demonstrate our sensor placement and estimation results. Moreover, in Section 4.2 we give a more detailed analysis of estimation results and compare both flow and water depth sensor placement results. Finally, in Section 5, we conclude the results of both case studies with a discussion to examine the useability of the proposed method in other sewer collection networks.

2. Sensor Placement Method

2.1. System Model

We model a wastewater network as a graph where the nodes describe the elements of the wastewater system e.g. manholes and pumps, and the edges describe the pipes connecting the different elements. Specifically, the graph is characterized by the set of nodes $\mathcal{V} = \{1, 2, \ldots, n\}$ with $n \in \mathbb{N}$, where each node corresponds to a system asset/element, and the set of edges as $\mathcal{E} = \{(i, j) \in \mathcal{V} \times \mathcal{V} :$ node i is connected to node $j\}$, where each edge represents a pipe in the network. Jointly, the set of edges $\mathcal{E}$ and the set of nodes $\mathcal{V}$ define an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. Without loss of generality, we assume the state of node $i \in \mathcal{V}$ is determined by the vector $\mathbf{x}_i \in \mathbb{R}^l$, where $l \in \mathbb{N}$ denotes the number of physical magnitudes describing the state of the system element, e.g. water depth and volumetric flow. In this paper, we focus initially on volumetric flow, and therefore, the state $x_i \in \mathbb{R}$ fully describes the state of the element $i \in \mathcal{V}$. We note that the case study in Section 4.2 also considers sensor placement for water depth as well as volumetric flow.

In the remainder of this section, we define the mathematical objects that enable us to model the

Table of Notation

Symbol	Description
$\emptyset$	The empty set
X_i	The state at node i
$N(\mu, \sigma^2)$	The Gaussian distribution with mean μ and variance σ^2
f_{X_i}	The probability density function of the random variable X_i
$f_{X,Y}$	The joint probability density function of the random variables X and Y
$\mathbf{I}_k$	The $k \times k$ identity matrix
$\mathbb{R}_+$	The set of all positive real numbers > 0
S^n_{++}	The set of all $n \times n$ matrices that are positive definite
$\|\mathbf{X}\|_F$	The Frobenius norm of a matrix $\mathbf{X}$, calculated as $\sqrt{\sum_i \sum_j \mathbf{X}^2_{i,j}}$

wastewater system and formulate the sensor placement problem. The following definition describes the selection of sensor placements in the wastewater system in terms of the node selection problem of the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$. In this setting, we denote the elements of the system that are monitored by the set $\mathcal{A} \subseteq \mathcal{V}$. The sensor placement problem is equivalent to identifying the set of nodes in which sensors are placed with the aim of aiding the operator in monitoring the state of the system.

Definition 1. *Consider the graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ describing the wastewater system, the set of all k-configurations with k sensor placements is given by the set*

$$\mathcal{M}_k := \{\mathcal{A} \subseteq \mathcal{V} : |\mathcal{A}| = k\}, \tag{1}$$

where $|\cdot|$ denotes the cardinality of the set.

Remark 1. *The cardinality of the set $\mathcal{M}_k$ for a graph with n nodes is given by*

$$|\mathcal{M}_k| = \frac{n!}{k!(n-k)!}. \tag{2}$$

Definition 1 and Remark 1 unveil the difficulty in sensor placement problems when we consider networks containing upwards of thousands of nodes, i.e. due to the exponential growth of combinations as n grows, the search space becomes computationally expensive to explore by exhaustive search methods.

2.1.1. Random Model for the States

To address the complexity and underlying uncertainty in the wastewater network, we model the states in the system as a random process. In doing so, we aim to capture any partial model information, dynamical behavior, and stochasticity of the physical magnitudes describing the state of the system elements. That being the case, the states as random variables are denoted by $X_i \sim P_{X_i}$ for $i \in \mathcal{V}$. The spatial dependence between elements of the system is described by the n-dimensional random vector where the joint distribution $P_{X^n} := P_{X_1,X_2,\ldots,X_n}$ captures the spatial dependencies between different nodes. While the temporal dependence of the states is an important aspect of a wastewater system, in this paper we adopt the view that spatial dependencies between nodes are invariant to time shifts. Note that this does not imply that the statistics of states are stationary, but rather the dependencies describing different spatial locations are invariant to time. More specifically, we will assume that the covariances between states (i.e. X_i and X_j) over different time shifts (i.e. 1 month or 6 months) are the same *on average* when we estimate the covariances of the states using time series data. This in turn implies that we model the covariance matrix which describes dependencies between the states as independent of time. This assumption is put into place explicitly in Assumption A3 in Section 2.3.2.

2.1.2. Additive Noise Sensing Model

Sensors introduce noise in the measured physical magnitude, and therefore, this needs to be incorporated into the sensing model. Additionally, the additive noise model enables us to account for other sources of uncertainty in the sensing process, e.g. dispersions arising from diverse installation settings

or unknown operational conditions impinging on the performance of the sensor. We adopt an additive noise model approach and include two assumptions on the noise for the sensing model.

Assumption 1 (A1): sensor readings are subject to additive white Gaussian noise (AWGN), denoted as Z_i at node $i \in \mathcal{V}$.

Assumption 2 (A2): the AWGN process for each sensor is independent and identically distributed (i.i.d.) with mean 0 and variance $\sigma^2 \in \mathbb{R}_+$. Formally, for all $i \in \mathcal{V}$ we have that $Z_i \sim N(0, \sigma^2)$.

As a result of these assumptions and the fact that the state variables are modeled as random variables, the observations obtained by the sensor are also random variables. Let us denote the observation at node $i \in \mathcal{V}$ as $Y_i \sim P_{Y_i}$. Then, it follows from Assumption 1 and Assumption 2 that the observation is given by

$$Y_i = X_i + Z_i. \tag{3}$$

Moreover, assuming we have k sensors in the network selected amongst n nodes, then the observation vector Y^k is defined as

$$Y^k := (Y_{i_1}, \dots, Y_{i_k})^\mathsf{T}, \tag{4}$$

where the subscript i_j denotes the j-th selected sensor. For ease of algebraic manipulation, it is convenient to describe (4) in matrix form, such that we can express Y^k for any set $\mathcal{A} \in \mathcal{M}_k$ of sensor placements in terms of an observation matrix $\mathbf{H}$, defined in the following.

Definition 2. *The set of linear observation matrices $\mathcal{H}_k$ is described by*

$$\mathcal{H}_k := \Big\{ \mathbf{H} \in \{0,1\}^{k \times n} : \mathbf{H} = \left(\mathbf{e}_{i_1}^\mathsf{T}, \mathbf{e}_{i_2}^\mathsf{T}, \dots, \mathbf{e}_{i_k}^\mathsf{T}\right)^\mathsf{T} \text{ with } i_j \in \mathcal{A} \setminus \{i_1, \dots, i_{j-1}\} \text{ for } j = 1, \dots, k \Big\}, \tag{5}$$

where $\mathbf{e}_i \in \{0,1\}^n$ is the i-th column basis vector, i.e. 1 in the i-th position and 0 otherwise.

Combining Definition 2 with (4) yields the following observation model:

$$Y^k := \mathbf{H}X^n + Z^k, \quad \text{for all } \mathbf{H} \in \mathcal{H}_k. \tag{6}$$

2.2. Performance Measure for Sensor Placement

The objective of the operator for the sensor placement is to guarantee that operationally significant data is acquired with the purpose of use for real-time monitoring, flood risk prediction, blockage detection, and state estimation procedures, among others. The data describing the states feeds into multiple services and functionalities with often different objectives, captured by different performance measures, e.g. probability of detection for blockage detection. Naturally, designing the sensor placement procedure according to different performance measures yields different placement strategies that might not provide guarantees for all the different procedures in the system. Given the economic and implementation constraints that sensor placements entail, it is necessary to devise sensor placement procedures that provide a wide range of performance guarantees across all services and functionalities that the operator needs to implement with the data produced by the sensors in the system. As a result of the scale and complexity of typical wastewater networks, it is imperative to design sensor placement procedures that are computationally implementable, make use of as few sensing resources as possible, and provide general performance guarantees.

Mutual information is an information-theoretic measure that provides an operational definition of the amount of *evidence* contained in the data. Specifically, it quantifies the information that is shared between two different random processes, and in doing so, establishes a quantitative framework to evaluate the utility of data in fundamental terms, i.e. without targeting specific performance measures. By doing so, we effectively decouple the assessment of data utility from the specific service or functionality. This, in turn, enables us to pose the sensor placement problem in fundamental terms and to provide sensor placement guidelines that target the amount of information captured by the sensors, irrespective of the use that is given to that data. Given this, we define the cost function for the sensor placement problem as the measure of mutual information between the observations and the state variables.

Definition 3. *The mutual information (Cover, 2005) between two continuous random variables X and Y, denoted as $I(X;Y)$, with joint probability density function $f_{X,Y}(x,y)$ is given by*

$$I(X;Y) := \int_{\mathbb{R}^2} f_{X,Y}(x,y) \log\left(\frac{f_{X,Y}(x,y)}{f_X(x) f_Y(y)}\right) dxdy.$$

Mutual information is a non-negative and non-linear measure of dependence between the two random variables X and Y. Note that it is a symmetric measure, i.e. $I(X;Y) = I(Y;X)$. Equipped with this measure, we now formulate the sensor placement problem as a mutual information maximization problem below.

2.3. Problem Formulation

2.3.1. Mutual Information Maximisation

The sensor placement problem consists of obtaining the set of sensor placement indices $\mathcal{I}_k \in \mathcal{M}_k$ such that

$$\mathcal{I} := \underset{\mathcal{I}_k \in \mathcal{M}_k}{\arg\max}\ I(X^n; Y^k). \tag{7}$$

Combining (6) with (7), we can re-write (7) in terms of the observation matrix such that

$$\mathbf{H}_k^* := \underset{\mathbf{H} \in \mathcal{H}_k}{\arg\max}\ I(X^n; \mathbf{H}X^n + Z^k), \tag{8}$$

where $\mathbf{H}_k^*$ is the observation matrix describing the optimal sensor placement for k sensors.

2.3.2. Gaussian Model for State Variables

Mutual information is a functional of the joint probability distribution and the marginals of the random variables involved. As a result, computing it requires knowledge of the probability density functions, which in practical settings is not known. To circumvent this issue, we need to assume a probability distribution for the underlying random processes governing the state variables X^n. Essentially, this boils down to modeling the random characteristics of the state variables by choosing an appropriate distribution to describe them. In the following, we adopt a Gaussian model for all the state variables based on a maximum entropy principle interpretation of the inductive bias introduced by the model (Cover, 2005, Ch 12). By assigning a multivariate Gaussian distribution to a joint probability distribution of the random variables describing the state variables, we are introducing the least amount of bias in our modeling choice. In fact, the multivariate Gaussian distribution is the maximally non-committal distribution that satisfies the second-order moment constraint that the data exhibits.

Assumption 3 (A3): the probability distribution of the state variables satisfies $X^n \sim N_n(\boldsymbol{\mu}, \boldsymbol{\Sigma})$, where $\boldsymbol{\mu} \in \mathbb{R}^n$ and $\boldsymbol{\Sigma} \in S_{++}^n$.

In the following, we describe the mutual information between the states and the observations for the case in which the states are Gaussian distributed.

Theorem 1. *Under Assumption 3, it holds that*

$$I(X^n; \mathbf{H}X^n + Z^k) = \frac{1}{2} \log\left(\frac{1}{\sigma^{2k}} \det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\mathsf{T} + \sigma^2 \mathbf{I}_k\right)\right), \tag{9}$$

where $\det(\cdot)$ *denotes the determinant of a square matrix.*

Proof. Appendix A. □

As a result of Theorem 1 and the aforementioned assumptions, the optimization problem (8) can be reformulated as

$$\mathbf{H}_k^* = \underset{\mathbf{H} \in \mathcal{H}_k}{\arg\max} \frac{1}{2} \log\left(\frac{1}{\sigma^{2k}} \det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\mathsf{T} + \sigma^2 \mathbf{I}_k\right)\right). \tag{10}$$

2.4. Proposed Sensor Placement Solution

The computational complexity of optimization problems similar to (10) are NP-hard (Ko et al., 1995), as is the case with most nonlinear cost functions associated with the sensor placement problems. Hence, proposing a method to solve (10) depends on the size of the network, but in general it is computationally intractable. For larger networks, typically two approaches are considered; exact and heuristic. Exact approaches for this type of problem generally include some form of Branch and Bound algorithm (Lee and Fampa, 2022), where the algorithm splits the problem into smaller sub-problems and uses computable upper and lower bounds of the optimization problem to remove sub-problems that perform worse than the current optimal solution. The bounds are problem-dependent, and without them, the algorithm becomes an exhaustive search of the state space. Hence, Branch and Bound can be computationally expensive and do not guarantee convergence to an exact solution. A common heuristic for solving non-linear discrete optimization problems is the greedy algorithm (Jungnickel, 2013) and its variations (Taillard, 2023), due to its simplicity of implementation and cost-benefit. The greedy algorithm uses the heuristic of taking the best solution available at every step or stage, and sequentially repeating this step until some predefined criteria or condition is met. The greedy

Algorithm 1 One-step modified greedy algorithm

Input : $n = |\mathcal{V}|$ from (1);
k from (1);
σ^2 from (**A3**);
$\mathbf{\Sigma}$ from (9).

Output: $\mathbf{H}_j^\dagger$ from (9) for all $j \in [1, k]$.

for $i = 1 : n$ **do**
  $\mathbf{H}_1^\dagger[i] \leftarrow (\mathbf{e}_i)^\mathsf{T}$ from (2)
  for $j = 2 : k$ **do**
    $\mathcal{A}_j \leftarrow \emptyset$
    for $v \in \mathcal{V} \setminus \mathcal{I}_k$ **do**
      $\mathbf{H}_j \leftarrow \begin{pmatrix} \mathbf{H}_{j-1}^\dagger[i] \\ \mathbf{e}_{iv}^\mathsf{T} \end{pmatrix}$
      $\mathcal{A}_j \leftarrow \mathcal{A}_j \cup \mathbf{H}_j$
    end
    $\mathbf{H}_j^\dagger[i] \leftarrow \underset{\mathbf{H}_j \subseteq \mathcal{A}_j}{\arg\max} \ \frac{1}{2} \log \left(\frac{1}{\sigma^{2j}} \det \left(\mathbf{H}_j \mathbf{\Sigma} \mathbf{H}_j^\mathsf{T} + \sigma^2 \mathbf{I}_j \right) \right)$
  end
end
$\mathbf{H}_k^\dagger \leftarrow \underset{i \in [1,n]}{\max} \mathbf{H}_k^\dagger[i]$

heuristic only considers finding locally optimal solutions (at each step), and will not find the optimal global solution in more complex optimization problems.

Wastewater networks pose an interesting challenge to the sensor placement problem due to the size of the networks, often comprising of several thousand nodes. The size further exacerbates the difficulty posed by the non-linearity of the determinant operator in (10). Alternatively, to tackle the optimization problem in (10), we propose a modified version of the greedy algorithm, which we coin the *one-step modified greedy algorithm*.

The modified greedy algorithm runs a standard greedy selection procedure for each available node as the initial solution to the greedy heuristic and compares the performance attained by each of the initialization steps. The selection of all the initialization choices that yield maximum mutual information is the selected sensor placement.

This approach has several advantages:

- The standard greedy heuristic does not consider solutions outside of its initial solution set. By allowing the extension from the modified greedy algorithm, we allow for the exploration of other solutions that do not include the true initial solution to the greedy heuristic.
- The algorithm can be tweaked to include a set of fixed selected sensors in the initial solution which gives flexibility to design sensor extensions in existing sensor networks.
- Flexibility in implementing other constraints not currently considered (i.e. distance constraints).
- The mutual information can be computed for an arbitrary number of sensors selected in the network, which provides a framework for how many sensors are needed, assuming the sensing requirements of a network.

Since the proposed modified greedy algorithm is a heuristic, we do not expect to solve (10) exactly, but enable the analysis in large networks. The one-step modified greedy algorithm is shown in Algorithm 1, and a complexity analysis is presented in Appendix A. The cost function as shown in Theorem 1 admits the following properties: it is submodular, nondecreasing, and satisfies the condition of taking the value 0 when no sensors are placed. The performance of the one-step modified greedy

algorithm is lower bounded by the standard greedy heuristic, which in turn is lower bounded by 63% of the optimal solution (for a fixed k) (Nemhauser et al., 1978), providing the aforementioned conditions are satisfied. This admits a lower bound on the performance of the one-step modified greedy algorithm as 63% of the optimal solution. We provide the proof of these claims in (Crowley and Esnaola, 2024).

3. Approaches to Validating Sensor Placements

To validate the sensor placement obtained by implementing Algorithm 1 on a given network, we will consider two case studies and compare the mutual information performance between different sensor placements and assess this performance based on methods we will now introduce.

3.1. Estimation of Unmonitored Nodes

3.1.1. Estimation Framework

The proposed sensor placement procedure targets mutual information as the performance measure. However, in practical settings, mutual information does not yield operational insight into the network performance. Knowledge of the state of the networks is vital for the management and operation of the network. Therefore, to validate the results of the proposed sensor placement Algorithm 1, we use conventional estimation techniques to estimate the state variables for the unmonitored nodes. The aim of doing this is twofold: firstly, it provides a benchmarking framework to compare different algorithm placement techniques with an operationally meaningful comparison measure. Secondly, it enables us to validate the use of mutual information as a performance measure for the design of the sensor placement procedure. Indeed, we show that the maximization of mutual information yields a performance improvement in the state estimation setting. Formally, consider the random variable vector $X^n = (X_1, \dots, X_n)^\mathsf{T}$ as the states at all node locations, and define the vector containing the k observations of the selected sensors by $Y^k := (Y_{i_1}, \dots, Y_{i_k})^\mathsf{T}$ with $i_j \in \mathcal{I}_k$ for $j = 1, \dots, k$. Furthermore, we define the states of unmonitored nodes as $X^{-k} = (X_{i_1}, \dots, X_{i_{n-k}})^\mathsf{T}$ with $i_j \in \mathcal{V} \setminus \mathcal{I}_k$ for $j = 1, \dots, n-k$. Given the state observations Y^k, we aim to estimate the unmonitored state variables, i.e. $\hat{X}^{-k} := g\left(Y^k\right)$, where $g : \mathbb{R}^k \to \mathbb{R}^{n-k}$ is the estimation procedure. We use two standard estimation procedures to assess the sensor placements: a general linear model (GLM) and a general regression neural network (GRNN). The GLM is adopted because of its low complexity and asymptotically tends to the minimum mean square error estimator. The GRNN is further adopted for comparison to the GLM because it is a purely data-driven method and has been used in various estimation applications in the water (Bowden et al., 2006) and wastewater (Heddam et al., 2016) literature. For the general linear model, we assume that the estimate is expressed as

$$X^{-k} = \begin{pmatrix} 1 \\ Y^k \end{pmatrix}^\mathsf{T} \boldsymbol{\beta} + \boldsymbol{\epsilon}, \tag{11}$$

where $\boldsymbol{\beta} \in \mathbb{R}^{(k+1)\times(n-k)}$ and $\boldsymbol{\epsilon} \in \mathbb{R}^{n-k}$ is the vector of independent normal random variables with mean 0 and variance σ_j^2. Note that for this model σ_j^2 varies for each node. The matrix $\boldsymbol{\beta}$ contains the parameters of the general linear model obtained using ordinary least squares. The function $g_1(Y^k)$ is then defined as

$$g_1(Y^k) := \begin{pmatrix} 1 \\ Y^k \end{pmatrix}^\mathsf{T} \boldsymbol{\beta}. \tag{12}$$

We also consider a general regression neural network (GRNN), first conceived in (D. F. Specht, 1991) which is a type of probabilistic neural network (Wasserman, 1993), that estimates the conditional expectation of a random output variable given some input variables. The GRNN is calculated by estimating the joint probability density function between input and output random variables using kernel density estimators which incorporate training data. By the definition of conditional probability and marginalizing the joint distribution, the conditional expectation of the state X_j, with $j \in \mathcal{V} \setminus \mathcal{I}_k$ is given by

$$\mathbb{E}\left[X_j | Y^k\right] = \frac{\displaystyle\int_{\mathbb{R}} x_j f_{Y^k, X_j}(Y^k, x_j)\, dx_j}{\displaystyle\int_{\mathbb{R}} f_{Y^k, X_j}(Y^k, x_j)\, dx_j}. \tag{13}$$

The joint probability density f_{Y^k, X_j} is often unknown in practice, therefore it can be estimated using sample training data and a non-parametric estimate of the joint probability density (kernel density estimators). If we assume that $f_{Y^k, X_j} \sim N_{k+1}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$, with $\boldsymbol{\mu}_j = \left(\mathbb{E}[Y^k], \mathbb{E}[X_j]\right)^\mathsf{T} \in \mathbb{R}^{k+1}$

and $\mathbf{\Sigma}_j \in S_{++}^{k+1}$, then the density can be expressed as

$$f_{Y^k,X_j}(\mathbf{y},x_j) = \left(\frac{1}{(2\pi)^{\frac{k+1}{2}}\det(\mathbf{\Sigma})^{\frac{1}{2}}}\right)\times \exp\left(-\frac{1}{2}\left(\begin{bmatrix}\mathbf{y}\\ x_j\end{bmatrix}-\boldsymbol{\mu}_j\right)^{\mathsf{T}}\mathbf{\Sigma}^{-1}\left(\begin{bmatrix}\mathbf{y}\\ x_j\end{bmatrix}-\boldsymbol{\mu}_j\right)\right). \tag{14}$$

By further assuming $\mathbf{\Sigma} = \sigma^2\mathbf{I}_{k+1}$, where $\sigma > 0$ is a spread parameter to be tuned by the user, and replacing $\boldsymbol{\mu}_j$ with the Q sample training points $\mathbf{y}^i \in \mathbb{R}^k$ and $x_j^i \in \mathbb{R}$ with $i = 1,2,\ldots,Q$, such that the non-parametric estimate of f_{Y^k,X_j}, denoted by $\hat{f}_{Y^k,X_j}$, can be expressed as

$$\hat{f}_{Y^k,X_j}(\mathbf{y},x_j) := \left(\frac{1}{(2\pi)^{\frac{k+1}{2}}\sigma^{k+1}}\right)\times \left[\frac{1}{n}\sum_{i=1}^{Q}\exp\left(-\frac{(\mathbf{y}-\mathbf{y}^i)^{\mathsf{T}}(\mathbf{y}-\mathbf{y}^i)}{2\sigma^2}\right)\times\exp\left(-\frac{(x_j-x_j^i)^2}{2\sigma^2}\right)\right]. \tag{15}$$

By substituting (15) into (13) and some manipulation, we obtain the conditional expectation given by

$$g_{X_j}(Y^k) := \frac{\sum_{i=1}^{Q} x_j^i \exp\left(-\frac{\|Y^k-\mathbf{y}^i\|_2^2}{2\sigma^2}\right)}{\sum_{i=1}^{Q}\exp\left(-\frac{\|Y^k-\mathbf{y}^i\|_2^2}{2\sigma^2}\right)}. \tag{16}$$

The GRNN estimate is then defined in closed form as

$$g_2(Y^k) := (g_{X_{j_1}}, g_{X_{j_2}}, \ldots, g_{X_{j_{n-k}}})(Y^k), \tag{17}$$

or more commonly formulated as a neural network, with the architecture shown in (D. F. Specht, 1991).

3.2. Loss Function

To validate the performance of the estimation, the loss function chosen is normalized mean square error (NMSE) due to being operationally meaningful across different orders of magnitudes for the states we want to estimate. For each k sensor placement, there will be $(n-k)$ unmonitored nodes we are estimating. Let us also define $\mathbf{X}^v$ as the matrix of validation data for unmonitored nodes and $\hat{\mathbf{X}}^v$ as the estimated matrix from each of the estimation methods, then the normalized mean square error of $\hat{\mathbf{X}}^v$ is given as

$$\text{NMSE}\left(\hat{\mathbf{X}}^v\right) = \frac{\|\mathbf{X}^v-\hat{\mathbf{X}}^v\|_F^2}{\|\mathbf{X}^v\|_F^2}. \tag{18}$$

Sewer network systems are mostly gravity-driven systems, and hence the larger flows in the network are predominantly downstream toward the end of the network. To ensure we do not penalize estimating smaller flows by mainly placing sensors in locations with larger flows, we adopt normalizing the loss function from the standard mean square error.

3.3. Heuristic Sensor Placements

To compare the performance of the sensor placements obtained in Algorithm 1, we also need to consider other heuristic sensor placements for comparison for estimation performance. Since there are currently no placements in the literature for our purpose that we are aware of, we have decided to compare choosing sensor placements according to:

- Nodes with the greatest (maximum) sum of state measurements in the training data (in the sensor placement process). Ranked from largest to smallest, and the first k locations are chosen for sensor placement.
- Randomly choosing locations with uniform probability for sensor placement.

Intuitively, random placements give an unbiased baseline for the amount of information gained about the network and an unbiased baseline performance obtained from the NMSE. Sensor placement based on the largest state measurements provides some physical insight into the underlying hydrodynamics and provides another purely state-driven benchmark for the performance of NMSE. Note that in practice, the method of placing sensors using the largest state measurements methodology is not practical. However, it provides an interesting opportunity to compare mutual information and NMSE between this placement and other heuristic methods.

Finally, a *rule based* approach is introduced and will be adopted in the more complex second case study. The locations of sensors are chosen in line with the following guidelines: (Britain, 1987, Section 2.3.1)

(i) At the system outfalls.

(ii) In sewers that drain large sub-catchments, placing sensors near trunk sewers.

(iii) At points along the main trunk or major junctions where the effects of major sub-catchment flows can be monitored.

The rule based approach incorporates expert knowledge obtained solely from GIS data and is the state-of-the-art benchmark for sewer systems, which provides a fair comparison when considering mutual information and NMSE performance.

4. Numerical Results

The two case studies we will present showcase a variety of different characteristics and data availability. For example, the first study looks at a relatively small single urban sewer network, and the latter study is a larger urban network with significantly more complexity in its topology where several catchments are individually clustered and connected. On data availability, the first case study has two days' worth of simulated hydraulic time series data, whereas the latter study has two years of simulated hydraulic time series data. The first case study is a foul system, so there is no precipitation involved - however, the second case study is a combined sewer network and so includes real-life precipitation measurements obtained from rain gauges. The dynamics of the system are inherently captured in the topology of the system, which is then reflected in the simulated time series data. This provides an interesting opportunity to test the impact of mutual information for both case studies which showcase different degrees of topology complexity as well as the presence and absence of rainfall. When the system is larger and more complex, we expect to see larger amounts of information contained in the system and more complex spatial relationships. By demonstrating our algorithm for both of these case studies, we show that the proposed algorithm is scalable and can be applied to any type of sewer network with a given time series data set.

Water utilities are increasingly choosing to install water level monitors over flow monitors due to their reduced cost. For this reason, we also consider the sensor placement problem for water depth measurements in the more complex second case study and contrast this placement with the flow sensor placement to observe any noticeable differences.

4.1. Case Study 1

Case Study 1 is based on an urban drainage catchment in the South of England. Due to confidentiality issues from the water utility that supplies services in this area, we do not provide any specific details, but a general overview of the catchment. The network model consists of 479 nodes and 567 links. For this paper, we were provided with simulation data obtained from a calibrated hydrodynamical model of this network from the wastewater utility operator. This consists of two days of dry weather flow data (m^3s^{-1}), at a 2-minute temporal resolution. The results consist of one-weekday daily flow profile and one-weekend daily flow profile at all nodes in the network, of which the first day in the data set is the weekday and the second day is the weekend profile. We only consider the flow rate in this case study. The network is populated by approximately 12000 people and covers an area of $5km^2$.

Figure 1: A map depicting the case study catchment in the South of England with data obtained from shape files supplied by the water utility.

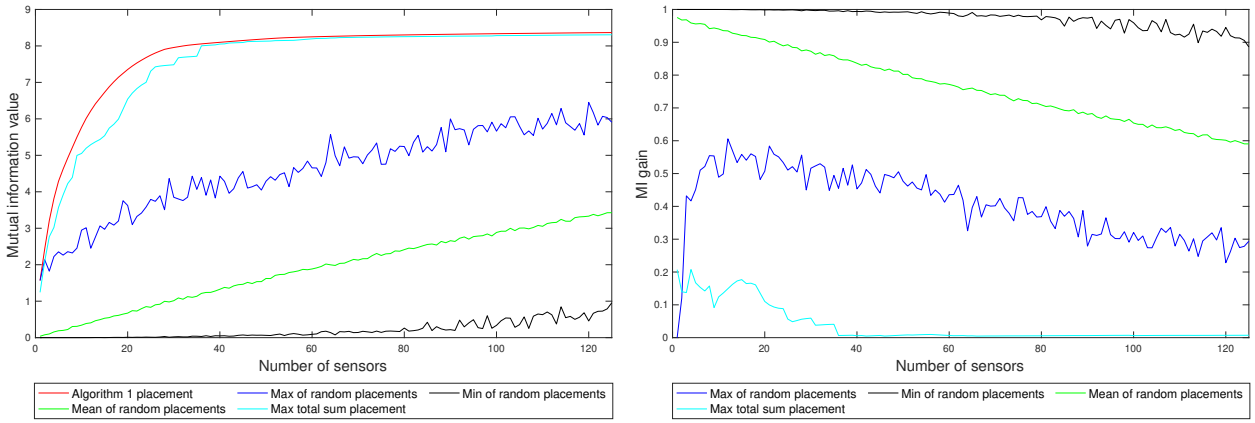

Figure 2: The first graph (left) displays the mutual information for each aforementioned sensor placement, ranging from 1 to 125 sensors placed in the network. The second graph (right) displays the mutual information gained from selecting the sensor placement found from Algorithm 1 compared to the other heuristic methods. We note that *max total sum placement* is short for nodes with the greatest sum of state measurements (see Section 3.3).

4.1.1. Data Preparation

For the catchment in the South of England, the 2-day time series of simulated volumetric flow data was split into two separate data sets. The training data set contains the first 70% of the 2-day time series of simulated volumetric flow data. The remaining 30% is to be used as our validation test set. The training set is used for the sensor placement procedure, and also for the training of the GLM and GRNN estimation methods. The validation test set will be used to validate the performance of both estimation techniques.

4.1.2. Sensor Placement

We apply Algorithm 1 to the training data set for the catchment in the South of England, and compare the mutual information obtained from the algorithm's sensor placement against the methods described in Section 3.3. We apply the sensor placement algorithm with the following constraints:

- We set the standard deviation σ of the additive noise introduced by the sensors measuring volumetric flow as $\sqrt{10^{-4}}$ $\mathrm{m^3 s^{-1}}$.

  The standard deviation is set to $\sqrt{10^{-4}}$ $\mathrm{m^3 s^{-1}}$ following advice given in the MCERTs report (Environment-Agency, 2020, Table 6, Pg. 9), which states that an acceptable standard deviation for flow sensors (class 3) should be within 4-5%, and hence the variance should be within 16-25%. This quantification of variance is not an exact physical magnitude, so we assume that accounting for some external noise (i.e. from installation), a realistic sensor variance for flow monitoring is given. For context, $\sqrt{10^{-4}}$ $\mathrm{m^3 s^{-1}}$ is equivalent to $\sqrt{0.1}$ litres s^{-1}.

- We run the modified greedy algorithm with a stopping criterion $k = 125$, which is about 25% of the total number of node locations.

- Given the parameters of the network, $n = 479$ and $k = 125$, the one-step modified greedy algorithm completed in approximately 18 minutes.

For each fixed number of sensors deployed, we randomly pick 1000 locations to simulate and compute the mutual information with respect to the minimum, mean, and maximum mutual information achieved for each simulation. The results are depicted in Figure 2. The results also display the worst-case gain in mutual information of $10 - 20\%$ until 20 sensors are placed when comparing the sensor placement obtained from Algorithm 1 with the sensor placement obtained from the maximum sum of state measurements heuristic. As the number of selected sensors increases to 40, the mutual information gain decreases to 0, i.e. our placement aligns very similarly with the nodes with the greatest sum of state measurements. When compared to the best mutual information achieved by random placement of sensors, the mutual information obtained with Algorithm 1 placement is 30-60% larger than the largest mutual information obtained from the random placement of 1000 location simulations.

Number of sensors	10		25		50		100	
Estimation method	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE
Algorithm 1	**0.0368**	**0.0311**	**0.0109**	**0.0352**	0.0070	**0.0198**	**0.0002**	**0.0058**
Largest total sum	0.2016	0.2163	0.0326	0.0571	**0.0050**	0.0215	0.0117	0.0495
Random placement 1	0.3593	0.3586	0.3178	0.3480	0.0738	0.0975	0.0170	0.0749
Random placement 2	0.3418	0.3478	0.3396	0.3843	0.0914	0.1157	0.3059	0.3825
Random placement 3	0.3592	0.3570	0.0988	0.1076	0.0587	0.1221	0.1750	0.3189
Random placement 4	0.3338	0.3340	0.2065	0.2066	0.3685	0.3726	0.0642	0.1640
Random placement 5	0.3586	0.3572	0.3156	0.3141	0.2727	0.3406	0.0435	0.0895

Table 1: Normalised mean square error results using the volumetric flow validation data set for estimation for the catchment in the South of England. The results show sensor placements for the set of numbers {10,25,50,100}, comparing Algorithm 1's sensor placement against some simple heuristics and 5 random placements. The smallest NMSEs for each sensor placement and estimation method are shown in bold font.

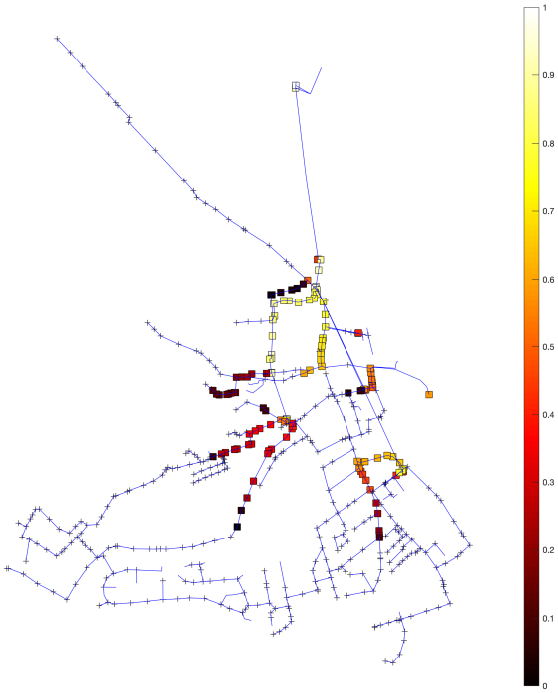

Figure 3: Likelihood of sensor placement for 125 sensor placements from Algorithm 1.

The heat map presented in Figure 3 shows the likelihood of each node being selected for each of Algorithm 1 placements when up to 125 sensor placements are considered. The heat map identifies the placements that are more likely to acquire more information. It is interesting to note that the locations chosen by Algorithm 1 are in the areas of high flows. However, when more than 40 sensors are placed in the network the sensor placement contains the same amount of mutual information as the max total sum placement, as shown in Figure 2. This suggests that for networks with low or moderate sensor deployment rates, the information content of the locations is not governed by the largest flow locations and mutual information successfully uncovers underlying dependencies between locations that are in turn leveraged by Algorithm 1.

4.1.3. Estimation Results

To validate the sensor placement chosen by Algorithm 1, we apply our estimation framework to the catchment in the South of England (Section 3.1). Note that the sensor placement observations used for the estimation simulations are the true state measurements since the observations are obtained from the hydrodynamic model (i.e. $Y^k = X^k$). Similarly to the sensor placement process, we use the training data for training the estimation procedures, and then estimation is performed over the remaining validation data according to the sensor placement.

The results shown in Table 1 highlight that using standard GLM and GRNN techniques, our proposed placement method performs better in terms of normalized mean square error than the other placements for different numbers of placed sensors. The exception is the case with 50 sensors, where the nodes with the greatest sum of state measurements slightly outperform the NMSE that Algorithm 1 yields by approximately 0.0019.

4.2. Case Study 2

Case study 2 is based on an urban drainage system in the village of Bellinge, Odense, Denmark. The network consists of 1020 nodes and 1050 links. The catchment model involves several smaller suburban towns, to the left in Figure 4, is Braendekilde. The center of Figure 4 shows Bellinge, and the far right of Figure 4 shows Dyrup. The wastewater treatment plant is downstream of Dyrup (not captured

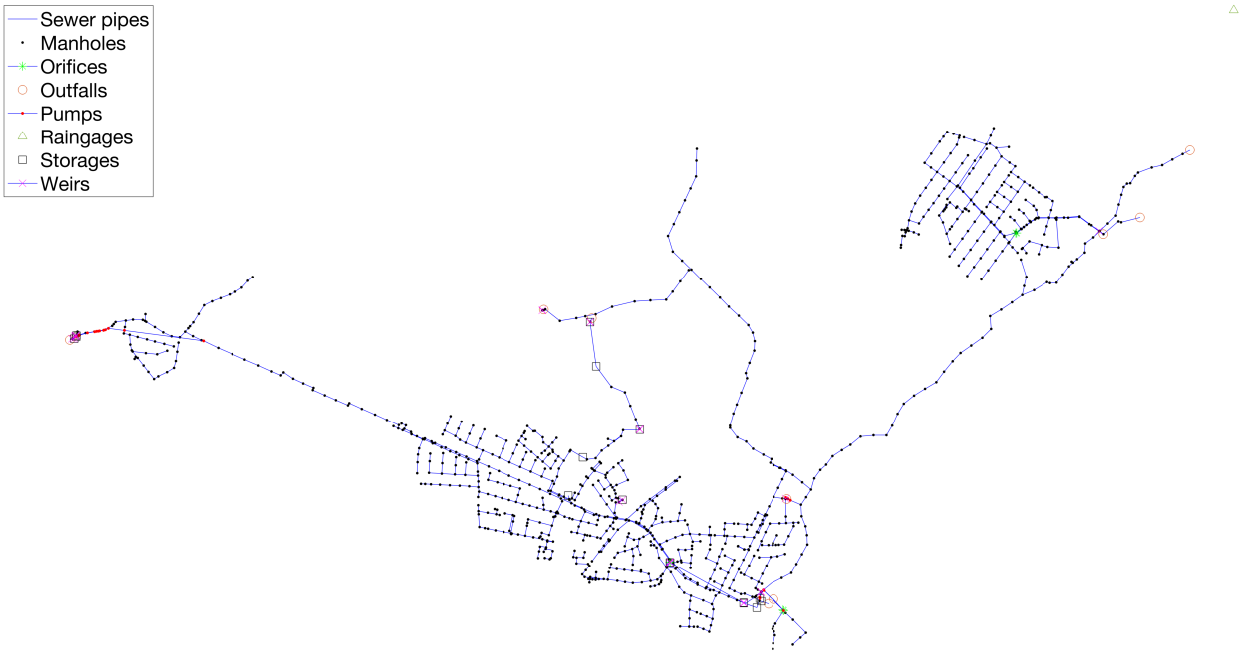

Figure 4: Graph depicting Bellinge from the SWMM shape files.

in the catchment), which travels to the Ejby Molle water resource recovery facility in central Odense (Pedersen et al., 2021, Figure 1). In this work, we use the Storm Water Management Model (SWMM) from (Pedersen et al., 2021) for Bellinge that generates flow rate and water depth data for a 2 year interval with 2-minute resolution. The model incorporates the observed rainfall data over the same 2 year interval over which flow rate data is generated. For a more detailed description of the topology of Bellinge and the model used herein, please refer to (Pedersen et al., 2021).

4.2.1. Data Creation and Preparation

The SWMM model for Bellinge is used to simulate four approximate six-month intervals for a total interval of two years of data. The dates simulated are shown below in (MM/DD/YYYY) format.

- Simulation 1: 01/01/2018 to 07/01/2018.
- Simulation 2: 07/01/2018 to 01/01/2019.
- Simulation 3: 01/01/2019 to 07/02/2019.
- Simulation 4: 07/02/2019 to 01/03/2020.

Simulations 1 and 2 are combined to form the first-year data set which we use as our training data set. Simulations 3 and 4 are combined to form the second-year data set which we use for validation purposes.

4.2.2. Sensor Placement

We apply Algorithm 1 to the training data set for Bellinge, and compare the sensor placements mutual information against those described in Section 3.3. We apply Algorithm 1 with the following constraints:

- We set the standard deviation σ introduced by the sensors to $\sqrt{10^{-4}}$ $\mathrm{m^3 s^{-1}}$, in line with Case Study 1.
- We run the modified greedy algorithm with stopping criterion $k = 250$, about 25% of the total number of node locations.
- Given the parameters of the network, $n = 1020$ and $k = 250$, the one-step modified greedy algorithm completed in approximately 14 hours.

In Figure 5, for each fixed number of sensors deployed, we simulate 1000 random realizations to select sensor locations and plot the mutual information with respect to the minimum, mean, and maximum mutual information achieved for each of those 1000 realizations.

Figure 6 shows the results obtained from Algorithm

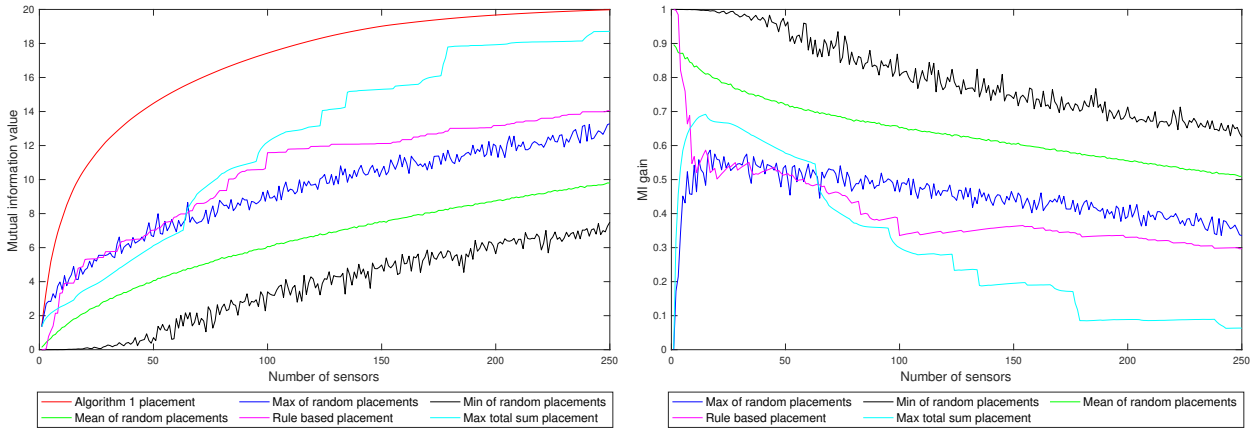

Figure 5: The first (left) graph displays the mutual information for each sensor placement method, running from 1 to 250 sensors placed in the network. The second (right) displays the gain in mutual information by selecting the sensor placement found from Algorithm 1 compared to the other heuristic methods.

Number of sensors	25		50		75		100	
Estimation method	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE
Algorithm 1	**0.0069**	**0.0343**	**0.0032**	**0.0284**	**0.0027**	**0.0273**	**0.0041**	**0.0333**
Rule based	0.0300	0.0604	0.0196	0.0570	0.0198	0.0562	0.0116	0.0448
Largest total sum	0.3149	0.3124	> 1	0.3021	> 1	0.3065	> 1	0.3119
Random placement 1	0.0655	0.0893	0.0680	0.0987	0.0880	0.0965	0.0866	0.0783
Random placement 2	> 1	0.0941	0.1213	0.0918	> 1	0.0690	> 1	0.0940
Random placement 3	0.0732	0.1047	> 1	0.0849	> 1	0.0845	> 1	0.0802
Random placement 4	0.1677	0.1506	0.0432	0.0776	> 1	0.0729	0.0588	0.0783
Random placement 5	0.2063	0.1157	> 1	0.0928	0.0619	0.0885	> 1	0.0805

Table 2: Normalised mean square error results using the volumetric flow validation data set for estimation in Bellinge. The results for the set $\{25, 50, 75, 100\}$ of sensors selected are shown, comparing Algorithm 1's sensor placement against heuristic placements and 5 random placements. The smallest NMSEs for each sensor placement and estimation method are shown in bold font.

Number of sensors	125		150		175		200	
Estimation method	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE
Algorithm 1	**0.0094**	0.0621	**0.0174**	0.1166	**0.0142**	0.0943	**0.0154**	0.1079
Rule based	0.0149	**0.0455**	0.0307	**0.0484**	0.0327	**0.0503**	0.0314	**0.0520**
Largest total sum	> 1	0.3212	> 1	0.3230	> 1	0.3260	> 1	0.3276
Random placement 1	> 1	0.0862	0.2182	0.0805	> 1	0.0746	> 1	0.0837
Random placement 2	> 1	0.0831	0.1929	0.0710	> 1	0.0809	0.1956	0.0807
Random placement 3	0.2601	0.0815	0.1546	0.0754	> 1	0.0758	> 1	0.0761
Random placement 4	0.3657	0.0803	0.0834	0.0774	> 1	0.0772	> 1	0.0732
Random placement 5	0.1052	0.0795	0.0735	0.0736	> 1	0.0762	> 1	0.0732

Table 3: Similarly to Table 2, we obtain NMSE results for the set $\{125, 150, 175, 200\}$ of sensors selected. The smallest NMSEs for each sensor placement and estimation method are shown in bold font.

1, which identifies several hot spots that contain larger amounts of information about the state of the network. Remarkably, the pipes that carry the flow from Bellinge to Dyrup are particularly informative, with other places of high information including the southern base of Bellinge, pipes exiting Braendekilde, and pipes exiting Dyrup.

4.2.3. Rule Based Placement

Considering the WRC rules from Section 3.3, the model is then further split into three segments to enforce an approximately uniform spatial density coverage. Segment 1 is Braendekilde, whose border is split midway between Bellinge and Braendekilde. Segment 2 is Bellinge, whose border is split midway between Bellinge and Dyrup, and segment three is Dyrup. For every 10 sensors deployed:

Figure 6: Likelihood of sensor placement for 250 sensor placements from Algorithm 1 using volumetric flow data.

(i) 1 sensor is placed in segment 1.

(ii) 7 sensors are placed in segment 2.

(iii) 2 sensors are placed in segment 3.

After approximately 100 sensors are placed, the remaining are placed spatially to get greater network coverage as the guidelines issued by the WRc are satisfied.

4.2.4. Estimation Results

Similarly to Case Study 1, the observations that we are using from the sensor placements are the true state measurements since the observations are obtained from a hydrodynamic model (i.e. $Y^k = X^k$). The results from Table 2 show better performance across the board for the sensor placement of Algorithm 1 when compared to the other placements for both estimation techniques. Table 3 shows similar performance for Algorithm 1 using the general linear model, whereas the rule based sensor placement performs better when using the general regression neural network.

4.2.5. Sensor Placement Example

We investigate limiting the number of sensors due to resource allocation constraints, e.g. budget constraints. We fix the number of placed sensors in the network as approximately 5% of the total node locations, for a total of 50 nodes. The results obtained by selecting this number are shown in Figure 8. We are particularly interested in understanding the range and variability of the error introduced by the estimation framework. To illustrate this, Figure 7 depicts the histograms of the square error in logarithmic base 10 for all unmonitored sites and realizations for both GLM and GRNN.

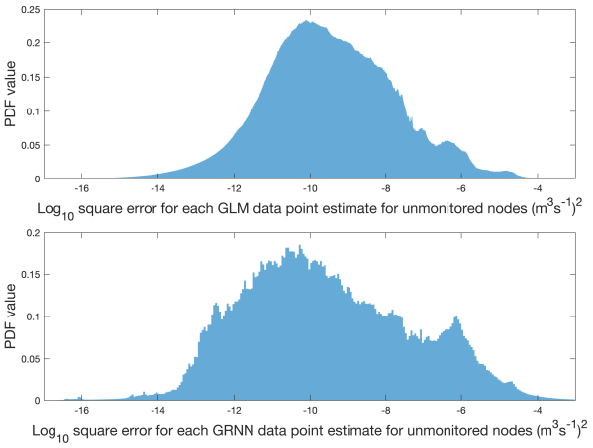

Figure 7: Logarithmic base 10 of the square errors for all realizations in the validation data set for both the GLM and GRNN estimation techniques.

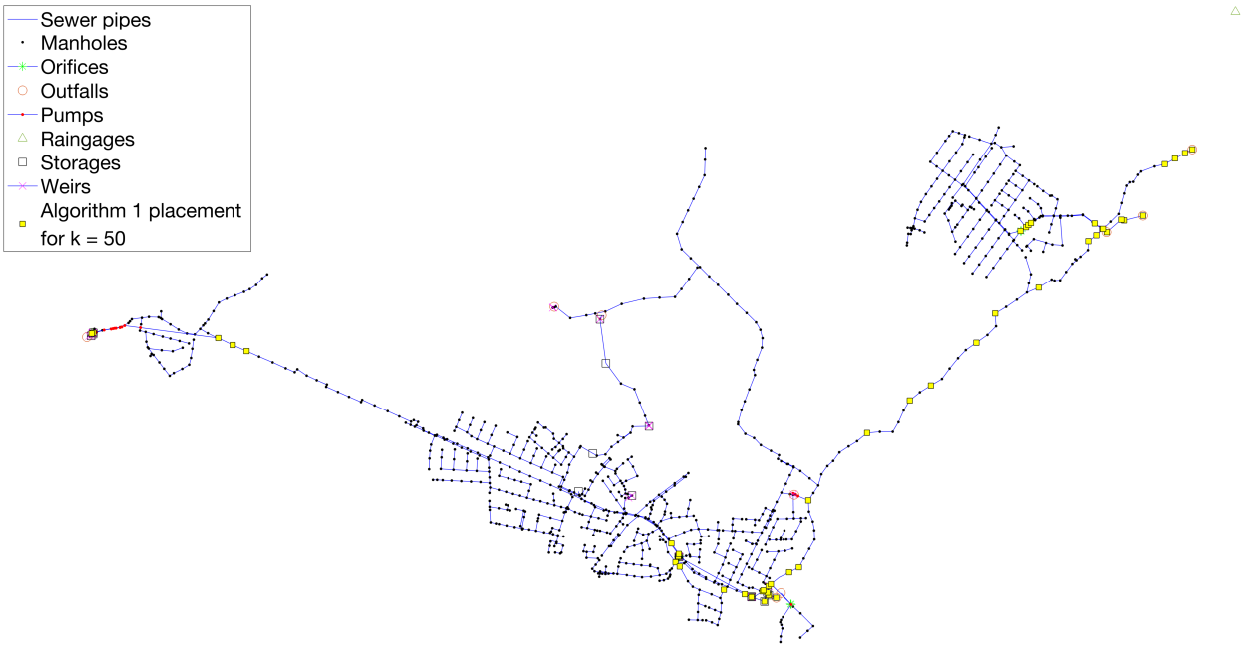

Figure 8: The sensor placement obtained by Algorithm 1 for 50 placed sensors.

The majority of square errors illustrated in Figure 7 fall within the interval $[10^{-16}, 10^{-4}]$, or $[-16, -4]$ as shown in logarithmic base 10. Both histograms show multimodal distributions, with secondary peaks appearing in the larger error tail. The GLM and GRNN demonstrate comparable average performance; however, the GRNN exhibits a heavier tail towards larger errors, while the GLM's error distribution decays more rapidly in this region. This implies that GRNN estimates are more likely to introduce larger errors but the bulk of the errors, i.e. the more typical errors, are smaller for the GRNN. On the other hand, the GRNN square errors are more spread out with comparatively heavier tails, and the mode of GRNN is slightly translated towards smaller errors.

We now turn our attention to the strict weather conditions case. Specifically, we investigate the estimation performance for both techniques for unmonitored nodes using the sensor placement depicted in Figure 8 under strict wet weather conditions. Two intervals of wet weather are selected:

- Event 1: 03/03/2019 to 03/17/2019.
- Event 2: 10/08/2019 to 10/19/2019.

The rainfall data for these events is provided within the rainfall *.dat* file attached to the SWMM model.

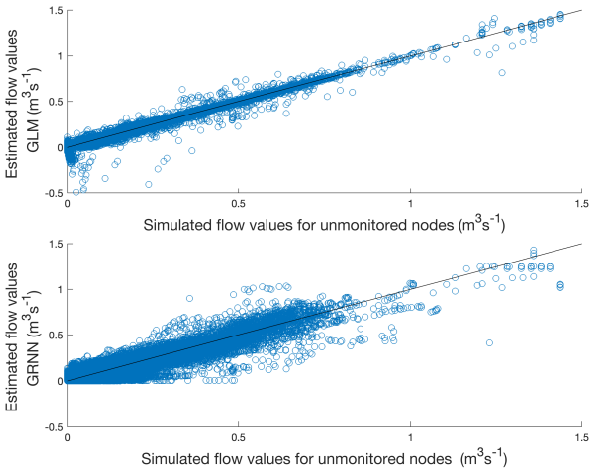

Figure 9: Simulated unmonitored flow realizations plotted against estimates obtained from the sensor placement in Figure 8 for realizations in the wet weather events. The scatter plots are plotted against the line $y = x$, scattered points below the line indicate the estimator has underestimated the flow and overestimated above the line $y = x$.

Figure 9 shows that for the given sensor placement, the GLM outperforms the GRNN with only a few outliers deviating far away from the line $y = x$, indicating that the GLM shows better results for

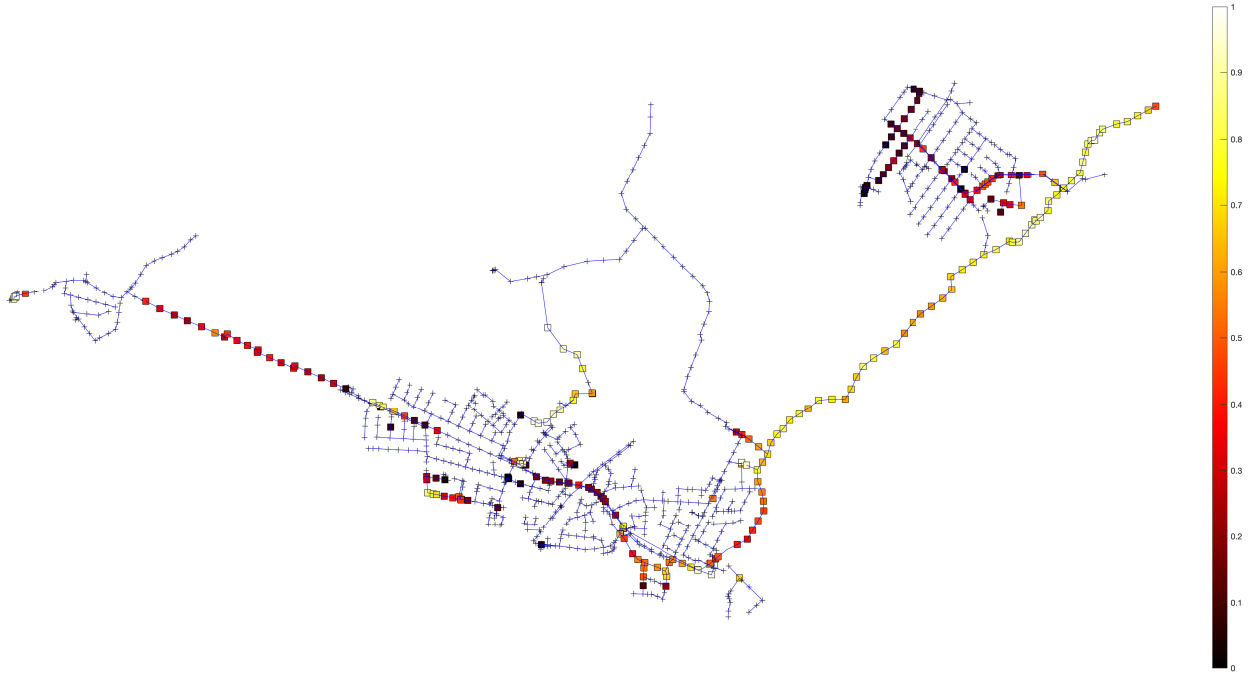

Figure 10: Likelihood of sensor placement for 250 sensor placements from Algorithm 1 using water depth data.

estimating flows in wet weather conditions for this sensor placement. The mean square error for the GLM and GRNN in these event intervals are 1.711×10^{-6} and 1.382×10^{-5} $(\mathrm{m}^3\mathrm{s}^{-1})^2$ respectively.

4.2.6. Water Depth Sensor Placement

So far, we have only considered sensor placements using volumetric flow data. We now consider water depth instead, i.e. readings obtained from an SLM, to analyze how the physical magnitude used to describe the state of the network impacts the placement produced by Algorithm 1.

- We set the standard deviation σ introduced by the level sensors as $\sqrt{25}$ mm.

  The standard deviation is set to $\sqrt{25}$ mm in light of the MCERTs report (Environment-Agency, 2020, Pg. 5), which states that the resolution requirements for level sensors in a certification range of 1 m to 5 m for class 3 sensors should be less than 20 mm. We account for external noise in our setting, i.e. from installation or the conditions of the sewer that govern the noise, so setting the standard deviation to $\sqrt{25}$ mm is a sensible choice.

- We run the modified greedy algorithm with stopping criterion $k = 250$, about 25% of the total number of node locations.

The results obtained from Algorithm 1 using depth data are shown in Figure 10. The estimation simulations run in Section 4.2.4 are also run for water depth data. The rule based sensor placement and the random sensor placements are unchanged, and the largest total sum heuristic placement is updated based on the water depth data.

It is worth noting the difference in the likelihood of sensor placements between volumetric flow and water depth data for Algorithm 1's sensor placement. To illustrate the contrast, a heat map is shown in Figure 11 depicting the absolute difference of the likelihood ratios for both data scenarios. The heatmap displays differences and highlights that the most significant change pertains to the main cluster located in the sewers joining Bellinge from the north (from the west side).

4.3. Interpretation of Results

The results shown in Table 2 for flow estimation indicate that for both the GLM and GRNN, the NMSE obtained by the placement of Algorithm 1 outperforms the rule based and heuristic placements when up to 100 sensors are considered. Fig-

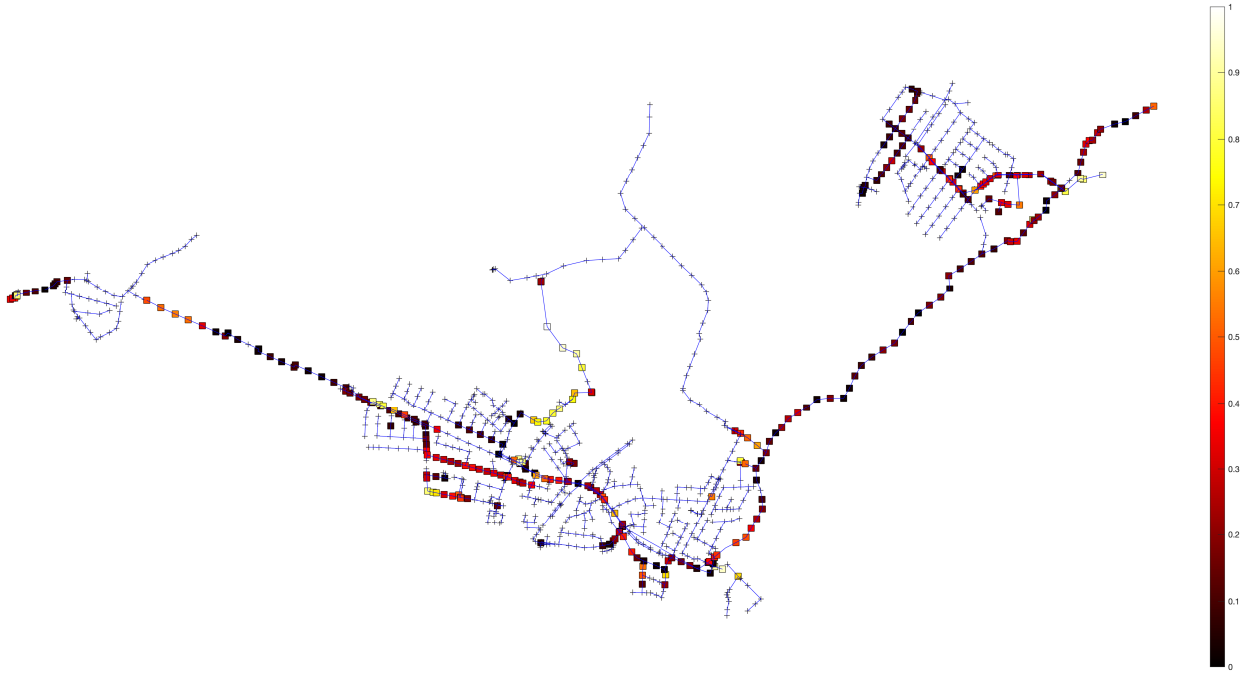

Figure 11: **Absolute difference of the likelihoods** between sensor placements from volumetric flow data vs water depth data for 250 sensor placements. When observing the absolute ratio difference, the closer to white (or the value 1), means the node was virtually chosen for every sensor placement with water depth data but wasn't selected with volumetric flow data.

Number of sensors	25		50		75		100	
Estimation method	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE
Algorithm 1	**0.0103**	**0.0142**	**0.0051**	**0.0078**	**0.0042**	**0.0065**	**0.0035**	**0.0041**
Rule based	0.0472	0.0408	0.0191	0.0191	0.0241	0.0193	0.0156	0.0199
Largest total sum	0.0747	0.0706	0.0749	0.0702	0.0971	0.0612	0.2649	0.0621
Random placement 1	0.0670	0.0576	0.5498	0.0517	0.1407	0.0426	0.1209	0.0346
Random placement 2	0.2600	0.0285	0.5335	0.0580	0.1577	0.0517	> 1	0.0306
Random placement 3	0.0956	0.0625	> 1	0.0754	> 1	0.0606	0.1617	0.0266
Random placement 4	0.1317	0.0454	> 1	0.0608	0.1618	0.0160	0.1374	0.0405
Random placement 5	0.4481	0.0669	0.0697	0.0358	0.0961	0.0298	> 1	0.0284

Table 4: Normalised mean square error results using the water depth validation data set for estimation in Bellinge. The results for the set $\{25, 50, 75, 100\}$ of sensors selected are shown, comparing Algorithm 1's sensor placement against heuristic placements and 5 random placements. The smallest NMSEs for each sensor placement and estimation method are shown in bold font.

Number of sensors	125		150		175		200	
Estimation method	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE
Algorithm 1	0.0194	0.0295	**0.0136**	0.0277	**0.0131**	0.0273	**0.0129**	0.0232
Rule based	**0.0086**	**0.0110**	0.0211	**0.0099**	0.0311	**0.0097**	0.0228	**0.0088**
Largest total sum	0.9007	0.0628	> 1	0.0634	> 1	0.0639	> 1	0.0640
Random placement 1	> 1	0.0285	0.0362	0.0223	0.1910	0.0221	> 1	0.0285
Random placement 2	0.3273	0.0246	0.1654	0.0177	0.1083	0.0265	0.8148	0.0249
Random placement 3	0.7026	0.0296	0.7541	0.0388	> 1	0.0376	> 1	0.0167
Random placement 4	0.5170	0.0397	0.2257	0.0461	0.2015	0.0272	0.2666	0.0207
Random placement 5	> 1	0.0261	0.0752	0.0251	0.1716	0.0203	0.9024	0.0261

Table 5: Similarly to Table 4, we obtain NMSE results for the set $\{125, 150, 175, 200\}$ of sensors selected. The smallest NMSEs for each sensor placement and estimation method are shown in bold font.

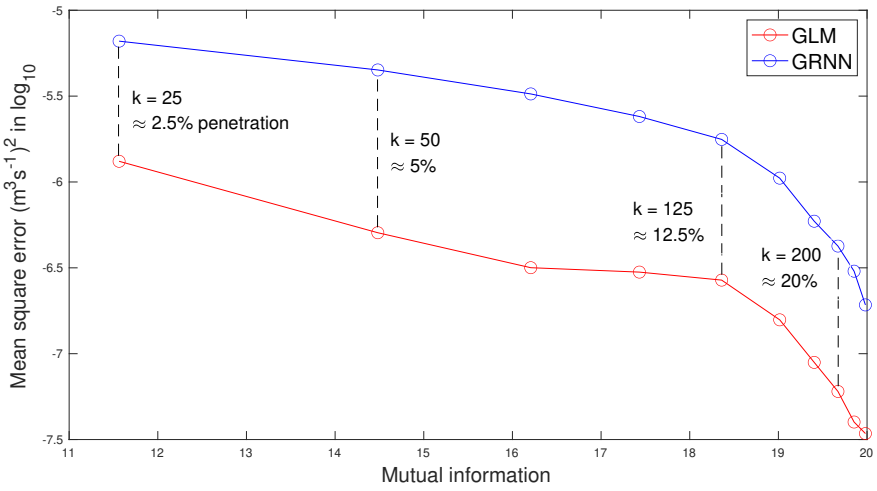

Figure 12: A plot showing the mean square error of Algorithm 1's sensor placement estimating flow at unmonitored nodes vs mutual information obtained from Algorithm 1's sensor placement. This was plotted for each number of sensors we simulated in Tables 2 and 3.

ure 5 (right) shows in comparison to the estimation results that our proposed sensor placement method yields at least a 40% gain in mutual information with respect to all other sensor placements when considering between 5 and 50 sensors. Between 50 and 100 sensors, we observe a gain in mutual information of at least 30%. However, for the same interval, the total sum placement obtains more mutual information than the rule based placement but performs worse in terms of NMSE for all estimation methods. The results in Table 3 suggest that Algorithm 1's sensor placement displayed the best NMSE using the GLM, yet the rule based placement yielded a better NMSE using the GRNN in comparison with the GRNN from Algorithm 1's sensor placement.

To assess the interplay between mutual information and the estimation performance that results from Algorithm 1 placements we plot both measures in Figure 12. Interestingly, the mean square error decay is moderate for low values of k but decreases exponentially fast with the mutual information for large values of k. Remarkably, the transition seems to occur around 125 placed sensors, which suggests a phase transition type effect once a sufficient amount of sensors are placed in the network.

5. Discussion

The results shown from both case studies show that by using our proposed Algorithm to maximize mutual information, we obtain superior estimation results using NMSE in comparison to other heuristic sensor placements. The simulated network cases have significantly different topology and complexity characteristics, as well as the amount of data used, weather conditions, etc. Yet, both networks visually demonstrate similar performances when applying our proposed sensor placement framework (i.e. outperforming other sensor placements for both mutual information and NMSE estimation performance). Figure 12 further supports our hypothesis, that as we maximize mutual information, MSE decreases monotonically with the number of placed sensors.

However, one limitation within the problem formulation and algorithm implementation that should be noted is that specific network elements such as storage tanks and outfalls, etc are all treated equally as 'nodes' alongside manholes, which of course is operationally significant for the intended use of each element. Specifically, we see this within Case Study 2. To circumvent this issue, the user may wish to implement their specific constraints with this in mind. For example, modifying the optimization problem

such that:

$$\mathbf{H}_k^* = \underset{\mathbf{H}\in\mathcal{H}_k}{\arg\max}\ \frac{1}{2}\log\left(\frac{1}{\sigma^{2k}}\det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^\mathsf{T}+\sigma^2\mathbf{I}_k\right)\right),$$
$$\text{Subject to: User requirements 1,}$$
$$\text{User requirements 2,}$$

where

- User requirements 1: Preselect $\{X_{i_1},\ldots,X_{i_y}\} \subset \mathbf{H}, y < k$ for outfalls, storages (user specific).
- User requirements 2: Sparsity restraints. User-specific requirements such as distance or connectivity constraints between sensor placements.

With this in mind, the user can pick (in user requirements 1) operationally important nodes (for example, storage tanks) to be preselected within the algorithm's sensor placement choice. For user requirements 2, if required, the user can specify some type of sparsity in sensor placement chosen by the algorithm. To implement these constraints into the algorithm, only minor modifications within the search space would be needed. We provide sensor placement examples with the distance constraints applied to Case Study 2 in the supplementary material. The distance constraint that introduced sparsity to the sensor placement degraded the NMSE compared to the unconstrained case described in the paper.

To implement the proposed framework for an arbitrary wastewater network, the user requires access to a hydrodynamic model (and hence the capacity to generate simulated flow data) and the respective GIS files associated with the hydrodynamical model. The code that runs the simulations and shows the figures seen throughout this paper converts the GIS shape files into structured arrays in MATLAB and relies on the order of time series data for the nodes to be in the same order as the nodes in the structured array.

The results displayed in both case studies show that sensor placements that maximize the amount of information in sewer networks tend to cluster, i.e. there are parts of the network that are more locally informative about the global state of the network. We note that should the user want to implement other constraints on the proposed sensor placement algorithm, this can be attained at the expense of decreasing the mutual information.

The estimation techniques used in this study are standard generic techniques that require minimal tuning for implementation. The goal of this paper is not to devise the best estimation technique but rather to showcase the performance of different standard techniques under different sensor placement strategies.

6. Conclusion

This work presents an objective and computationally efficient approach to optimizing sensor placements for large sewer networks using mutual information as the performance measure. Validation of performance is obtained using standard estimation techniques. We show that the sensor placement obtained from Algorithm 1 performs significantly better for varying amounts of sensors when considering normalized mean square error in comparison to other heuristic sensor placement approaches in both case studies presented. We conclude that mutual information is an appropriate performance measure for sensor placement procedures that aim to perform network estimation in sewer networks. To the best of the knowledge of the authors, this is the first work that considers sensor placement in sewer networks for network state estimation by using mutual information. The proposed framework and modified greedy algorithm provide wastewater utility operators with a systematic sensor placement method that enables them to design the data acquisition and monitoring infrastructure for the network. The proposed method provides robust information acquisition guarantees that translate into the utility of the data for a wide range of applications. The low computational burden of the proposed solution opens the door to designing sensor placements for large-scale networks and to the development of resilient monitoring solutions that can be evaluated for a large range of operational regimes.

6.1. CRediT Authorship Contribution Statement

George Crowley: Writing – original draft, Writing – review & editing, Conceptualization, Formal Analysis, Investigation, Methodology, Resources, Software, Validation, Visualization.

Iñaki Esnaola: Writing – review & editing, Conceptualization, Methodology, Supervision, Validation.
Simon Tait: Writing – review & editing, Conceptualization, Methodology, Supervision.
George Panoutsos: Writing – review & editing, Supervision.
Vanessa Speight: Writing – review & editing, Supervision.

6.2. 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.

6.3. Data Availability

The data used in Case Study 1 is not publicly available, but the data used in Case Study 2 will be made available. The code to reproduce the general results alongside where to download the data can be found in the following GitHub repository: `https://github.com/George-Crowley/WR_ITSP`.

6.4. Acknowledgements

The author GC acknowledges the financial contribution of Thames Water, UK Engineering and Physical Sciences Research Council, and the Centre for Doctoral Training in Water Infrastructure and Resilience (EP/S023666/1).

Appendix A.

Proof of Theorem (1). We first denote the joint probability density of X^n and Y^k as f_{X^n,Y^k}, which can be written as

$$\begin{aligned} f_{X^n,Y^k} &\sim N\left(\begin{pmatrix}\boldsymbol{\mu}\\ \mathbf{H}\boldsymbol{\mu}\end{pmatrix}, \begin{pmatrix}\boldsymbol{\Sigma} & \boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}\\ \mathbf{H}\boldsymbol{\Sigma} & \mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}+\sigma^2\mathbf{I}_k\end{pmatrix}\right),\\ &\sim N(\mu_{n+k}, \boldsymbol{\Sigma}_{n+k}). \end{aligned}$$

We can then calculate the mutual information between X^n and Y^k as

$$\begin{aligned} I(X^n;Y^k) &= \mathrm{tr}\left(\mathbb{E}\left[\log\frac{f_{X^n,Y^k}}{f_{X^n}f_{Y^k}}\right]\right)\\ &= \mathbb{E}\left[\mathrm{tr}\left(\log\frac{f_{X^n,Y^k}}{f_{X^n}f_{Y^k}}\right)\right]\\ &= \log\left[\frac{(2\pi)^{(n+k)/2}\det(\boldsymbol{\Sigma})^{1/2}\det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}+\sigma^2\mathbf{I}_k\right)^{1/2}}{(2\pi)^{(n+k)/2}\det\left(\boldsymbol{\Sigma}_{n+k}\right)^{1/2}}\right]\\ &= \log\left[\frac{\det(\boldsymbol{\Sigma})^{1/2}\det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}+\sigma^2\mathbf{I}_k\right)^{1/2}}{\det\left(\boldsymbol{\Sigma}\right)^{1/2}\det\left(\sigma^2\mathbf{I}_k\right)^{1/2}}\right]\\ &= \log\left[\frac{\det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}+\sigma^2\mathbf{I}_k\right)^{1/2}}{\det\left(\sigma^2\mathbf{I}_k\right)^{1/2}}\right]\\ &= \frac{1}{2}\log\left(\frac{1}{\sigma^{2k}}\det\left(\mathbf{H}\boldsymbol{\Sigma}\mathbf{H}^{\mathsf{T}}+\sigma^2\mathbf{I}_k\right)\right). \end{aligned}$$

For the calculation of $\det(\boldsymbol{\Sigma}_{n+k})$, we have used the Schur complement property as described in (Seber, 2007, Pg. 296: 14.17. (c)). □

Complexity Analysis of the One-step Modified Greedy Algorithm.

The matrix $\mathbf{H}_j\boldsymbol{\Sigma}\mathbf{H}_j^{\mathsf{T}}+\sigma^2\mathbf{I}_j$ is positive definite ($\in S_j^{++}$) since $\mathbf{H}_j\boldsymbol{\Sigma}\mathbf{H}_j^{\mathsf{T}}$ is a principle submatrix of $\boldsymbol{\Sigma}$, which is positive definite, and $\boldsymbol{\Sigma}$ and $\sigma^2\mathbf{I}_j$ are both symmetric positive definite ($\boldsymbol{\Sigma}$ by assumption). Using the property that $\mathbf{H}_j\boldsymbol{\Sigma}\mathbf{H}_j^{\mathsf{T}}+\sigma^2\mathbf{I}_j$ is positive definite, there exists a Cholesky decomposition of $\mathbf{H}_j\boldsymbol{\Sigma}\mathbf{H}_j^{\mathsf{T}}+\sigma^2\mathbf{I}_j$, which costs $\mathcal{O}\left(j^3\right)$ operations to compute (Press et al., 2007, Pg. 100). With the Cholesky factorization, the determinant can be calculated using elementary operations. For one iteration of the modified greedy algorithm, stopping once we have reached k selected sensor placements, costs $\mathcal{O}\left(\sum_{j=2}^{k}(n-j+1)(j)^3\right)$ operations, since there are $(n-j+1)$ possible branches for the greedy heuristic to search, and the determinant cost is $(j)^3$ operations. From the modified greedy algorithm, we are now searching over each of the n initial nodes, and hence the total cost is $\mathcal{O}\left(\sum_{j=2}^{k}n(n-j+1)(j)^3\right)$ operations. It follows that

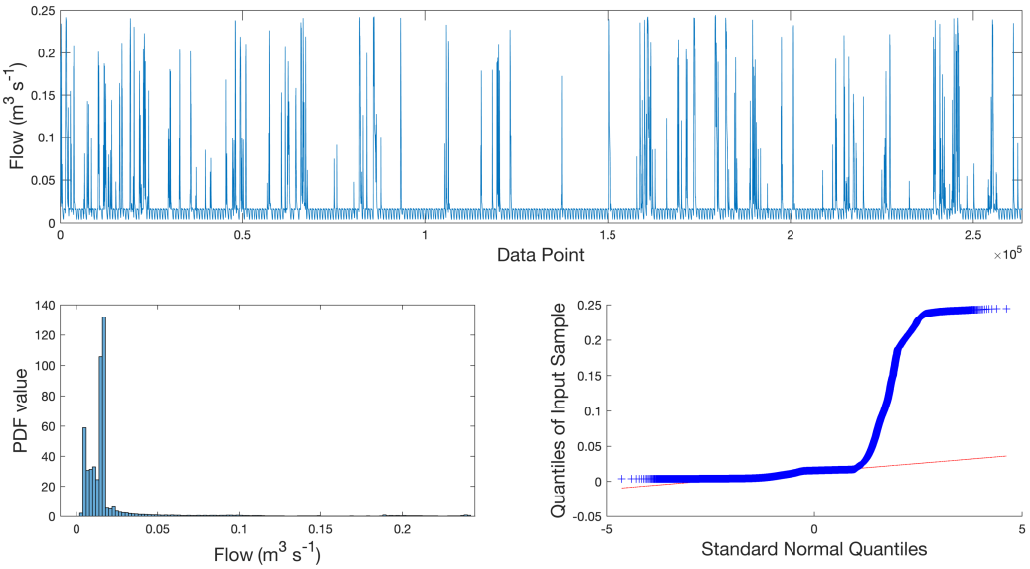

Figure 13: This figure illustrates several interesting plots of node 1 from training data for flow from the Bellinge model (Case Study 2). The top figure shows the time series plot for each data point. The figure in the bottom left shows a histogram of the training data, and finally, the last image is a *qqplot* which measures similarity to the normal distribution, which we have assumed in Assumption 3 (A3).

$$\begin{aligned}\mathcal{O}\left(\sum_{j=2}^{k} n(n-j+1)(j)^3\right) &= \mathcal{O}\left(\sum_{j=2}^{k} n^2(j)^3\right)\\ &= \mathcal{O}\left(n^2\left(\sum_{j=1}^{k} j^3 - 1\right)\right)\\ &= \mathcal{O}\left(n^2k^4\right).\end{aligned}$$

Simulations for computing Algorithm 1's sensor placement times were run on Matlab 2023a, using a Mac studio with an Apple M1 max chip and 32GB of unified memory.

Appendix B.

Link to supplementary material.

References

Ashley, R., Hopkinson, P., 2002. Sewer systems and performance indicators—into the 21st century. Urban Water 4, 123–135.

Banik, B., Alfonso, L., Di Cristo, C., Leopardi, A., 2017a. Greedy Algorithms for Sensor Location in Sewer Systems. Water 9, 856.

Banik, B., Alfonso, L., Di Cristo, C., Leopardi, A., Mynett, A., 2017b. Evaluation of Different Formulations to Optimally Locate Sensors in Sewer Systems. Journal of Water Resources Planning and Management 143.

Banik, B., Alfonso, L., Torres, A., Mynett, A., Di Cristo, C., Leopardi, A., 2015. Optimal Placement of Water Quality Monitoring Stations in Sewer Systems: An Information Theory Approach. Computing and Control for the Water Industry (CCWI2015) Sharing the best practice in water management 119, 1308–1317.

Bowden, G.J., Nixon, J.B., Dandy, G.C., Maier, H.R., Holmes, M., 2006. Forecasting chlorine residuals in a water distribution system using a general regression neural network. Application of Natural Computing Methods to Water Resources and Environmental Modelling 44, 469–484.

Britain, W.R.C.G., 1987. A Guide to Short Term Flow Surveys of Sewer Systems. WRc Engineering.

C. E. Shannon, 1948. A Mathematical Theory of Communication. The Bell System Technical Journal 27, 379–423.

Calle, E., Martínez, D., Brugués-i Pujolràs, R., Farreras, M., Saló-Grau, J., Pueyo-Ros, J., Corominas, L., 2021. Optimal selection of monitoring sites in cities for SARS-CoV-2 surveillance in sewage networks. Environment International 157, 106768.

Clemens, F., 2002. Evaluation of a Method for the Design of Monitoring Networks in Urban Drainage. Pages: 17.

Cover, T.M., 2005. Elements of Information Theory. John Wiley & Sons.

Crowley, G., Esnaola, I., 2024. Submodularity of Mutual Information for Multivariate Gaussian Sources with Additive Noise. URL: `https://arxiv.org/abs/2409.03541`,

arXiv:2409.03541.
D. F. Specht, 1991. A general regression neural network. IEEE Transactions on Neural Networks 2, 568–576.
Environment-Agency, 2020. MCERTS: performance standards and test procedures for continuous water monitoring equipment - part 3 water flowmeters. URL: https://assets.publishing.service.gov.uk/media/5e5f6038d3bf7f108889c93e/MCERTS-performance-standards.pdf. (Visited on 2024-05-01).
Faris, N., Zayed, T., Aghdam, E., Fares, A., Alshami, A., 2024. Real-Time sanitary sewer blockage detection system using IoT. Measurement 226, 114146.
Fattoruso, G., Agresta, A., Guarnieri, G., Lanza, B., Buonanno, A., Molinara, M., Marrocco, C., De Vito, S., Tortorella, F., Francia, G.D., 2015. Optimal Sensors Placement for Flood Forecasting Modelling. Computing and Control for the Water Industry (CCWI2015) Sharing the best practice in water management 119, 927–936.
Heddam, S., Lamda, H., Filali, S., 2016. Predicting Effluent Biochemical Oxygen Demand in a Wastewater Treatment Plant Using Generalized Regression Neural Network Based Approach: A Comparative Study. Environmental Processes 3, 153–165.
Javaid, M., Haleem, A., Singh, R.P., Rab, S., Suman, R., 2021. Significance of sensors for industry 4.0: Roles, capabilities, and applications. Sensors International 2, 100110.
Jungnickel, D., 2013. The Greedy Algorithm, in: Graphs, Networks and Algorithms. Springer Berlin Heidelberg, pp. 135–161.
K. Deb, A. Pratap, S. Agarwal, T. Meyarivan, 2002. A fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation 6, 182–197.
Kang, O., Lee, S., Wasewar, K., Kim, M., Liu, H., Oh, T., Janghorban, E., Yoo, C., 2013. Determination of key sensor locations for non-point pollutant sources management in sewer network. Korean Journal of Chemical Engineering 30, 20–26.
Ko, C.W., Lee, J., Queyranne, M., 1995. An Exact Algorithm for Maximum Entropy Sampling. Operations Research 43, 684–691.
Krause, A., Leskovec, J., Guestrin, C., Vanbriesen, J., Faloutsos, C., 2008a. Efficient Sensor Placement Optimization for Securing Large Water Distribution Networks. ASCE Journal of Water Resources Planning and Management 134, 516–526.
Krause, A., Singh, A., Guestrin, C., 2008b. Near-Optimal Sensor Placements in Gaussian Processes: Theory, Efficient Algorithms and Empirical Studies. Journal of Machine Learning Research 9, 235–284.
Larson, R.C., Berman, O., Nourinejad, M., 2020. Sampling manholes to home in on SARS-CoV-2 infections. PLOS ONE 15, e0240007.
Lee, J., Fampa, M., 2022. Maximum-Entropy Sampling: Algorithms and Application. Springer Nature.
Li, J., 2021. Exploring the potential of utilizing unsupervised machine learning for urban drainage sensor placement under future rainfall uncertainty. Journal of Environmental Management 296, 113191.
Nash, J., Sutcliffe, J., 1970. River flow forecasting through conceptual models part I — A discussion of principles. Journal of Hydrology 10, 282–290.
Nemhauser, G.L., Wolsey, L.A., Fisher, M.L., 1978. An analysis of approximations for maximizing submodular set functions — I. Mathematical Programming 14, 265–294.
Nourinejad, M., Berman, O., Larson, R.C., 2021. Placing sensors in sewer networks: A system to pinpoint new cases of coronavirus. PLOS ONE 16, e0248893.
OFWAT, 2017. Reporting guidance – Sewer flooding. URL: https://www.ofwat.gov.uk/wp-content/uploads/2018/03/Reporting-guidance-sewer-flooding-updated-April-2018.pdf. (Visited on 2024-04-02).
Ogie, R., Shukla, N., Sedlar, F., Holderness, T., 2017. Optimal placement of water-level sensors to facilitate data-driven management of hydrological infrastructure assets in coastal mega-cities of developing nations. Sustainable Cities and Society 35, 385–395.
Pedersen, A., Pedersen, J., Vigueras-Rodríguez, A., Brink-Kjaer, A., Borup, M., Mikkelsen, P., 2021. The Bellinge data set: Open data and models for community-wide urban drainage systems research. Earth System Science Data 13, 4779–4798.
Press, W., Teukolsky, S., Vetterling, W., Flannery, B., 2007. Numerical Recipes 3rd Edition: The Art of Scientific Computing.
Rosin, T.R., Kapelan, Z., Keedwell, E., Romano, M., 2022. Near real-time detection of blockages in the proximity of combined sewer overflows using evolutionary ANNs and statistical process control. Journal of Hydroinformatics 24, 259–273.
Rossman, L.A., 2010. Storm water management model user's manual, version 5.0. Cincinnati: National Risk Management Research Laboratory, Office of Research and Development, US Environmental Protection Agency.
Seber, G., 2007. A Matrix Handbook for Statisticians. John Wiley & Sons, Inc.
Simone, A., Di Cristo, C., Guadagno, V., Del Giudice, G., 2023. Sewer networks monitoring through a topological backtracking. Journal of Environmental Management 346, 1–17.
Storn, R., Price, K., 1997. Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization 11, 341–359.
Sumer Derya, Gonzalez Javier, Lansey Kevin, 2007. Real-Time Detection of Sanitary Sewer Overflows Using Neural Networks and Time Series Analysis. Journal of Environmental Engineering 133, 353–363.
Taillard, E.D., 2023. Design of Heuristic Algorithms for Hard Optimization. Graduate Texts in Operations Research, Springer Nature.
Vonach, T., Tscheikner-Gratl, F., Rauch, W., Kleidorfer, M., 2018. A Heuristic Method for Measurement Site Selection in Sewer Systems. Water 10.
Wang, S., Zhang, X., Wang, J., Tao, T., Xin, K., Yan, H., Li, S., 2023. Optimal sensor placement for the routine monitoring of urban drainage systems: A re-clustering method. Journal of Environmental Management 335, 117579.
Wasserman, P.D., 1993. Advanced methods in neural computing. John Wiley & Sons, Inc.
Yazdi, J., 2018. Water quality monitoring network design for urban drainage systems, an entropy method. Urban Water Journal 15, 227–233.

Supplementary Material for 'Information-Theoretic Sensor Placement for Large Sewer Networks'

George Crowley, Simon Tait, George Panoutsos, Vanessa Speight, Iñaki Esnaola

October 28, 2024

In this supplementary material, we assess the proposed distance constraints in our sensor placement framework. We first note that different distance constraints (e.g. 150m or 250m) enforces a different amount of possible sensor placements, i.e. more sensors can be placed under a 150m constraint than a 250m constraint for a fixed network. Hence, the 25% coverage we originally showed in all papers' original figures of the manuscript can't be attained under these constraints.

We provide two examples of the proposed distance between sensor placements constraint, 150m and 250m. We consider the volumetric flow sensor placements for these comparisons.

1. **150m distance constraints**

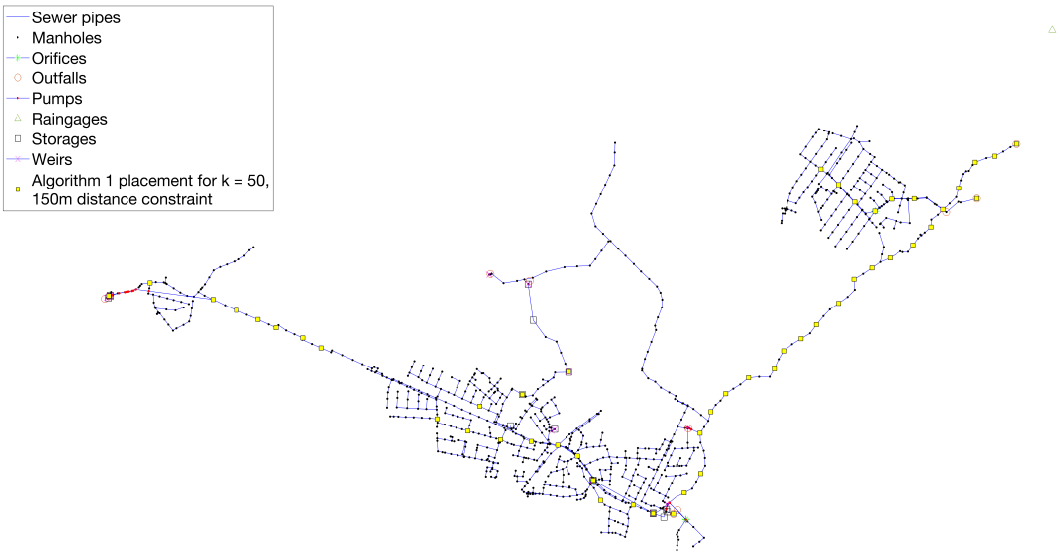

Figure 1: The sensor placement obtained by Algorithm 1 for 50 placed sensors with 150m distance constraints between chosen nodes.

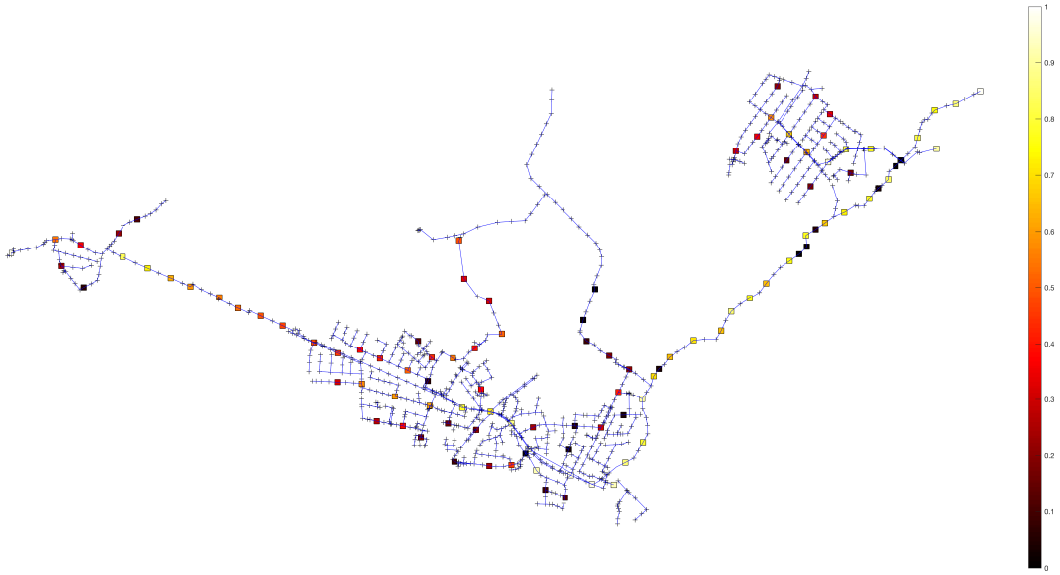

Figure 2: Likelihood of sensor placement for 100 sensor placements from Algorithm 1 using volumetric flow data with 150m distance constraints.

2. **250m distance constraints**

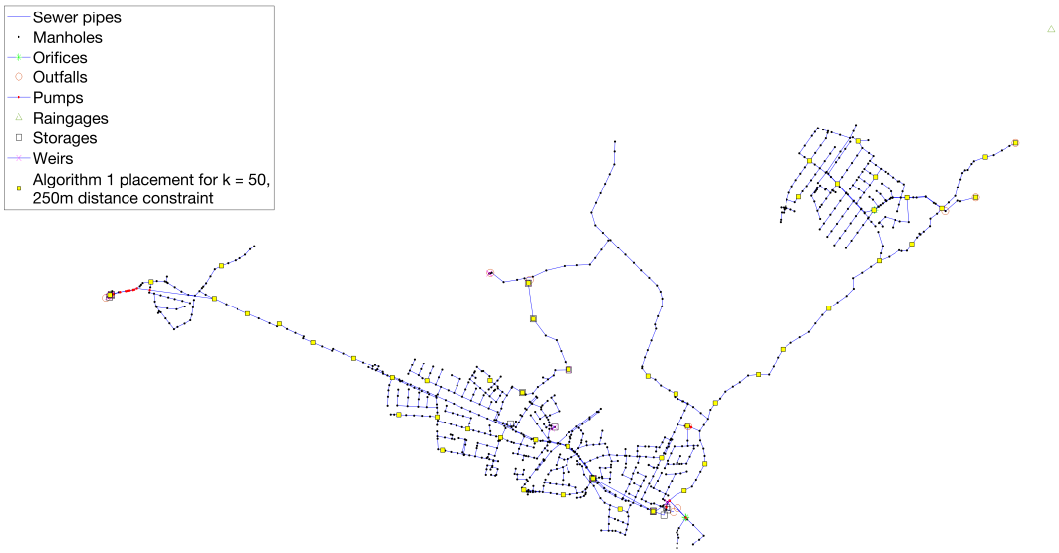

Figure 3: The sensor placement obtained by Algorithm 1 for 50 placed sensors with 250m distance constraints between chosen nodes.

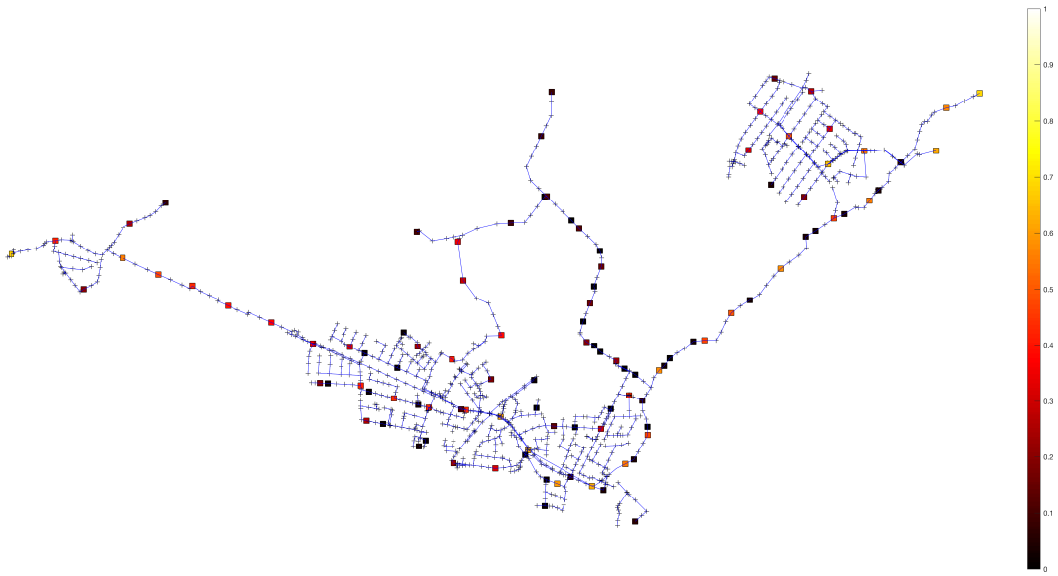

Figure 4: Likelihood of sensor placement for 70 sensor placements from Algorithm 1 using volumetric flow data with 250m distance constraints.

We can see in Figure 4 that the likelihood of each node is significantly lower than that of those in Figure 1, which indicates that the sensor placements vary for different k sensor mutual information maximization realizations. The above figures show that for both the 150m and 250m distance-constrained cases, the sensor placement is more spread across the branches in the network when compared to the unconstrained case.

Interestingly, while the constrained sensor placements do indeed provide a better geographical coverage of the network, the estimation performance degrades with respect to the unconstrained case. In fact, the constrained estimation error is approximately doubled for the general linear model (GLM) for both cases and the general regression neural network (GRNN) estimates degrade slightly (13% degradation) for the case with $k = 25$ and significantly (42 % degradation) for the case with $k = 50$. This observation further confirms that mutual information is indeed an appropriate metric to guide the sensor placement problem as it captures the evidence that the data contains.

Number of sensors	25		50	
Estimation method	GLM NMSE	GRNN NMSE	GLM NMSE	GRNN NMSE
Algorithm 1	0.0069	0.0343	0.0032	0.0284
Rule based	0.0300	0.0604	0.0196	0.0570
Algorithm 1 with 150m distance constraint	0.0152	0.0387	0.0133	0.0403
Algorithm 1 with 250m distance constraint	0.0158	0.0391	0.0148	0.0494

Table 1: Normalized mean square error results for $k = 25$ and $k = 50$ for each of the new sensor placement methodologies compared alongside Algorithm 1 and Rule based from the manuscript.