Impact Driven Sensor Placement for Leak Detection in Community Water Networks

Praveen Venkateswaran*, Qing Han*, Ronald T. Eguchi‡, Nalini Venkatasubramanian*
* University of California, Irvine, ‡ ImageCat Inc., CA, US.

***Abstract*—Community water networks have become increasingly prone to failures due to aging infrastructure, resulting in an increased effort to instrument and monitor networks using IoT (Internet of Things) sensors. However, identifying optimal locations to instrument these sensors to detect and localize failures such as leaks is challenging due to the growing scale and complexity of water networks. Current sensor placement algorithms use heuristics that focus mainly on enabling network coverage. In this paper, we propose a multilevel approach to model and quantify the real-world impact of a failure on a community using various geospatial, infrastructural and societal factors. We present techniques to integrate failure impact, IoT sensing data, and simulation based analytics to drive two novel sensor placement algorithms with the objective of reducing community-scale impact. We evaluate our proposed algorithms on various failure scenarios using multiple real-world water networks at different scales and compare them to existing solutions. The experimental results show that the proposed algorithms result in sensor placements that can achieve an 80% reduction in impact while using a comparable number of sensors for diverse real-world networks.**

I. Motivation

Community water networks are critical infrastructure responsible for the supply, distribution, and storage of water resources. In addition to the various societal water needs, the presence of a resilient water infrastructure is important to service continuity of other lifelines and domains such as power, sanitation, agriculture, healthcare and manufacturing. Today, with widespread urban development and population growth, community water networks have grown in scale and complexity. This, combined with the fact that water infrastructure is often decades old, has made the networks increasingly vulnerable to failures. Pipe leaks or breaks can result from stress caused by various factors such as corrosion, pipe displacements, extreme weather, disaster events, etc.

Failures in water networks can have significant impact on community infrastructures and processes. They can cause economic losses by damaging property and create clean water shortages in places where there is a critical need. Large pipe leaks, or breaks, typically result in an outflow of water to the surrounding region inconveniencing citizens. They can also cause localized flooding which may result in soil erosion, sink holes, or ponding of water. [1], [2]. This may lead to heavy economic damage as water floods into homes and buildings, or cause mud and debris flows which can affect the functioning of the community by disrupting transportation, civic amenities, educational institutions, local businesses, etc. Urban development and the lack of proper planning can further exacerbate the effects of flooding. For instance, paved streets and roads (because of their impervious surfaces) can increase the volume and speed of flowing water, reduce infiltration of water into the ground, and accelerate runoff to ditches and streams which can result in flooding[3], [4]. This can also lead to health and security issues through the introduction and propagation of contaminants in the network. Pipe leaks are also costly in terms of energy consumption due to the need to pump additional water to service the affected zones[5]. The water seepage from pipe bursts can result in cascading effects and impact other infrastructures such as nearby industries, transportation, and healthcare among others. It is therefore essential to have an effective methodology in place to monitor water networks and quickly identify failures, in particular, those that disrupt and endanger societies at large.

Today, water networks are predominantly monitored to gather usage statistics for billing purposes. Utilities typically take a reactive approach to failures and analyze the metering data to identify differences in the amount of water supplied and consumed to infer the presence of leaks. Also, since the majority of water pipes are below ground, agencies rely on external reports, (e.g.) through human observations, to gain knowledge about failures. This results in delays in initiating the process of leak detection, namely the identification of failure locations and their severity. Common methods used to locate leaks include the use of acoustic listening devices [6][7][8], ground penetrating radar, or even physical inspection[9], [10]. Utilities also complement meter readings with hydraulic simulations to isolate leaks[11], [12]. These approaches are computationally expensive, slow and exhibit low detection accuracy in large-scale water networks with multiple leaks [13]. Research efforts to deploy low cost sensors to monitor water networks include PIPENET[14] in USA, WaterWise[15] in Singapore, and WaterBox[16] in UK. However, the inherent scale of today's water networks means that instrumenting the entire network with these sensors would be prohibitively expensive, thus calling for selective placement solutions.

We argue that an intelligent sensor placement methodology is important for rapid and cost effective identification of failures throughout the network. In this paper, we propose to drive sensor placement decisions in community water networks based on the impact that failures would have on the community at hand. Leveraging prior community knowledge, we specifically focus on pipe leaks (or bursts) in the water distribution network as the key failure mode to design intelligent sensor placement strategies.

In contrast to current placement approaches that are coverage based and treat all leaks uniformly, our approach is based on identifying the needs of the community and the impact of leaks. There are three key observations that drive our impact-driven approach to instrumenting water networks. Firstly, while detecting all leaks is important, **not all leaks are equally impactful**. For instance, a leak affecting a hospital is more critical to identify rapidly as compared to a leak affecting a household. Secondly, leaks affecting the same community may have different outflows and hence **have different severity**. Finally, diverse communities may be **vulnerable to different extents** to leaks based on their location, structure, demographics, built infrastructure, urbanization, etc.

Contributions. The novelty of this work lies in the uniquely structured geosocial approach for sensor placement that is contextualized to community-scale water distribution networks. A novel aspect of our approach is to combine sensing, simulations and geospatial information sources to (a) characterize the unique impact of each leak event and (b) use this to drive sensor placement for the community at hand. Specifically, in this paper we make the following contributions:

- We present a methodology to model community vulnerability that takes into account various geophysical, societal, demographical and topological factors. (Section II)
- We propose a novel approach to incorporate these geosocial correlations to characterize impact into the sensor placement decision making process. (Section III)
- We present an approach that combines impact, sensing data, and simulation engines developed by domain experts to explore a range of sensor placement choices. (Section III)
- We propose two novel sensor placement algorithms that are driven by impact. (Section IV)
- We perform extensive evaluations of the proposed algorithms using 3 real-world water networks. (Section V)

II. Impact of Pipe Failures: A GeoSocial Approach

In this section, we further develop the notion of community impact due to failures in the water network. We argue that a comprehensive approach to sensor placement must consider a range of socioeconomic and geospatial factors and discuss how this differs from prior approaches to instrument the water infrastructure. Specifically, we discuss the various geophysical, infrastructural, economic and societal factors that should be considered while modeling impact. We categorize these factors as follows:

• **Terrain and Topography:** Terrain is a major factor in determining the direction and speed of water flow. For instance, the flow of water would be faster down a slope than along a flat terrain. Also, the flow of water from a leak at a higher elevation can affect regions that are downstream. Hence, modeling the terrain elevation and gradient is important to determine the regions that would get affected in the event of a leak.

• **Leak Characteristics and Network Structure:** Modeling the outflow of water from a leak or break is essential to determine the extent of flooding. The outflow rate can depend on various factors such as the size of the leak, water pressure, pipe elevation, type of soil surrounding the pipe, etc. The impact of a failure would also depend on the topology of the water network (e.g.) a failure upstream at a central distribution hub would have more impact on the community than a failure at an end node in the network.

• **Population Scale and Demographics:** The societal impact of a leak or break is closely related to the scale of population that it affects. Failure events in a population center would cause a larger disruption to the community than in a region of low population density. It is also important to understand the demographics of the region while modeling impact, since a failure affecting an old-age home or a school could have a high impact. Another factor is system redundancy. If no or little redundancy (alternative water supplies or conduits) exists, then the impact on the affected population would be higher.

• **Economic Impact:** The outflow of water from leaks can disrupt other lifelines and services and cause significant damage to property. Modeling the monetary costs associated with recovery and reconstruction activities can be used to model the impact of a leak (i.e.) disruption of services can affect local businesses, thus causing secondary losses to the community.

• **Cascading Effects:** There is a potential for leaks to cause cascading failures in other high risk infrastructure. The seepage of water into nearby chemical or other industrial plants can result in additional damage that could be exponentially more than the damage from flooding alone. An example of this type of cascading failure was observed during Hurricane Harvey this year[17] when a chemical plant in Crosby, Texas lost electrical power from backup generators because of flooding which eventually caused several explosions and ensuing fires.

In addition to modeling the geospatial aspects of failure, sensor placement methods must capture temporal metrics, i.e. detection time while modeling impact, and consequently response to failures. Our goal is to design sensor placement techniques that ensure adequate coverage of high impact regions (spatial aspect) with low detection times (temporal aspect). In the remainder of this paper, we aim to answer the following questions. How do we define and quantify the spatio-temporal factors of impact accurately and meaningfully? How can we use these factors to model the vulnerability of a community? How do we use the notion of impact to drive sensor placement in order to minimize the effects of pipeline failure and disruption? Our approach, (See Fig.1), is to first model the water network and simulate various failure scenarios to capture the detection capabilities of the network. We then model the community structure based on the topography, infrastructure and demographics. We then use these models to determine the vulnerability of the community. We simulate the effects of flooding and utilize this to model the impact caused. We finally use this impact model to drive the placement of sensors.

Related Research: Existing research on instrumenting water networks, has focused on (a) measuring water quality and (b) leak detection. Typical heuristics used to design sensor placement for contaminant detection aim to optimize factors

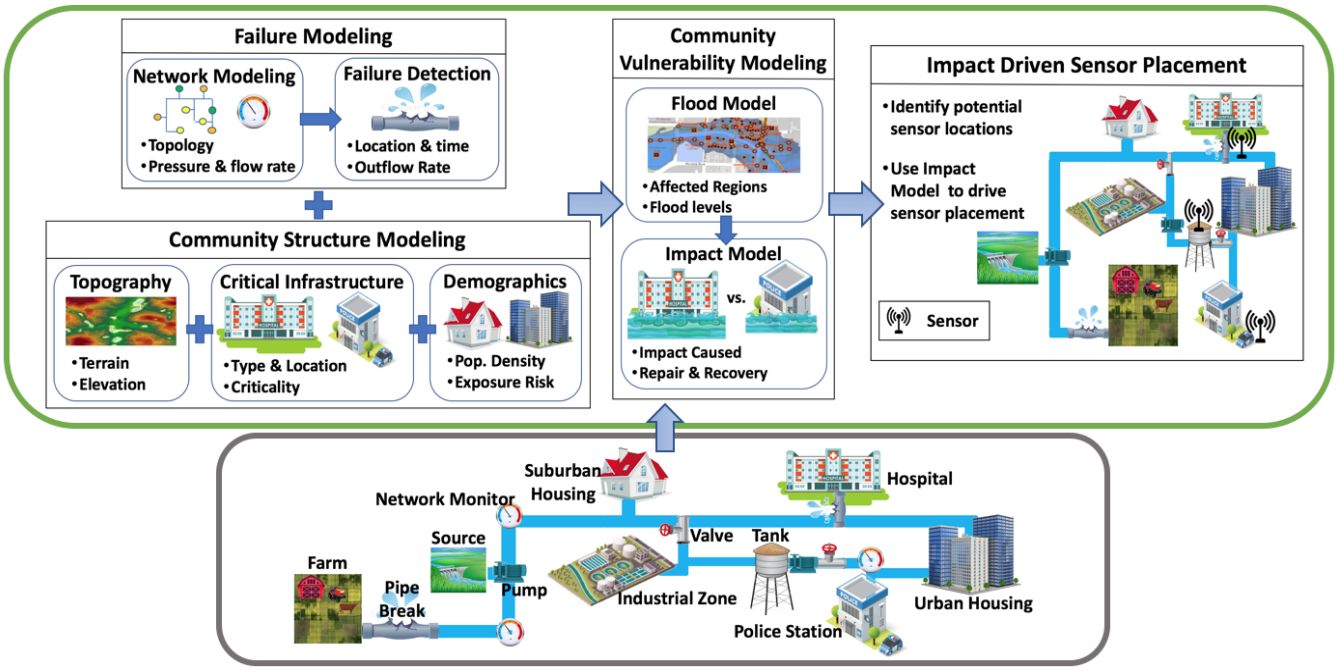


Fig. 1: Solution Workflow for Impact Driven Sensor Placement Architecture

such as detection time, affected population, likelihood of detection, etc. These include graph theoretic approaches[18], [19] that utilize shortest path and set cover variants as well as optimization methods using mixed integer programming [20], [21] that typically work for small networks. Challenges such as the Battle of Water Sensor Networks (BWSN) [22] have resulted in more efficient approximation methods [23] that scale to larger networks. Typical leak detection techniques use network coverage as the prime objective to design placement heuristics. Deterministic methods like Branch-and-Bound can guarantee optimal solutions[24] for limited scale. Techniques utilizing genetic algorithms can scale [25], but have long runtimes and are hard to tune. Approximate solutions include the Greedy approach [26] which is not effective in localizing leaks and the recent Augmented Greedy approach [27].

In our recent work [28], we designed and built a cyber-physical-human middleware, AquaSCALE, for gathering, analyzing and localizing failures in community water services. In contrast to previous approaches that addressed single leak failures, [10], [29], [30], we developed solutions to the multiple-leak localization problem, by leveraging dynamic data from multiple information sources (e.g. IoT devices, weather, social media reports). Since the focus of [28] is to address the multiple-leak localization problem, the placement techniques are simple and limited to a clustering based approach that does not take leak impact into account. In contrast, this paper provides a more in-depth, holistic and practical solution to the sensor placement problem by systematically utilizing multiple geo-social factors (e.g. hospitals and schools are critical infrastructures and need better instrumentation). Modeling the various influencing factors to represent "impact" comprehensively is the topic of the next section.

III. Modeling Community Impact

In this section, we provide a detailed formulation of our proposed methodology for modeling the factors influencing impact. We denote the set of potential sensor locations as $\mathcal{S} = \{s_1, s_2, ..., s_n\}$ and the set of potential failure (i.e., pipe leak) locations as $\mathcal{L} = \{l_1, l_2, ..., l_n\}$ where s_i refers to a sensor placed at location i and l_j refers to a leak at location j. Without loss of generality, we assume that leaks occur at pipe junctions and that all junctions can be instrumented with sensors (i.e) $|\mathcal{S}| = |\mathcal{L}|$. This can easily be extended to cases where leaks occur in both junctions and pipes, as well as cases where sensors can be placed only in certain locations by increasing or decreasing $\mathcal{S}$ and $\mathcal{L}$. We also use a sample water network (Fig. 2(a)) with 7 junctions as a running example to illustrate our approach. The notations used in this paper have been summarized in Appendix A.

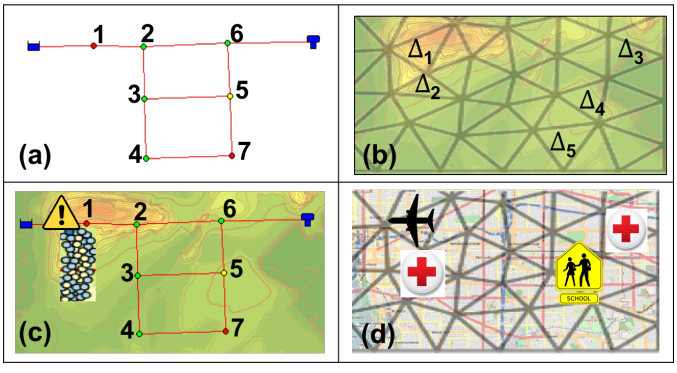


Fig. 2: Running Example - (a) sample water network, (b) terrain and triangular grid, (c) outflow from leak at junction 1 and (d) critical infrastructure locations.

A. Community Structure Modeling

Modeling the structure and vulnerability of a community is important to accurately determine the impact of leaks. Since leaks can affect different parts of the community, we partition the community into regions and model the relative criticality of these regions. In practice, the partitioning of regions is typically driven by flood maps which depend on the terrain. We build a grid by dividing the terrain of the community area into triangular regions using a delaunay triangulator, Triangle [31]. We use triangular rather than rectangular grids because they allow for localized grid refinement and easily conform to terrains with irregular shapes [32]. We denote the set of triangular regions as $\Delta = \{\triangle_1, \triangle_2, ..., \triangle_n\}$. We then build an elevation map for the area based on the elevation of the vertices of the triangular regions in Δ. Fig. 2(b) shows the triangular regions and the terrain elevation map for the sample water network.

We then identify the boundary coordinates of each of these regions, and determine their relative criticality based on the population density as well as the critical infrastructure present within the region. Some categories of critical infrastructure that we have considered include healthcare, transportation, government facilities, educational, etc. We assign relative importance scores to each of these categories and compute the sum of the scores of the infrastructure present within each region.

Definition.*(Region Criticality)* We denote the criticality of a region $\triangle_k$ by $\eta(\triangle_k)$ and compute it as -

$$\eta(\triangle_k) = \lambda_1 Pop(\triangle_k) + \lambda_2 Infra(\triangle_k) \tag{1}$$

where $Pop(\triangle_k)$ is the normalized population density and $Infra(\triangle_k)$ is the normalized sum of infrastructure scores of the region $\triangle_k$ and λ_1, λ_2 are weights that can be adjusted to reflect the relative importance of population density and infrastructure.

Example: From Fig. 2(d) we can see the locations of some critical infrastructure (hospital, school, airport) to which we assign scores of $\{10, 7, 9\}$ respectively. For this example we assume that the populations of all the regions are the same. We can hence determine $\eta(\Delta)$ for the 5 regions ($\triangle$) in Fig. 2(b) as $\{0.9, 1, 1, 0.7, 0\}$ respectively.

B. Water Network and Failure Modeling

A community water network can be represented by a graph $G(V, E)$, where the vertices V represent nodes and pipe junctions, and the edges E represent links (pipes, valves, and pumps). Pipe leaks cause a disturbance in the flow of water in the network, which moves through the system as a pressure wave with high velocity [33]. During a leak, the cross-sectional area of the leak orifice increases. From Bernoulli's equation, we get that $\frac{p}{\rho} + \frac{v^2}{2} + gz = const$, where p is the pressure, ρ the water density, g the acceleration due to gravity, z the elevation, and v the velocity of water which is inversely proportional to the cross-sectional area. This means that since the cross-sectional area of the leak orifice is lesser than that of the pipe, the leak velocity increases. From the equation, since velocity is inversely proportional to pressure, this increase in velocity results in a pressure drop. This implies that pipe bursts can be identified by detecting changes in the hydraulic pressure [27].

Definition.*(Outflow Rate)* The outflow rate of water from a leak can then be computed as -

$$Q = E_c \times p^{\beta} \tag{2}$$

where Q is the outflow rate from the leak, E_c is the effective leak area of the orifice, p is the pressure head at the leak and β is a constant [34].

C. Failure Sensitivity Modeling

Since the disturbance caused by a pipe leak dissipates with distance, the associated pressure change may manifest itself at only a certain number of junctions in the network. There is also a time delay associated with this pressure change that increases with distance from the leak. Hence, the placement of a sensor would be sensitive to a subset of leak events $L \subseteq \mathcal{L}$ within a certain time period. Our goal is to determine whether a sensor at location s_i can detect a leak at l_j, and if so, the corresponding detection time. In order to model leak sensitivity, we run simulations on a hydraulic simulator EPANET [35] which simulates the hydraulic behavior within pressurized water distribution pipe networks. Similar to [28], we enhance EPANET to support the placement of IoT devices, introduction of leaks, and the measurement of time taken for leak propagation. We introduce leaks by adding emitter devices which model the outflow of water through an orifice using (2) and measure leak propagation time using the structural properties of the input water network, namely pipe lengths and the flow rates.

Using EPANET, we first instrument all the potential sensor locations ($\mathcal{S}$) and introduce a leak in each of the potential leak locations ($\mathcal{L}$) one at a time. For each introduced leak $l_j \in \mathcal{L}$, we monitor the pressure values at every sensor location $s_i \in \mathcal{S}$ to determine if s_i can detect l_j within the simulation time period. As in [36] we build a *detection capability matrix* $\mathcal{M}_{dc}$ where the rows represent the potential sensor locations and the columns represent the potential leak locations. The entries of the matrix are binary valued (0 or 1) depending on whether a given leak results in a pressure change at a sensor. We denote the observed pressure values over time at location s_i as p_i under normal conditions, as $\hat{p}_i$ after the leak is introduced, and set a detection threshold ϵ. Then the value of the matrix at location s_i for leak l_j can be determined as:

$$\mathcal{M}_{dc}[s_i, l_j] = \begin{cases} 1, & \text{if } p_i - \hat{p}_i \geq \epsilon \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

In addition to the detection capability matrix, we build the corresponding *detection time matrix* $\mathcal{M}_{dt}$ which reflects the time taken for a sensor location to detect a leak. The time value is bounded by the specified simulation time period. In the case where the sensor is incapable of detecting the leak, we set the value of the corresponding entry to be infinity. Hence, the entries of the detection time matrix can be computed as -

$$\mathcal{M}_{dc} = \begin{array}{c} \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{array} \begin{array}{c} \begin{array}{ccccccc} l_1 & l_2 & l_3 & l_4 & l_5 & l_6 & l_7 \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 1 \end{pmatrix} \end{array} \quad \mathcal{M}_{dt} = \begin{array}{c} \\ s_1 \\ s_2 \\ s_3 \\ s_4 \\ s_5 \\ s_6 \\ s_7 \end{array} \begin{array}{c} \begin{array}{ccccccc} l_1 & l_2 & l_3 & l_4 & l_5 & l_6 & l_7 \end{array} \\ \begin{pmatrix} 0.2 & \infty & \infty & \infty & \infty & \infty & \infty \\ 7.2 & 0.3 & \infty & \infty & \infty & \infty & \infty \\ 13 & 5.7 & 0.3 & \infty & \infty & \infty & \infty \\ 16 & 9 & 14 & 0.6 & \infty & \infty & \infty \\ \infty & 8.7 & 8.3 & \infty & 0.4 & 10 & \infty \\ \infty & \infty & \infty & \infty & \infty & 0.2 & \infty \\ \infty & \infty & 12 & 6.1 & 15 & \infty & 0.3 \end{pmatrix} \end{array} \quad \mathcal{M}_{fl} = \begin{array}{c} \\ l_1 \\ l_2 \\ l_3 \\ l_4 \\ l_5 \\ l_6 \\ l_7 \end{array} \begin{array}{c} \begin{array}{ccccc} \triangle_1 & \triangle_2 & \triangle_3 & \triangle_4 & \triangle_5 \end{array} \\ \begin{pmatrix} 3.4 & 2.7 & 0 & 0 & 0 \\ 1.6 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 \\ 0 & 0 & 1.1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2.8 \end{pmatrix} \end{array}$$

Fig. 3: Detection capability, Detection time, and Flood level matrices for Running Example

$$\mathcal{M}_{dt}[s_i, l_j] = \begin{cases} \delta(s_i)_{l_j}, & \text{if } M_{dc}[s_i, l_j] = 1 \\ \infty, & \text{otherwise} \end{cases} \tag{4}$$

where $\delta(s_i)_{l_j}$ is the time taken for sensor s_i to detect l_j measured in minutes. Using $\mathcal{M}_{dc}$ in conjunction with $\mathcal{M}_{dt}$ allows us to localize leaks by using the relative time difference in detection of leak events by sensors using sufficiently high sensing rates.

Example: For the sample network in Fig. 2(a), we introduce a leak at each of the 7 junctions one at a time. We run the simulation on EPANET and obtain the detection capability and detection time matrices ($\mathcal{M}_{dc}$ and $\mathcal{M}_{dt}$ in Fig. 3). For each leak, the differing detection time values reflect the time delay in propagation of the leak event.

D. Modeling Flood Levels

Given a set of failures, their impact on the community is dependent on the water outflow and seepage from the leaks. In order to model the vulnerability of a community to flooding, we simulate the outflow of water from a leak as well as its propagation along the surrounding terrain using a hydrodynamic flood simulation algorithm BreZo [37].

For each leak $l_j \in \mathcal{L}$ introduced earlier, we compute its outflow rate using Equation (2) and then run the BreZo simulation, providing as input the terrain elevation, triangular grids, leak location and the outflow rate. The simulation returns the regions affected by flooding and their corresponding flood levels. We use this to build a *flood level matrix* $\mathcal{M}_{fl}$ with the potential leak locations as the rows and the triangular grids as the columns. $\mathcal{M}_{fl}[l_j, \triangle_k]$ thus reflects the flood levels in region $\triangle_k$ due to a leak at location l_j. The entries of the flood level matrix can be computed as -

$$\mathcal{M}_{fl}[l_j, \triangle_k] = \begin{cases} 0, & \text{if leak at } l_j \text{ does not impact } \triangle_k \\ H(l_j)_{\triangle_k}, & \text{otherwise} \end{cases} \tag{5}$$

where $H(l_j)_{\triangle_k}$ denotes the maximum flood level at $\triangle_k$ due to a leak at l_j. Fig. 2(c) shows the simulation of water outflow from a leak at junction 1 along the terrain map.

Example: We simulate the outflow of water from a leak at each junction in the sample water network (Fig.2(a)). We obtain the following flood level matrix (values in ft) where we restrict the columns to only the triangular regions marked in Fig.2(b) ($\mathcal{M}_{fl}$ in Fig. 3). The large impact of l_1 on its immediate surroundings is reflected by the higher flood levels in $\triangle_1$ and $\triangle_2$.

E. Impact Modeling of Failures

The notion of impact of leaks, which we denote as $\mathcal{I}_{\mathcal{L}}$, can be thought of from two viewpoints -

1) **Nodal Impact:** The impact that a particular leaking node/junction has on the surrounding regions. For a given leak l_j, this involves identifying the regions $\triangle_k \in \Delta$ such that $\mathcal{M}_{fl}[l_j, \triangle_k] > 0$ and then obtaining the criticality of those regions ($\eta(\triangle_k)$).
2) **Regional Impact:** The impact caused to a particular region by leaks at different nodes. For a given region $\triangle_k$, we determine the leak locations $l_j \in \mathcal{L}$ such that $\mathcal{M}_{fl}[l_j, \triangle_k] > 0$ and determine the levels of flooding.

In both these cases, we see that $\mathcal{I}_{\mathcal{L}}$ depends on the criticality of the regions and the corresponding flood levels. In the next section we present two approaches to instrument sensors that are driven by each of the above viewpoints.

IV. Sensor Placement : Problem and Algorithms

The effectiveness or utility of a sensor placement depends on the impact resulting from the time taken to detect leaks by the sensors in the placement. As discussed in Section II, a good placement is one that can quickly detect leaks that have high impact. We can define this as :

Definition.*(Overall Utility)* Given a sensor placement set $\mathcal{P} \subseteq \mathcal{S}$ of sensor locations, we define its overall utility $\mathcal{U}(\mathcal{P})$ as $\mathcal{I}_{\mathcal{L}}/min(\mathcal{M}_{dt}[\mathcal{P}, \mathcal{L}])$, where $min(\mathcal{M}_{dt}[\mathcal{P}, \mathcal{L}])$ is the shortest detection time of leaks $l_j \in \mathcal{L}$ by the sensors $s_i \in \mathcal{P}$.

For each sensor $s_i \in \mathcal{S}$, we define $C_i \subseteq \mathcal{L}$ as the set of leaks covered by sensor s_i and C_{cov} as the set of leaks covered by all the sensors in the placement set (i.e.) $C_{cov} = \bigcup C_i$, $\forall s_i \in \mathcal{P}$. The objective of the *impact-driven placement problem* is to then identify a minimum cardinality sensor placement set $\mathcal{P} \subseteq \mathcal{S}$ that maximizes the overall utility $\mathcal{U}(\mathcal{P})$, while also ensuring that when a leak $l_j \in \mathcal{L}$ occurs, there exists at least one $s_i \in \mathcal{P}$ that can detect l_j

When there exist leaks that cannot be detected by any of the potential sensor locations, a maximum coverage (detecting all leaks) would not be possible and hence a maximal coverage is defined to be the set of all detectable leaks. The simplified version of the impact-driven placement problem is when the impact of all leaks is the same which is equivalent to the *maximum coverage problem* that can be defined as -

Definition.*(Maximum Coverage Problem)* Given a number $|\mathcal{S}|$ and the collection of sets $\mathcal{C}=\{C_1, C_2, ..., C_n\}$, the maximum coverage is to find a subset $\mathcal{C}' \subseteq \mathcal{C}$ such that $|\mathcal{C}'| \leq |\mathcal{S}|$ and $|\bigcup_{C_j \in \mathcal{C}'} C_j|$ is maximized.

In this definition, if $\mathcal{S}$ is the set of potential sensor locations and each C_i is the set of leaks covered by s_i, finding C_{cov}=$\mathcal{L}$ is a solution to the *MCP* and hence shows the equivalence. *MCP* is NP-hard[38] and hence finding a solution to the impact driven placement problem is also computationally complex. In this section, we describe two novel approximate solutions driven by the Nodal Impact and Regional Impact viewpoints.

A. Max Nodal Impact

In this algorithm, we first determine the utility of adding a sensor to the existing sensor placement set $\mathcal{P}$. We then select, in each iteration, the sensor location with the largest utility, until all the detectable leak locations have been covered. The utility of adding a sensor to the existing placement depends on how quickly it can detect leaks and the resulting impact in terms of flood levels and criticality of the affected regions.

Definition. *(Leak Utility)* We denote the utility of placing a sensor at s_i for a given leak l_j as its leak utility $U_{l_j}(s_i)$. The overall utility of sensor s_i, $\mathcal{U}(s_i)$, is then the sum of all its leak utilities which we compute as follows -

1) We use the detection capability matrix $\mathcal{M}_{dc}$ to determine the set of leaks that s_i can detect. For each of these leaks, we use the detection time matrix $\mathcal{M}_{dt}$ to determine whether s_i can detect the leak quicker than the sensors in the existing placement set since the impact of a leak depends on the earliest time of detection.
2) For each of the leaks that s_i detects faster, we determine the criticality of the regions that the leak affects ($\eta(\Delta)$) as well as the corresponding flood level using $\mathcal{M}_{fl}$. We then compute the impact of leak l_j as $\mathcal{I}_{l_j}$=$\eta(\triangle_k) \times \mathcal{M}_{fl}[l_j, \triangle_k]$, where k represents the affected regions.
3) To compute the leak utility of s_i for l_j, we divide the leak impact by the corresponding time taken by s_i to detect the leak (i.e.) $U_{l_j}(s_i)$=$\mathcal{I}_{l_j}$ / $\mathcal{M}_{dt}[s_i, l_j]$. Hence a sensor capable of detecting a high impact leak quickly would have a larger utility.
4) We finally compute the overall utility of sensor s_i as the sum of its individual leak utilities. $\mathcal{U}(s_i)$=$\Sigma U_{l_j}(s_i)$.

As described in Algorithm 1, in a given iteration, the sensor location with the maximum utility score (s_v) is added to the placement set $\mathcal{P}$ and the set of covered leaks is updated to $C_{cov} \cup C_v$. We recompute the scores for each iteration until all detectable leaks are covered (C_{cov}=$\mathcal{L}$). Since every detectable leak has at least one sensor that covers it, the algorithm is guaranteed to complete. The running time of the algorithm is a function of the number of sensors, leak locations, and regions $\mathcal{O}(|\mathcal{S}||\mathcal{L}||\Delta|)$, which would result in long runtimes for large scale networks with numerous junctions and pipes. However, we can use the property of *submodularity* to significantly reduce the number of function evaluations thus reducing runtime [39], [40].

Definition. *(Submodularity)* A set function F on a finite set N is submodular if for all subsets $\mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{S}$ and elements $s \in \mathcal{S}$, it satisfies $F(\mathcal{P} \cup s) - F(\mathcal{P}) \geq F(\mathcal{Q} \cup s) - F(\mathcal{Q})$.

Hence the marginal utility of adding s to the smaller set $\mathcal{P}$ is greater than adding it to $\mathcal{Q}$. In the context of leak detection, this means that as the coverage of leaks by the sensors in the placement set increases, the marginal utility of adding a sensor decreases. To utilize submodularity, we note that the leak utility of a sensor placement is a function of the leak utilities of its sensor locations ($U_{l_j}(\mathcal{P}) = max(U_{l_j}(s_i)), s_i \in \mathcal{P}$). Also, for a given leak, the relative utility of two sensor placements depends on how fast they detect it. Hence the leak utility function has the following properties - (1) it is non-negative ($U_{l_j}(\mathcal{P}) \geq 0, \forall \mathcal{P}$), (2) it is non-decreasing (i.e.) for placements $\mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{S}$, $U_{l_j}(\mathcal{P}) \leq U_{l_j}(\mathcal{Q})$. We can then show that -

Algorithm 1 Max Nodal Impact (MNI)

Input: $\mathcal{S}, \mathcal{L}, \Delta, \eta, \mathcal{M}_{dc}, \mathcal{M}_{dt}, \mathcal{M}_{fl}, \{C_1, C_2, ..., C_n\}$
Output: $\mathcal{P} \subseteq \mathcal{S}$
Initial Conditions: $\mathcal{P} = \emptyset, \mathcal{C}_{cov} = \emptyset$
while $C_{cov} \neq \mathcal{L}$ **do**
  for $i = 1 \rightarrow |\mathcal{S}|$ **do**
    if $s_i \in \mathcal{P}$ **then** continue **end if**
    for $k = 1 \rightarrow |C_i|$ **do**
      $U_{l_k}(s_i)$=0
      if $\mathcal{M}_{dc}[s_i, l_k]$=1 & $\mathcal{M}_{dt}[s_i, l_k]$<min($\mathcal{M}_{dt}[\mathcal{P}, l_k]$) **then**
        for $j = 1 \rightarrow |\Delta|$ **do**
          $\mathcal{I}_{l_k} = \eta(\triangle_j) \times \mathcal{M}_{fl}[l_k, \triangle_j]$
          $U_{l_k}(s_i)$ += $\mathcal{I}_{l_k}$ / $\mathcal{M}_{dt}[s_i, l_k]$
        end for
      end if
    end for
    $\mathcal{U}(s_i)$=$\Sigma U_{l_k}(s_i)$
  end for
  $s_v = \{s_i : max(\mathcal{U}(s_i)), \forall i : 1 \rightarrow |\mathcal{S}|\}$
  $\mathcal{P} \leftarrow \mathcal{P} \cup s_v$, $C_{cov} \leftarrow C_{cov} \cup C_v$
end while

Theorem 1. *The leak utility of a sensor placement $U_{l_j}(\mathcal{P})$ is submodular.*

Proof - Let $\mathcal{P}$ and $\mathcal{Q}$ be two sensor placement sets such that $\mathcal{P} \subseteq \mathcal{Q} \subseteq \mathcal{S}$. Consider a leak $l_j \in \mathcal{L}$ that can be detected by sensor $s_i \in \mathcal{S} \setminus \mathcal{Q}$. Depending on the detection time, there are three cases -

(1) $\mathcal{M}_{dt}[s_i, l_j] \geq min(\mathcal{M}_{dt}[\mathcal{P}, l_j])$: From this we get that $min(\mathcal{M}_{dt}[\mathcal{P} \cup \{s_i\}, l_j]) = min(\mathcal{M}_{dt}[\mathcal{P}, l_j])$ and $min(\mathcal{M}_{dt}[\mathcal{Q} \cup \{s_i\}, l_j]) = min(\mathcal{M}_{dt}[\mathcal{Q}, l_j])$ and hence $U_{l_j}(\mathcal{P} \cup \{s_i\}) - U_{l_j}(\mathcal{P}) = U_{l_j}(\mathcal{Q} \cup \{s_i\}) - U_{l_j}(\mathcal{Q}) = 0$.

(2) $min(\mathcal{M}_{dt}[\mathcal{Q}, l_j]) \leq \mathcal{M}_{dt}[s_i, l_j] < min(\mathcal{M}_{dt}[\mathcal{P}, l_j])$: This implies $U_{l_j}(\mathcal{Q} \cup \{s_i\}) = U_{l_j}(\mathcal{Q})$ and hence $U_{l_j}(\mathcal{P} \cup \{s_i\}) - U_{l_j}(\mathcal{P}) \geq U_{l_j}(\mathcal{Q} \cup \{s_i\}) - U_{l_j}(\mathcal{Q})$.

(3) $\mathcal{M}_{dt}[s_i, l_j] < min(\mathcal{M}_{dt}[\mathcal{Q}, l_j])$: Here, $U_{l_j}(\mathcal{P} \cup \{s_i\}) \geq U_{l_j}(\mathcal{Q} \cup \{s_i\})$ and $U_{l_j}(\mathcal{P}) \leq U_{l_j}(\mathcal{Q})$ due to the non-decreasing property of U_{l_j}. Hence, we get $U_{l_j}(\mathcal{P} \cup \{s_i\}) - U_{l_j}(\mathcal{P}) \geq U_{l_j}(\mathcal{Q} \cup \{s_i\}) - U_{l_j}(\mathcal{Q})$.

Since the overall utility is a summation of the leak utilities, it is also submodular. As with the maximum coverage problem, our greedy approach gives an approximation ratio of $(1-1/e)$[40]. Then similar to [27], if in a given iteration of the algorithm, $\mathcal{U}(s_1) \geq \mathcal{U}(s_2) \geq \mathcal{U}(s_3) \geq ...$, then s_1 would be added to the placement. Then in the next iteration if $\mathcal{U}(s_2) \geq \mathcal{U}(s_3)$, we can conclude that $\mathcal{U}(s_2) \geq \mathcal{U}(s_i), \forall i \geq 3$,

thus reducing the number of evaluations which ensures the scalability of the algorithm.

B. Max Regional Impact

Algorithm 2 Max Regional Impact (MRI)

Input: $\mathcal{S}, \mathcal{L}, \Delta, \eta, \mathcal{M}_{dc}, \mathcal{M}_{dt}, \mathcal{M}_{fl}, \{C_1, C_2, ..., C_n\}$
Output: $\mathcal{P} \subseteq \mathcal{S}$
Initial Conditions: $\mathcal{P}$=$\emptyset$, C_{cov}=$\emptyset$, $\mathcal{L}'$=$\mathcal{L}$
$\mathcal{Q} = descendingSort(\eta(\triangle_k))$, $\forall \triangle_k \in \Delta$
while $C_{cov} \neq \mathcal{L}$ **do**
  $\triangle_i = \mathcal{Q}.pop()$
  l_v=$\{l_j : max(\mathcal{M}_{fl}[l_j, \triangle_i])$, $\forall j : 1 \rightarrow |\mathcal{L}'|\}$
  s_v=$\{s_k : max(U_{l_v}(s_k))$, $\forall k : 1 \rightarrow |\mathcal{S}|\}$
  $\mathcal{P} \leftarrow \mathcal{P} \cup s_v$, $C_{cov} \leftarrow C_{cov} \cup C_v$, $\mathcal{L}' \leftarrow \mathcal{L}' \setminus l_v$, $\mathcal{Q}.push(\triangle_i)$
end while

In this algorithm, we prioritize the placement of sensors to ensure that the impact on high criticality regions is minimized. We iterate over regions ordered by their criticality and in each iteration, we identify the leak location having the largest impact on the region and then select the sensor location with the largest utility for this leak. We again denote C_{cov} as the set of leaks covered by sensors in the existing placement $\mathcal{P}$ to which we add sensor locations as follows -

1) We first order the set of regions, Δ, in descending order of their criticality score $\eta(\Delta)$. We push the ordered regions into a queue $\mathcal{Q}$ such that the first element of the queue is the region with the highest criticality score.
2) We pop the topmost region of the queue, $\triangle_k$, and identify the leak location l_v which impacts it the most using $\mathcal{M}_{fl}$.
3) We then identify the sensor location s_v with the largest leak utility for l_v (i.e.) $max(U_{l_v}(s_i))$, $\forall s_i \in \mathcal{S}$, where $U_{l_v}(s_i)$=$\mathcal{I}_{l_v}/\mathcal{M}_{dt}[s_i, l_j]$. We add s_v to $\mathcal{P}$ and update the set of covered leak locations as $C_{cov} \cup C_i$ and remove l_v from subsequent iterations.
4) We push $\triangle_k$ to the back of the queue and repeat this process till all the leak locations are covered ($C_{cov} = \mathcal{L}$). This ensures that whenever a region is popped, at least one detectable leak location gets covered, thus guaranteeing that the algorithm will complete.

In order to ensure the efficiency of the algorithm described in Algorithm 2, for each region, we pre-order the leak locations based on their impact and the sensors based on their detection time for each leak. This ensures that the operations to identify the leak with the largest impact and the sensor with the largest leak utility are both constant. Though unlikely in practice, the worst case scenario is when each sensor can detect at most one leak, thus leading to a time complexity of $\mathcal{O}(|\mathcal{L}|)$.

V. Experimental Study

In this section, we test our proposed algorithms on three real-world water networks and compare their performance to two existing coverage-based approaches - Greedy[26] which selects in each iteration, the sensor with the maximum coverage, and Augmented Greedy[27] which iteratively selects sensors by scoring them on their ability to distinguish between leak events. We use the following networks - (1) a subzone of the Washington Suburban Sanitary Commission's (WSSC) water service area in Montgomery County, Maryland [28], (2) a model for the Wolf-Cordera Ranch (WCR) in Colorado Springs, Colorado, and (3) the Richmond water distribution system, part of the Yorkshire water supply area in the U.K. The data for (1) was obtained from WSSC while (2) and (3) from [41]. The layouts of the networks are shown in Fig. 4.

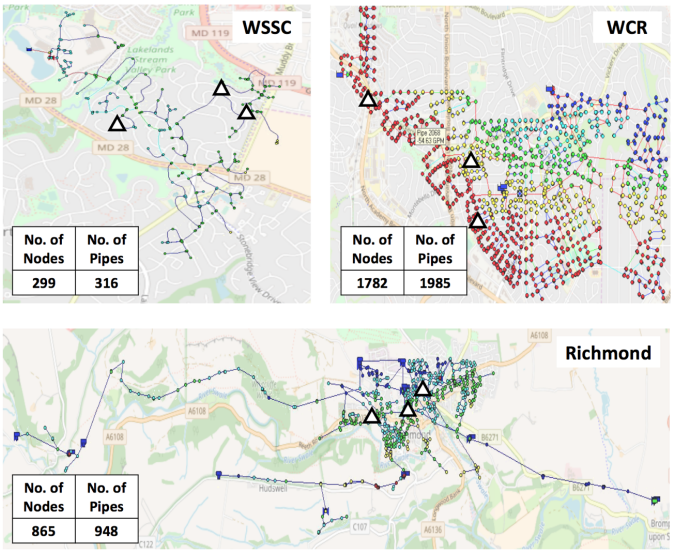


Fig. 4: Partial water network layouts of Washington DC (WSSC), Colorado (WCR) and Richmond (Richmond), with their bounding boxes and critical regions ($\triangle$)

A. Methodology to Model Impact

In order to model the impact of leaks to drive sensor placement, for each network we first determine a bounding box of the area they service. Due to privacy and security reasons, water agencies typically mask such information and hence we make an approximate determination. We then build the terrain elevation map for these areas using elevation data from[42], [43] and obtain population density information using census data from [44]. We extract the coordinates of critical infrastructure within the bounding box using the OpenStreetMap service [45]. We use this obtained population density and location of critical infrastructure to model impact as described in Section III. We show the bounding box and the top three critical regions for each of the networks in Fig. 4.

B. Sensor Placement Performance Comparison

We obtain the sensor placements determined by the four algorithms - Greedy(G), Augmented Greedy(AG), Max Nodal Impact(MNI), and Max Regional Impact(MRI) - for each of the water networks and compare their performance in detecting leaks by simulating the following failure scenarios:

• *Randomized Failures:* We introduce leaks in randomized locations ranging from 5% to 50% of the network's junctions.

	No. of Sensors			Randomized		Critical		Geo-correlated	
Algorithm	WSSC	Richmond	WCR	Avg. Det. Time	Avg. Impact	Avg. Det. Time	Avg. Impact	Avg. Det. Time	Avg. Impact
G	54	221	692	0.86	0.83	0.97	0.78	0.80	0.88
AG	121	321	991	0.12	0.49	0.54	0.53	0.20	0.51
MNI	82	235	740	0.61	0.29	0.48	0.18	0.56	0.25
MRI	117	304	864	0.27	0.09	0.28	0.07	0.13	0.06

Fig. 5: Summary of number of sensors placed and average normalized performance across all networks for the three failure scenarios (10% leak events)

• *Critical Failures:* We introduce leaks in critical locations ranging from the top 5% to 30% of leak locations ordered by impact.

• *Geo-correlated Failures:* We introduce leaks in spatially clustered locations ranging from 5% to 50% of the network's junctions where the number of leaks in each cluster is uniform.

We present the results of the randomized and geo-correlated failure scenarios as the average of multiple runs so as to remove any bias arising from the choice of leak locations. We then evaluate the algorithms using the following metrics:

Average Detection Time: We compare the average leak detection times of the algorithms for the above failure scenarios. For each introduced leak, we determine the quickest time taken for its detection by the sensor locations in each algorithm's placement set and compute the average of these quickest times for all the leaks in the scenario. Fig. 6a shows the comparison of the normalized average detection times. We observe that while the Augmented Greedy approach has low average detection times in general since it uses the most number of sensors, it is outperformed by the Max Regional Impact approach for smaller geo-correlated leaks and by both our proposed approaches in detecting highly critical leaks.

Average Impact Caused: The impact caused by a leak is proportional to the time taken to detect it. We compare the performance of the algorithms based on the average impact caused in each failure scenario. For each introduced leak, we determine its leak impact ($\mathcal{I}_{l_j}$) as described in Section IV-A and multiply it by the corresponding time taken by the sensor placements to detect the leak. We then compute the average impact caused over all the introduced leaks. The normalized average impact caused for the different algorithms are shown in Fig. 6b. We observe that across all the scenarios, the Max Regional Impact approach results in the lowest average impact followed by the Max Nodal Impact, Augmented Greedy and Greedy approaches. We see in particular, that the performance of the Max Regional Impact approach is significantly better for highly critical leaks.

Cost Effectiveness: Since the cost of instrumenting sensors is non-trivial, a good placement of sensors is one that ensures low impact of leak events while being cost effective. We determine the cost effectiveness of a placement as a product of the average impact caused and the number of sensors placed. We show the normalized cost effectiveness of the algorithms in Appendix B and observe that the cost effectiveness of the Greedy and Augmented Greedy approaches are comparable, as are the Max Nodal Impact and Max Regional Impact approaches.

C. Limiting the Number of Sensors

To evaluate the performance when using a limited number of sensors, for each algorithm, we vary the number of sensors from 10% to 100% of its original value. We then determine the leaks covered by the sensors and compare the total impact of these leaks. While the Greedy and Augmented Greedy approaches have more coverage using limited sensors, our proposed approaches cover more high impact leaks (Fig. 7).

D. Max Nodal Impact vs. Max Regional Impact

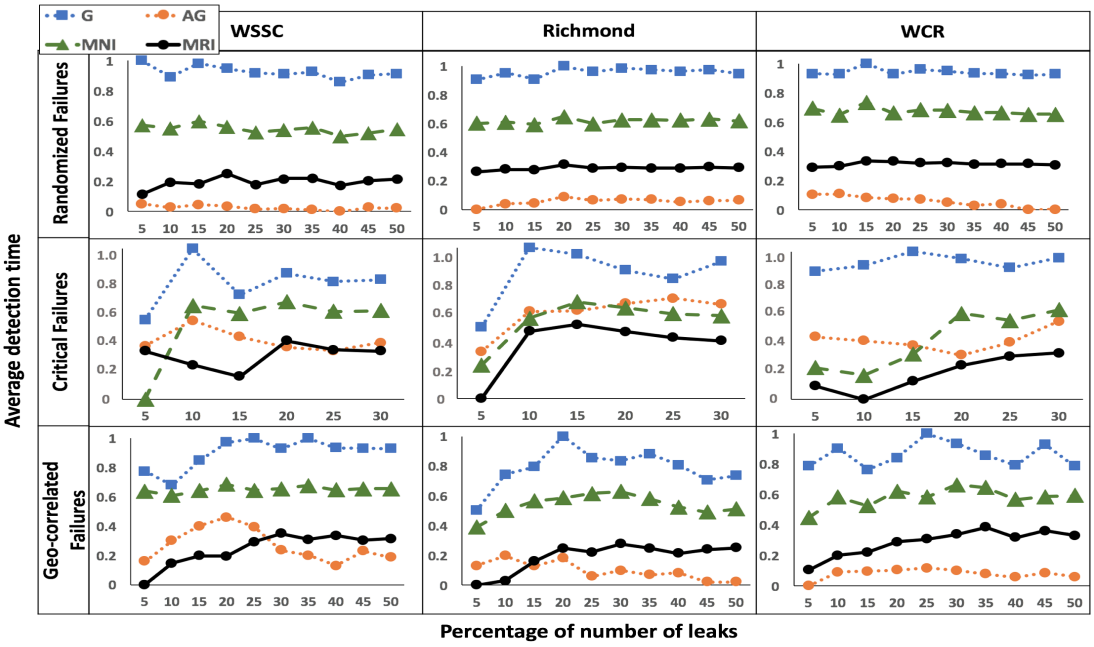
Our earlier experiments show that the better performance of MRI compared to MNI can be attributed to the larger number of sensors it instruments (Fig. 5). In this section, we perform a sensitivity analysis using the WSSC water network to explore the influence of the following factors -

Density of Critical Areas: The vulnerability of a community is closely tied to its most critical regions. Typically, there exist clusters of regions with high criticality such as a population center with infrastructure like hospitals, fire stations, and schools or an industrial zone with multiple critical facilities. We determine the influence of the criticality of these regions as well as their spatial distribution on the performance of the algorithms. For the WSSC network, we initially set the criticality of all regions to 0 and randomly reassign the top 100 original criticality scores while forming spatial clusters, such that the number of regions in the clusters vary from 1 to 10 and vary the criticality from 10% to 100% of the original value. From Fig. 8, we see that the number of sensors placed by the MNI approach increases for low cluster sizes and criticality values. On the other hand, for a given cluster size, the MRI approach returns the same number of sensors, and the number decreases as cluster size increases. The MNI approach is thus useful when there are a small number of large critical clusters.

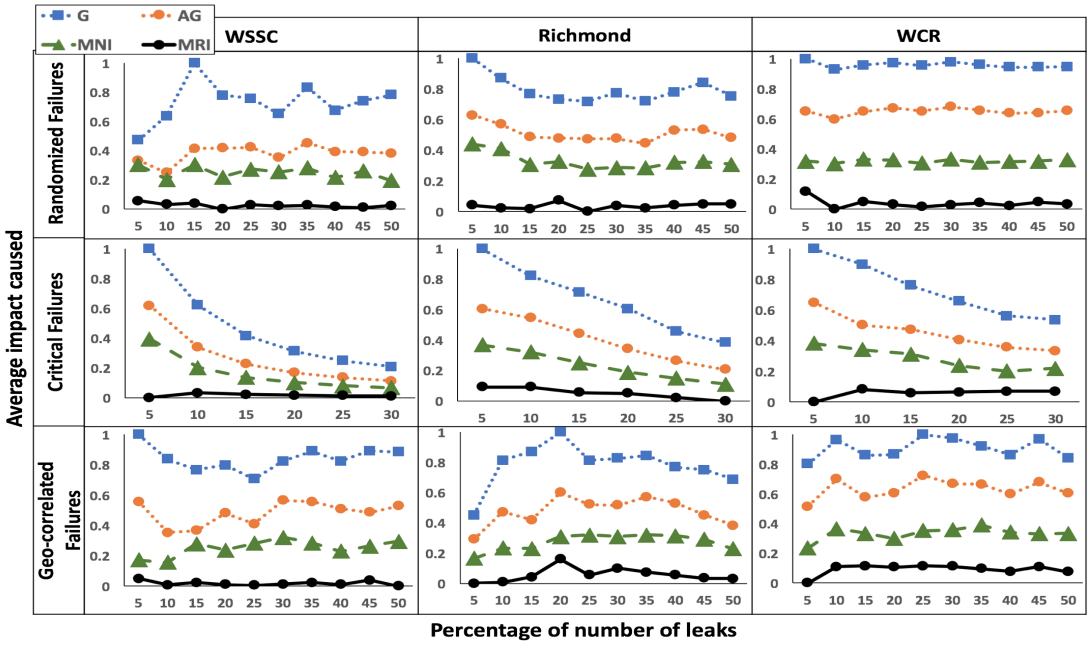
Intensity of Flooding: In order to determine the influence of intensity of leak events on sensor placement, we vary the outflow rate from each of the introduced leaks in Section III-C from 50% to 100%. We then compare the number of sensors placed in each case and determine the corresponding average impact as before. Fig. 9 shows that for low outflow rates, a similar number of sensors are placed by both approaches, thus resulting in comparable average impacts. However, as the outflow rate increases, the MRI approach places a larger number of sensors thus resulting in a lower average impact as compared to the MNI approach. This however means that the MNI approach is useful for limited budgets.

VI. Conclusions and Future Work

In this paper we proposed to drive sensor placement for leak detection based on the notion of impact of a leak on the community. We presented a methodology to characterize and quantify the factors influencing impact, and to incorporate them into the sensor placement decision making process. We presented two novel sensor placement algorithms driven by



(a) Average detection time



(b) Average impact caused by failures

Fig. 6: Comparison of (a) average detection time and (b) average impact caused by failures as a function of number of leaks for randomized, critical, and geo-correlated leak events

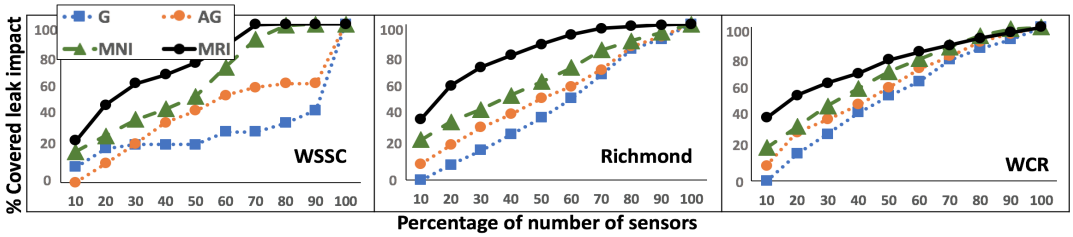


Fig. 7: Total impact of covered leaks as a function of number of sensors

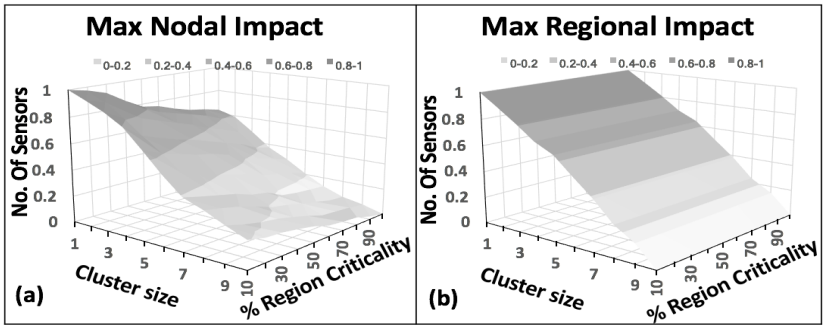


Fig. 8: Number of sensors placed as a function of region criticality and cluster size for (a) Max Nodal Impact and (b) Max Regional Impact approaches.

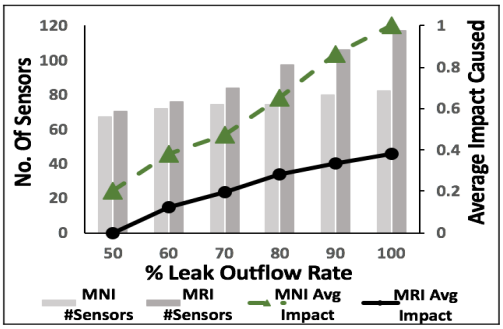


Fig. 9: Number of sensors and average impact caused as a function of the leak outflow rate

impact and evaluated their performance on real-world water networks. As part of our future work, we intend to model the interaction between multiple leaks and to explore other approaches to model the factors influencing impact such as different ways to define regions and their criticality.

REFERENCES

[1] C. Zevenbergen, A. Cashman, N. Evelpidou *et al.*, *Urban flood management*, 2010.

[2] J. A. ten Veldhuis, F. H. Clemens, and P. H. van Gelder, "Quantitative fault tree analysis for urban water infrastructure flooding," *Structure & Infra Eng.*, 2011.

[3] C. P. Konrad, "Effects of urban development on floods," 2003.

[4] B. Barroca, P. Bernardara, J.-M. Mouchel *et al.*, "Indicators for identification of urban flooding vulnerability," *Natural Hazards & Earth System Science*, 2006.

[5] A. F. Colombo and B. W. Karney, "Energy and costs of leaky pipes: Toward comprehensive picture," *WRPM*, 2002.

[6] Y. Gao, M. Brennan, P. Joseph *et al.*, "On the selection of acoustic/vibration sensors for leak detection in plastic water pipes," *Sound & Vibration*, 2005.

[7] J. Rajtar and R. Muthiah, "Pipeline leak detection system for oil and gas flowlines," *Manufacturing Science & Eng.*, 1997.

[8] G. Hessel, W. Schmitt, K. Van der Vorst *et al.*, "A neutral network approach for acoustic leak monitoring in the vver-440 pressure vessel head," *Progress in Nuclear Energy*, 1999.

[9] M. Farley and S. Trow, *Losses in water distribution networks*, 2003.

[10] A. F. Colombo, P. Lee, and B. W. Karney, "A selective literature review of transient-based leak detection methods," *Hydroenvironment*.

[11] A. Nasir, B.-H. Soong, and S. Ramachandran, "Framework of wsn based human centric cyber physical in-pipe water monitoring system," in *Control Auto. Robotics & Vision*, 2010.

[12] Z. Poulakis, D. Valougeorgis, and C. Papadimitriou, "Leakage detection in water pipe networks using a bayesian probabilistic framework," *Prob. Eng. Mechanics*, 2003.

[13] H. Fuchs and R. Riehle, "Ten years of experience with leak detection by acoustic signal analysis," *Applied acoustics*, 1991.

[14] I. Stoianov, L. Nachman, S. Madden *et al.*, "Pipenet: A wireless sensor network for pipeline monitoring," in *Sensor Networks*, 2007.

[15] A. J. Whittle, L. Girod, A. Preis *et al.*, "Waterwise: A testbed for continuous monitoring of the water distribution system in singapore," in *Water Distribution Systems Analysis*, 2010.

[16] S. Kartakis, E. Abraham, and J. A. McCann, "Waterbox: A testbed for monitoring and controlling smart water networks," 2015.

[17] "https://www.washingtonpost.com/news/monkey-cage/wp/2017/09/06/harvey-caused-a-chemical-plant-explosion-is-that-the-next-face-of-climate-change/."

[18] A. Kumar, M. Kansal, G. Arora *et al.*, "Detecting accidental contaminations in municipal water networks," *WRPM*, 1999.

[19] A. Kessler, A. Ostfeld, and G. Sinai, "Detecting accidental contaminations in municipal water networks," *WRPM*, 1998.

[20] J. W. Berry, L. Fleischer, W. E. Hart *et al.*, "Sensor placement in municipal water networks," *WRPM*, 2005.

[21] J. Berry, W. E. Hart, C. A. Phillips *et al.*, "Sensor placement in municipal water networks with temporal integer programming models," *WRPM*, 2006.

[22] A. Ostfeld *et al.*, "The battle of the water sensor networks (bwsn): A design challenge for engineers and algorithms," *WRPM*, 2008.

[23] A. Krause, J. Leskovec *et al.*, "Efficient sensor placement optimization for securing large water distribution networks," *WRPM*, 2008.

[24] R. Sarrate, V. Puig, T. Escobet *et al.*, "Optimal sensor placement for model-based fault detection and isolation," in *Decision & Control*, 2007.

[25] R. Perez, V. Puig, J. Pascual *et al.*, "Methodology for leakage isolation using pressure sensitivity analysis in water distribution networks," *Control Eng. Practice*, 2011.

[26] R. Pinzinger, J. Deuerlein, A. Wolters *et al.*, "Alternative approaches for solving the sensor placement problem in large networks," in *World Envir & Water Resources Congress*, 2011.

[27] L. S. Perelman, W. Abbas, X. Koutsoukos *et al.*, "Sensor placement for fault location identification in water networks: A minimum test cover approach," *Automatica*, 2016.

[28] Q. Han, P. Nguyen, R. T. Eguchi *et al.*, "Toward an integrated approach to localizing failures in community water networks," in *Distributed Computing Systems (ICDCS)*, 2017.

[29] J. Sun, R. Wang, and H.-F. Duan, "Multiple-fault detection in water pipelines using transient-based time-frequency analysis," *Hydroinformatics*, 2016.

[30] I. Narayanan, A. Vasan, V. Sarangan *et al.*, "One meter to find them all-water network leak localization using a single flow meter," in *Info Processing in Sensor Networks*, 2014.

[31] J. R. Shewchuk, "Triangle: Engineering a 2d quality mesh generator and delaunay triangulator," in *Applied computational geometry towards geometric eng.*, 1996.

[32] L. Begnudelli, B. F. Sanders, and S. F. Bradford, "Adaptive godunov-based model for flood simulation," *Hydraulic Eng.*, 2008.

[33] D. Misiūnas, *Failure Monitoring and Asset Condition Assessment in Water Supply Systems*, 2008.

[34] A. Lambert, "What do we know about pressure-leakage relationships in distribution systems," in *leakage control and WDS mgmt.*, 2001.

[35] L. A. Rossman *et al.*, "Epanet 2: users manual," 2000.

[36] R. Pérez, V. Puig, J. Pascual *et al.*, "Pressure sensor distribution for leak detection in barcelona water distribution network," *Water science & tech: water supply*, 2009.

[37] B. Sanders and L. Begnudelli, "Brezo: A hydrodynamic flood simulation algorithm," 2010.

[38] S. Khuller, A. Moss, and J. S. Naor, "The budgeted maximum coverage problem," *Information Processing Letters*, 1999.

[39] M. Minoux, "Accelerated greedy algorithms for maximizing submodular set functions," *Opt Techs*, 1978.

[40] G. L. Nemhauser *et al.*, "An analysis of approximations for maximizing submodular set functions," *Mathematical Programming*, 1978.

[41] http://emps.exeter.ac.uk/engineering/research/cws/downloads/benchmarks/, "Centre of water systems university of exeter."

[42] D. Gesch *et al.*, "The national map elevation," Tech. Rep., 2009.

[43] www.elevationmap.net, "Elevation map."

[44] T. Brinkhoff, "City population," 2005.

[45] M. Haklay and P. Weber, "Openstreetmap: User-generated street maps," *Pervasive Computing*, 2008.

APPENDIX

A. Table of Notations used in the paper

$\mathcal{S}$	Set of potential sensor locations
s_i	Sensor at location i
$\mathcal{L}$	Set of potential leak locations
l_j	Leak at location j
Δ	Set of all triangular regions
$\triangle_k$	k^{th} triangular region
$\eta(\triangle_k)$	Criticality of the k^{th} triangular region
$\mathcal{M}_{dc}$	Detection capability matrix
$\mathcal{M}_{dt}$	Detection time matrix
$\mathcal{M}_{fl}$	Flood level matrix
$\mathcal{I}_{l_j}$	Impact of leak at l_j
C_i	Set of leak locations covered by sensor at location i
C_{cov}	Set of leak locations covered by the sensor placement
$\mathcal{P}$	Set of sensor placements
U_{l_j}	Leak Utility
$\mathcal{U}$	Total utility

B. Cost Effectiveness Experiment

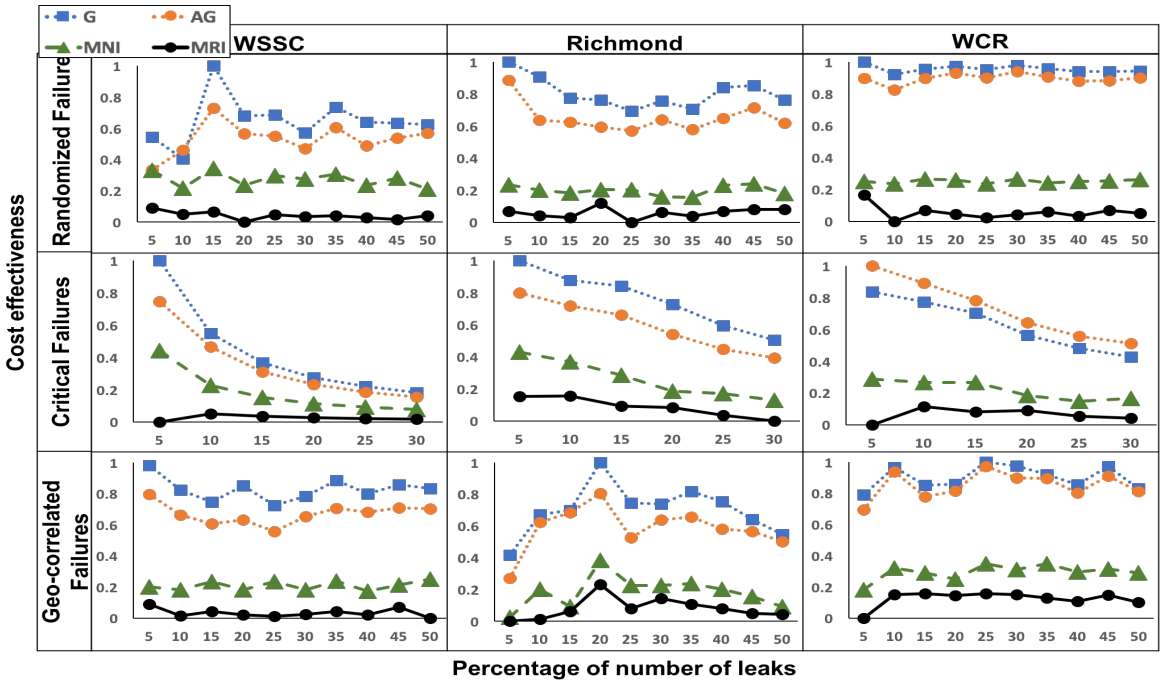


Fig. 10: Comparison of cost effectiveness as a function of number of leaks for randomized, critical, and geo-correlated leak events