Augmenting In-situ with Mobile Sensing for Adaptive Monitoring of Water Distribution Networks

Praveen Venkateswaran
University of California, Irvine
praveenv@uci.edu

Mahima Agumbe Suresh
San Jose State University
mahima.agumbesuresh@sjsu.edu

Nalini Venkatasubramanian
University of California, Irvine
nalini@ics.uci.edu

ABSTRACT

Instrumenting water distribution networks with sensors for monitoring is critical to maintain adequate levels of water quality and quantity. Existing efforts to detect and localize adverse events in the network have explored either installing in-situ sensors on junctions or deploying a number of mobile sensors through pipes. These approaches have high costs, low sensing accuracy, lack sufficient coverage or provide intermittent monitoring. In this paper, we combine the benefits of in-situ and mobile sensing with various geosocial factors to develop a cost-effective hybrid monitoring architecture that minimizes the impact of adverse water events on the community. The architecture can adaptively adjust sensing resolutions on-demand within the network, determine required sensing capabilities based on the event, and respond to varying event severities. We propose a two-phase planning and deployment approach that first integrates network structure, event, and community information with simulation based analytics to determine locations to install in-situ sensors and mobile sensor insertion infrastructure. We then incorporate network flow information to determine mobile sensor deployment locations and volume to quickly localize detected events to minimize their impact. We evaluate our approach using multiple real-world water networks for adverse water quality and loss events and compare it to existing approaches. Our results show that our proposed approach can achieve upto 79% reduction in impact with upto 68% greater cost efficiency compared to approaches using traditional coverage heuristics, and upto 30% reduction in impact while being upto 52% more cost efficient compared to approaches that attempt to minimize impact.

CCS CONCEPTS

• **Computer systems organization → Sensor networks**; *Maintainability and maintenance*;

KEYWORDS

Water Distribution Systems, Sensor Placement, Mobile Sensing, In-situ Sensing

ACM Reference Format:

Praveen Venkateswaran, Mahima Agumbe Suresh, and Nalini Venkatasubramanian. 2019. Augmenting In-situ with Mobile Sensing for Adaptive Monitoring of Water Distribution Networks. In *10th ACM/IEEE International Conference on Cyber-Physical Systems (with CPS-IoT Week 2019) (ICCPS '19), April 16–18, 2019, Montreal, QC, Canada.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3302509.3311048

ICCPS '19, April 16–18, 2019, Montreal, QC, Canada

ACM ISBN 978-1-4503-6285-6/19/04…$15.00
https://doi.org/10.1145/3302509.3311048

1 INTRODUCTION

Water distribution networks (WDNs) constitute one of the most critical urban infrastructures and are an important community lifeline. The monitoring of water networks is essential to ensure the availability of sufficient quantity and quality of water. Today's water networks are often decades old, and their growing scale and complexity make them increasingly vulnerable to adverse events [1]. Large pipe failures or leaks and the introduction of contaminants are the most common events affecting the quantity and quality of water in WDNs. Pipe leaks can result from stress caused by factors such as corrosion, pipe displacements, extreme weather, disaster events and can cause contaminants like nitrates, metals, and pesticides from the soil to enter the water pipelines through backflows [5]. The impact of physical damages to the infrastructure and compromises to the quality of supplied water can be devastating to society and cause huge economic and public health issues such as massive flooding, outbreak of waterborne epidemics, shortages of clean drinking water, damage to property, etc [28]. Having an effective methodology in place to monitor water networks is thus essential to localize and resolve these adverse events, in particular, those that disrupt and impact the community at large.

There have been several efforts towards instrumenting water distribution networks with sensors to detect and localize events in a timely manner. Some systems like PIPENET [26], WaterWise [30], WaterBox [10], and AquaSCALE [8], install in-situ or static sensors to measure network parameters like pressure, flow rate, turbidity, etc to detect the occurrence of events. Figure 1(a) shows the WaterWise multi-sensor probe that holds several commercial-off-the-shelf sensors for hydraulics and is inserted into the flow on pressured pipes. Static sensors provide continuous monitoring with one time installation and continuous communication costs. High-end static sensors provide deterministic performance, larger sensing ranges, and good accuracy. However, they are expensive (> $1000) [32] and the instrumentation of civic water infrastructures at large requires significant investments (millions of dollars).

Systems like SmartBall [6] and PipeProbe [12] drop mobile sensors into the network which traverse and monitor pipes by moving with the water flow. Figure 1(b) shows the deployment of a SmartBall into a water pipe. Mobile sensors detect events by traversing near them while flowing through the pipes and incur operational costs in terms of the manual effort or automated infrastructure required to deploy or retrieve them. Operating mobile sensors typically incurs lower cost than trenching and installing static sensors to retrofit existing pipe networks. Mobile sensors also allow for adaptive sensing on-demand as sensors can be deployed at different locations, at different times, and with different sensing capabilities based on the need. However, they do not provide continuous monitoring and have low sensing ranges. They also require a larger number of sensors since they have probabilistic movement through network junctions, and also need infrastructure support for their insertion into the network.

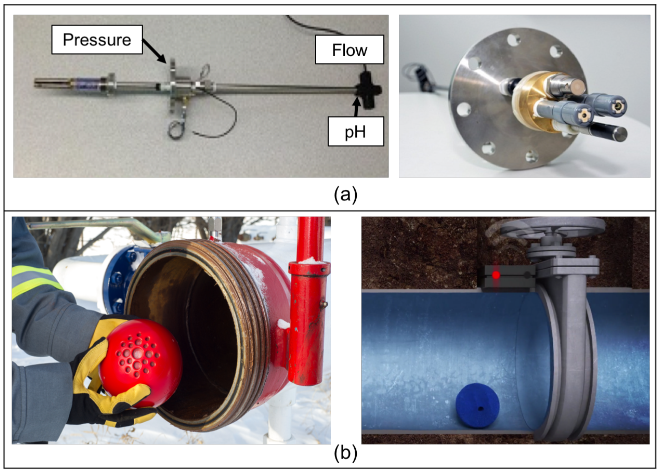


Figure 1: (a) WaterWiSe in-situ probe (b) SmartBall mobile deployment

In this paper, we propose to leverage the advantages provided by static sensors (continuous monitoring, sensing range, accuracy) and those by mobile sensors (adaptive sensing, on-demand monitoring, low cost) to develop a *hybrid* (i.e. in-situ and mobile) sensing architecture to provide *adaptive* monitoring of water networks. Our goal is to first plan and augment the placement of in-situ sensors with mobile sensor insertion infrastructure to quickly detect adverse events in the network and then determine locations from which to deploy mobile sensors to localize these events. Since the impact of events on the community are tied to their severity and the time taken to localize them, our planning and deployment methodology needs to be able to quickly detect and localize high impact events. In this paper, we focus on large pipe failures and contamination events.

There has been work on combining in-situ and mobile sensing in smart cities for pollution monitoring [15], community data collection [33], public safety [14] etc. However, achieving this in water networks presents several challenges - (1) Most of the network infrastructure is below ground, making it hard to deploy and operate sensors and involves other engineering efforts like developing mobile sensor insertion infrastructure, (2) The movement of mobile sensors is constrained by the direction and speed of the water flow that change over time, and (3) Communication is a challenge in underground networks where wireless networks suffer from attenuation while wired approaches require much cost and effort. Our contributions in this paper are as follows -

- We present a novel hybrid architecture that leverages the strengths of both in-situ and mobile sensors and combines it with various community geosocial factors to provide real-time adaptive monitoring of underground water distribution networks (Section 2).
- We design approaches to model the components of the architecture, the occurrence and propagation of events, and their resulting impact on the community structure (Section 3).
- We develop novel algorithms to (a) perform network planning to determine the placement of static sensors and mobile sensor insertion infrastructure and (b) determine mobile sensor deployment locations with the goal of reducing costs and ensuring low community impact while maintaining event detection and localization accuracy (Section 4).
- We evaluate the performance of our proposed hybrid architecture on three real-world water networks from Maryland, Colorado, and Richmond, and perform comparisons with existing approaches in detecting dynamic events resulting in the loss of water quality/quantity (Section 5).

2 HYBRID ADAPTIVE MONITORING ARCHITECTURE

In this section, we present our hybrid (in-situ plus mobile) architecture for the adaptive monitoring of water networks as shown in Figure 2. The physical infrastructure consists of the water distribution network, the surrounding community structure and the sensor deployments. In-situ or static sensors are installed on the pipes in contact with the water flow. Mobile sensors on the other hand, are inserted into, and extracted from the water flow through points that we denote as *Insertion/Extraction* (I/E) points. These could be manhole covers, fire hydrants or other specialized infrastructure. While the measurements from static sensors are uploaded as continuous data streams, the data from mobile sensors are uploaded once they are extracted at an (I/E) point.

Our system architecture has two phases. The first phase involves network planning where we leverage network information, community structure (terrain, population, locations of key infrastructure), and event propagation models in order to model the impact of different events occurring at various locations in the network on the surrounding community. We then use these impact models to drive our planning algorithm to determine the locations for urban planners and water agencies to install static sensors and mobile sensor (I/E) points. The static sensors continuously monitor the water network and whenever they detect the occurrence of a pipe failure or contamination event, they determine a *region of interest*, which constitutes a subset of junctions and pipes where the sensors believe that an event has occurred. In the second phase, we determine the $Insertion/Extraction$ points from which to deploy mobile sensors to quickly cover the *region of interest* so as to ensure the minimal impact of events on the surrounding community.

The adaptive monitoring capabilities of our hybrid architecture results from (a) the ability to dynamically adjust the sensing resolution of different areas of the network on-demand through the deployment of mobile sensors, (b) being able to pick and choose the exact sensing capabilities (sensing rate, types of sensors) of the mobile sensors required to localize each event based on the input provided by the static sensor deployment, and (c) catering to the severity of different events by incorporating information from static sensors to determine the number of mobile sensors required

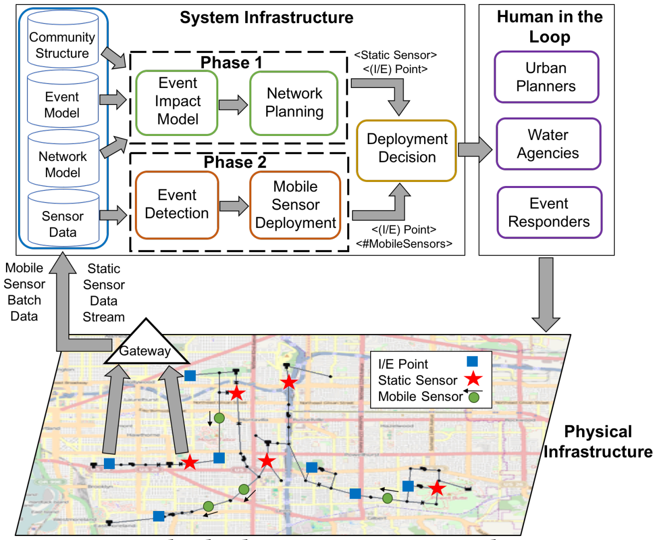


Figure 2: Hybrid Adaptive Monitoring Architecture

to provide adequate coverage and measurements from the *region of interest.*

Related Work. Existing work on combining mobile and in-situ sensor deployments in water networks typically assume the prior placement of static infastructure - either static sensors [18, 19, 22] that cover a predetermined portion of the network, or sink nodes/beacons [4, 27] in the junctions that communicate with the mobile sensors, with the objective of determining the number of mobile sensors needed and their release locations to cover the network. However, we argue that it is essential to consider the deployment of both types of sensors simultaneously since the placement of one type directly affects the performance of the other. Also, these prior approaches assume that all events are uniform and do not distinguish between them. However, (1) different events can impact a community in different ways and require unique sensing capabilities, (2) events of the same type can have varying severity, and (3) different communities can be impacted to differing extents by the same event based on their location, structure, demographics, built infrastructure, urbanization, etc. In our previous work [29], we developed a methodology to localize pipe failures using static sensors by characterizing the differing impacts caused by leaks on the community. We showed how this impact-driven approach can ensure low impact of pipe leaks on the community. However, our scope was limited to static sensor deployments and our methodology was tailored to failure detection. In contrast, this paper presents a more general adaptive monitoring framework for water distribution networks leveraging the strengths of both in-situ and mobile sensing and provides more cost-effective planning and deployment solutions to model and mitigate the impact of both failure and contamination events.

3 MODELING THE NETWORK, EVENTS AND ASSOCIATED COMMUNITY IMPACT

In this section, we describe our methodology for modeling the various components of the proposed hybrid infrastructure, the occurrence and propagation of contamination and failure events, the community structure and the associated impact of these events.

3.1 Modeling Infrastructure Components

Since the propagation of different events is dependent on the network flow, an event may manifest itself at only a subset of junctions in the network. There is also a time delay associated with the manifestation that increases with distance from the event source. Therefore, it is important to model the water network and the sensing capabilities of both types of sensors to ensure the continuous monitoring of the network and the timely detection and localization of events. We use a hydraulic simulator EPANET [23] developed by the United States Environmental Protection Agency, which simulates the hydraulic behavior within pressurized water distribution pipe networks, to model the sensing capabilities of the sensors.

Modeling the Water Network: A water distribution network can be represented as a graph, where the vertices represent nodes and junctions, while the edges represent links (pipes, valves, and pumps). We denote the set of potential locations for the occurrence of event $\mathcal{E}$ in the network as $\mathcal{E} = \{e_1, e_2, ..., e_n\}$, where e_j refers to an event occurring at location j. We also define the set of potential static sensor locations and mobile sensor *Insertion/Extraction* points as $\mathcal{S}^{stat} = \{s_1^{stat}, s_2^{stat}, ..., s_n^{stat}\}$ and $\mathcal{S}^{mob} = \{s_1^{mob}, s_2^{mob}, ..., s_n^{mob}\}$ respectively, where s_i^{stat}, s_i^{mob} refer to a static sensor and an (I/E) point at location i respectively. There could be locations deep underground where installing in-situ sensors is not possible but could be reached by mobile sensors deployed from existing (I/E) infrastructure. Similarly, there could be places where installing specialized (I/E) infrastructure is infeasible due to network access or cost, where in-situ deployments are more useful. Figure 3 shows a sample water network with five nodes that we use as a running example to illustrate our modeling approach.

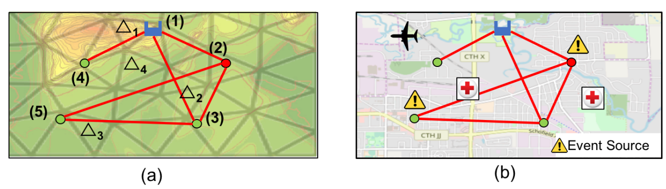


Figure 3: Running example of water network with its (a) elevation map and (b) key infrastructure information

$$\mathcal{M}_{dc} = \begin{array}{c c} & \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 \end{matrix} \\ \begin{matrix} s_1^{stat} \\ s_2^{stat} \\ s_3^{stat} \\ s_4^{stat} \\ s_5^{stat} \end{matrix} & \begin{pmatrix} 1 & 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 \end{pmatrix} \end{array} \quad \underset{\text{(secs)}}{\mathcal{M}_{dt}} = \begin{array}{c c} & \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 \end{matrix} \\ \begin{matrix} s_1^{stat} \\ s_2^{stat} \\ s_3^{stat} \\ s_4^{stat} \\ s_5^{stat} \end{matrix} & \begin{pmatrix} 3 & 12 & \infty & 22 & \infty \\ 14 & 5 & \infty & \infty & \infty \\ 27 & 9 & 2 & \infty & \infty \\ 19 & \infty & \infty & 4 & \infty \\ \infty & 35 & 26 & \infty & 4 \end{pmatrix} \end{array}$$

(a)

$$\mathcal{M}_{tc} = \begin{array}{c c} & \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 \end{matrix} \\ \begin{matrix} s_1^{mob} \\ s_2^{mob} \\ s_3^{mob} \\ s_4^{mob} \\ s_5^{mob} \end{matrix} & \begin{pmatrix} 1 & 7 & \infty & 6 & \infty \\ \infty & 1 & 3 & \infty & 4 \\ \infty & \infty & 1 & \infty & 1 \\ \infty & \infty & \infty & 1 & \infty \\ \infty & \infty & \infty & \infty & 1 \end{pmatrix} \end{array} \quad \underset{\text{(secs)}}{\mathcal{M}_{tt}} = \begin{array}{c c} & \begin{matrix} e_1 & e_2 & e_3 & e_4 & e_5 \end{matrix} \\ \begin{matrix} s_1^{mob} \\ s_2^{mob} \\ s_3^{mob} \\ s_4^{mob} \\ s_5^{mob} \end{matrix} & \begin{pmatrix} 4 & 44 & \infty & 67 & \infty \\ \infty & 6 & 36 & \infty & 84 \\ \infty & \infty & 3 & \infty & 91 \\ \infty & \infty & \infty & 7 & \infty \\ \infty & \infty & \infty & \infty & 5 \end{pmatrix} \end{array}$$

(b)

$$\underset{\text{(ft.)}}{\mathcal{M}_{fl}} = \begin{array}{c c} & \begin{matrix} \Delta_1 & \Delta_2 & \Delta_3 & \Delta_4 \end{matrix} \\ \begin{matrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{matrix} & \begin{pmatrix} 3.4 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1.6 \\ 0 & 0 & 2.7 & 0 \end{pmatrix} \end{array} \quad \underset{\text{(mg/L)}}{\mathcal{M}_{cl}} = \begin{array}{c c} & \begin{matrix} \Delta_1 & \Delta_2 & \Delta_3 & \Delta_4 \end{matrix} \\ \begin{matrix} e_1 \\ e_2 \\ e_3 \\ e_4 \\ e_5 \end{matrix} & \begin{pmatrix} 5.5 & 1.7 & 0 & 4.8 \\ 0 & 1.6 & 0.9 & 0 \\ 0 & 0 & 3.2 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \end{array}$$

(c)

Figure 4: Running example matrices for event detection and sensing capabilities of (a) static sensors and (b) mobile sensors, (c) Determining event propagation

Modeling In-Situ Sensors: In our hybrid architecture, the static sensors are responsible for detecting the occurrence of any event and determine a *region of interest* by continuously monitoring the network. An event would cause static sensors to return measurements depending on their ability to detect the event. The combination of these measurements can be used to identify junctions possibly affected by the event and hence track its propagation. Our goal is to then determine the potential event locations $e_j \in \mathcal{E}$ that can be detected by each static sensor location $s_i^{stat} \in \mathcal{S}^{stat}$ and the corresponding time taken to do so. To do this, we introduce an event at each potential location ($\mathcal{E}$) in EPANET. We then determine the sensor locations that can detect each event by monitoring the values of the requisite hydraulic variables for the event (pressure change for failures and contaminant concentration for contamination events). We build a *detection capability matrix* $\mathcal{M}_{dc}$, where the rows represent the potential static sensor locations and the columns represent the event locations. The entries of the matrix are binary valued (0 or 1) depending on whether the static sensor is capable of detecting the event [21]. We denote the observed values of the hydraulic variable under normal conditions as v_i and as $\hat{v}_i$ once the event is introduced. We also set a detection threshold ϵ for each event. Then the values of $\mathcal{M}_{dc}$ for a particular event can be computed as -

$$\mathcal{M}_{dc}[s_i^{stat}, e_j] = \begin{cases} 1, & \text{if } v_i - \hat{v}_i \geq \epsilon \\ 0, & \text{otherwise} \end{cases}$$

We then store the corresponding time taken for the static sensors to detect an event in a *detection time matrix* $\mathcal{M}_{dt}$ as -

$$\mathcal{M}_{dt}[s_i^{stat}, e_j] = \begin{cases} \eta(s_i^{stat})_{e_j}, & \text{if } \mathcal{M}_{dc}[s_i^{stat}, e_j] = 1 \\ \infty, & \text{otherwise} \end{cases}$$

where $\eta(s_i^{stat})_{e_j}$ is the time taken for a static sensor s_i^{stat} to detect an event at e_j measured in seconds. The values of $\mathcal{M}_{dt}$ are set to infinity for the locations where a static sensor is incapable of detecting the event.

Example. The detection capability and detection time matrices in Fig. 4(a) show the capability of static sensors to detect failures in the sample network. For instance, a static sensor at junction 1 can detect a leak at junction 2 with a delay of 12 seconds.

Modeling Mobile Sensors: Once the static sensors determine a *region of interest*, the mobile sensors are then deployed in order to localize the event. Since the mobile sensors flow along with the water, at each junction that connects multiple pipes, a mobile sensor may flow into any one of the outlet pipes. Thus, sensors released at the same time and location can take a number of possible traversal paths. We account for this uncertainty by adopting a probabilistic approach to model the flow of mobile sensors in the network [27].

At each junction, we assign the probability of the sensor flowing through an outlet pipe connecting junction i and j as p_{ij}=f_{ij}/T_i, where f_{ij} is the flow rate through the outlet pipe and T_i is the total flow rate out of junction i. We maintain this junction-to-junction transition probability information in a matrix (M) which reflects the probability that a mobile sensor at junction i would reach junction j in a single step. This also implies that the probability of a mobile sensor from junction i reaching junction k after two steps can be computed as p_{ik}= $\sum_j p_{ij}{*}p_{jk}$ for all intermediate junctions j, which translates to computing M^2. We repeat this for n steps until there are no more transitions (i.e., all the probabilities are 0) and create a traversal probability matrix as -

$$\mathcal{T} = \sum_{k=1}^{n} M^k, \text{such that } M^n = \mathbf{0} \tag{1}$$

where $\mathbf{0}$ denotes the zero matrix and each entry of $\mathcal{T}$ denotes the probability of a mobile sensor traversing from one junction to another in any number of steps. Our goal is to then translate these probabilities into finding the number of mobile sensors required to traverse a junction with a minimum coverage probability p_c. This can be modeled as a binomial distribution $b(n, p)$, where p is the probability that a mobile sensor will traverse to a junction, $(1 - p)$ the probability that it will not, and n the number of mobile sensors deployed. Hence, the probability that no mobile sensors will traverse to a junction can be represented as $(1-p)^n$. We define A as the event in which at least one mobile sensor traverses to a given junction. Our goal is to then determine n such that $Prob(A)$=$Prob(1-A')$=$1-(1-p)^n \geq p_c$. We thus obtain -

$$n \geq \left\lceil \frac{ln(1 - p_c)}{ln(1 - p)} \right\rceil \tag{2}$$

We see that for low traversal probabilities, a larger number of mobile sensors need to be deployed. We then use Equation (2) to build a *traversal capability matrix* $\mathcal{M}_{tc}$ where the rows denote the potential mobile sensor (I/E) points ($\mathcal{S}^{mob}$) and the columns denote the event locations ($\mathcal{E}$). The entries of $\mathcal{M}_{tc}$ represent the minimum number of mobile sensors required to traverse to each event location with probability p_c -

$$\mathcal{M}_{tc}[s_i^{mob}, e_j] = \begin{cases} n_{e_i e_j}, & \text{if } \mathcal{T}[s_i^{mob}, e_j] > 0 \\ \infty, & \text{otherwise} \end{cases}$$

We also build the corresponding traversal time matrix $\mathcal{M}_{tt}$ using the network flow rate information as the time taken for the mobile sensor to traverse from one junction to another.

$$\mathcal{M}_{tt}[s_i^{mob}, e_j] = \begin{cases} \theta(s_i^{mob}, e_j), & \text{if } \mathcal{M}_{tc}[s_i^{mob}, e_j] \neq \infty \\ \infty, & \text{otherwise} \end{cases}$$

where $\theta(s_i^{mob}, e_j)$ is the time taken for a mobile sensor to traverse from mobile sensor location s_i^{mob} to event location e_j as a sum of the time taken to traverse each intermediate pipe which we compute using the flow rates and pipe lengths.

Example. Fig. 4(b) shows the traversal capability and traversal time matrices for mobile sensors in detecting contamination events in the sample network. We see that traversing from junction 1 to junction 2 requires at least 7 mobile sensors to achieve a 95% coverage probability and they take 44 seconds to reach junction 2.

3.2 Modeling Events - Water Quality and Quantity

The occurrence of different events in the network can result in multiple hydraulic variables being affected. It is therefore important to accurately model the events to determine the manner of propagation of each event. Since different events can affect different parts of the community, we partition the community into smaller regions using a delaunay triangulator, Triangle [25]. Triangular grids allow for localized grid refinement and can easily conform to terrains with irregular shapes [2]. We denote the set of triangular regions as $\Delta = \{\triangle_1, \triangle_2, ..., \triangle_n\}$. We use EPANET to model the occurrence and propagation of contamination and failure events.

3.2.1 *Contamination Events.*

In contamination events, a dissolved contaminant travels down the network with the same average velocity as the carrier fluid while at the same time reacting (either growing or decaying) at some given rate. Hence, contamination events can be detected by monitoring the chemical concentration levels in the water. We assume that at junctions, the mixing of fluid is complete and instantaneous.

Definition.*(Contaminant Spread)* The spreading of contaminants in the network can be computed as -

$$\frac{\partial C_i}{\partial t} = -u_i \frac{\partial C_i}{\partial x} + r(C_i) \tag{3}$$

where C_i is the concentration in pipe i as a function of distance x and time t, u_i is the flow velocity in pipe i and r is the rate of reaction as a function of concentration [23].

Contamination levels The degree of contamination at each junction is an important factor to consider while measuring the impact of a contamination event on the community. People consuming water from a contaminated junction within a region would be adversely affected. We measure this using EPANET by injecting a contaminant at each potential event location $e_j \in \mathcal{E}$ and simulating its spread throughout the network using Equation (3). We then build a *contaminant level matrix* $\mathcal{M}_{cl}$ where the rows and columns correspond to the contamination event locations ($\mathcal{E}$) and the triangular regions (Δ) respectively. We compute the entries of $\mathcal{M}_{cl}$ as -

$$\mathcal{M}_{cl}[e_j, \triangle_k] = \begin{cases} 0, & \triangle_k \text{ does not consume from } e_j \\ C(e_j)_{\triangle_k}, & \text{otherwise} \end{cases}$$

where $C(e_j)_{\triangle_k}$ denotes the concentration levels at $\triangle_k$ due to the contaminant intrusion at e_j.

3.2.2 *Failure Events.*

Physical infrastructure failures, such as pipe leaks/breaks, cause a disturbance in the water flow resulting in a pressure wave that moves through the network with high velocity [16]. Past work has shown that the the velocity of water exiting from the leak orifice is faster than within the pipe causing a pressure drop, implying that pipe bursts can be identified by detecting changes in hydraulic pressure [20, 29].

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

$$Q = E_c \times p^{\beta} \tag{4}$$

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 [13].

Flood levels One of the main factors influencing the impact of pipe failures on the community is the flooding resulting from water outflow and seepage from leaks. We capture this by simulating the outflow of water from a leak as well as its propagation along the surrounding terrain using a hydrodynamic flood simulation algorithm BreZo [24]. We simulate leak events in EPANET by introducing emitters. We then compute the outflow rate for each leak event $e_j \in \mathcal{E}$ using Equation (4) and provide this as input to the BreZo simulator in addition to the triangular regions (Δ) and the leak location (e_j). The BreZo simulator returns the regions affected by flooding and the corresponding flood levels. We use this information to build a *flood level matrix* $\mathcal{M}_{fl}$ consisting of the leak event locations as the rows and the triangular regions as the columns. The entries of $\mathcal{M}_{fl}$ can be computed as -

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

where $H(e_j)_{\triangle_k}$ denotes the maximum flood level at $\triangle_k$ due to a leak at e_j.

Example. For the sample network, we measure the effects of failure and contamination events introduced at each junction for four triangular regions denoted by $\triangle$ in Fig. 3(a). The resulting flood and contamination level matrices are shown in Fig. 4(c).

3.3 Modeling Community Structure and Event Impact

Estimating the impact of any event in the network on the community requires developing a model of the community structure. We then extract the following community information from each of the triangular community regions (Δ) -

1) Critical Infrastructure: We use mapping services to identify the presence of critical infrastructure such as healthcare, transportation, government facilities, education, etc within the boundaries of each region and assign relative importance scores to each of these categories. We then compute the critical infrastructure score $\triangle_k^{inf}$, of region $\triangle_k$, as the sum of the scores of the infrastructure located in $\triangle_k$.

2) Population Information: We obtain the population density information of each of the triangular regions using census data and denote it as $\triangle_k^{pop}$.

3) Elevation Information: We build elevation maps for each of the triangular regions as the average elevation of its vertices and denote it as $\triangle_k^{ele}$.

4) Demand: We determine the average consumption of water by each triangular region $\triangle_k$ by averaging the total supply through each of the network junctions in $\triangle_k$, and denote this by $\triangle_k^{dem}$.

Example. Figure 3(a) shows the triangular grids and the terrain information used to obtain $\triangle_k^{ele}$ while Figure 3(b) shows the presence of critical infrastructure around the water network to compute $\triangle_k^{inf}$.

Measuring the impact of an event on the community ($\mathcal{I}^{\mathcal{E}}$) is dependent on the type of event that occurs in the network. Here, we present our methodology for measuring the impact of both pipe failures and contamination events.

Impact of Pipe Failures A large pipe failure can cause flooding in the surrounding area, thus affecting the population present as well as the functioning of critical infrastructure. We thus compute the impact of a leak event at location j on region $\triangle_k$ as -

$$\mathcal{I}_{e_j}^{leak} = \mathcal{M}_{fl}[e_j^{leak}, \triangle_k] * (\triangle_k^{pop} + \triangle_k^{inf}) \tag{5}$$

where $\mathcal{M}_{fl}[e_j^{leak}, \triangle_k]$ is the level of flooding caused by the leak event, $\triangle_k^{pop}$ is the population density, and $\triangle_k^{inf}$ is the critical infrastructure score of the region.

Impact of Contamination Events The impact of a contamination event would depend on the amount of contaminated water consumed as well as the number of people consuming the water. We estimate the impact of a contamination event at location j on region $\triangle_k$ as -

$$\mathcal{I}_{e_j}^{cont} = \mathcal{M}_{cl}[e_j^{cont}, \triangle_k] * (\triangle_k^{pop} + \triangle_k^{dem}) \quad (6)$$

where $\mathcal{M}_{cl}[e_j^{leak}, \triangle_k]$ is the contamination levels caused by the event, $\triangle_k^{pop}$ is the population density, and $\triangle_k^{dem}$ is the consumption demand of the region.

4 NETWORK PLANNING AND DEPLOYMENT ALGORITHMS

As described in Section 2, given a water network, our proposed architecture determines the placement of static sensors and *Insertion/Extraction* points to quickly identify the *regions of interest* and then identifies the deployment locations of mobile sensors to provide rapid localization of events. In this section, we present our network planning and deployment algorithms. The goal of our planning algorithm is to simultaneously determine locations to place both static sensors and mobile sensor *Insertion/Extraction* points while providing high utility and incurring low costs. We define a sensor placement $\mathcal{P}$ to consist of a set of static sensor locations and mobile sensor (I/E) points (i.e.) $\mathcal{P} \subseteq (\mathcal{S}^{stat} \bigcup \mathcal{S}^{mob})$. The utility of a placement can be measured in terms of the impact caused by events in the network due to the delay in their detection and localization. A sensor placement that provides high utility is thus one that 1) quickly detects and localizes high impact events, 2) results in low overall impact on the community, and 3) incurs low costs.

We denote C_i^{stat}, C_i^{mob} as the set of event locations $(\mathcal{E})$ that can be detected by a static sensor placed at, or traversed to by mobile sensors deployed from location i respectively. We also define $cost_s : cost_m$ as the cost ratio of static and mobile sensors and C^{cov} as the set of event locations detectable by the deployment. We acknowledge that modeling the various costs involved is complex and dependent on the event type and severity, and use sensor cost ratios to determine the relative utility provided by in-situ and mobile sensors. The objective of our planning algorithm is to then identify a minimum cost placement set $\mathcal{P} = (\mathcal{P}^{stat} \cup \mathcal{P}^{mob})$ that covers the set of all detectable events such that its overall utility is maximized. This is equivalent to the *weighted set cover* problem defined as -

Definition.*(Weighted Set Cover)* Let $\mathcal{L}$ be a finite set of elements and C=$\{C_1, C_2, ..., C_n\}$ be a set of subsets of $\mathcal{L}$ with weights $\mathcal{W} = \{w_1, w_2, ..., w_n\}$. The goal is to find a set $C_s \subseteq C$ such that all the elements are covered by C_s (i.e.) $\bigcup C_k = \bigcup C_i, \forall C_k \in C_s, \forall C_j \in C$, and the sum of weights in C_s is minimized (i.e.) $minimize(\sum w_k, \forall C_k \in C_s)$.

Using this definition, if C is the collection of event locations covered by each sensor, and $\mathcal{W}$ the corresponding costs of static and mobile sensors, finding a set cover C_s is a solution to the *WSC* problem and hence shows the equivalence. The *WSC* problem is NP-hard [11] thus implying that finding a solution for the planning problem is also computationally complex. We present our approximate Hybrid Impact Driven network planning algorithm as described in Algorithm 1 -

- For a given budget $\mathcal{B}$ and set of event locations $\mathcal{E}$, we iterate over the set of potential sensor locations and determine the utility of placing a static sensor or an *Insertion/Extraction* point at each location.
- For a static sensor s_i^{stat}, we compute its utility for event e_j as a function of the impact caused in the time taken for the sensor to detect the event (i.e.) $U_{e_j}^{stat}(s_i^{stat}) = \mathcal{I}_{e_j} / \mathcal{M}_{dt}[s_i^{stat}, e_j]$.
- The utility for an (I/E) point s_i^{mob} depends on the impact caused in the time taken for the existing static sensors to determine the *region of interest* and for the mobile sensors to traverse to the event location (i.e.) $U_{e_j}^{mob}(s_i^{mob}) = \mathcal{I}_{e_j} / (\delta_{dt}[e_j] + \mathcal{M}_{tt}[s_i^{mob}, e_j])$, where $\delta_{dt}[e_j]$ is the shortest time taken for the placed static sensors to detect e_j.
- We then compute the total utility as a function of the cost incurred to achieve the above utilities. This is

  $$\mathcal{U}^{stat}(s_i^{stat}) = \sum_{j=1}^{|\mathcal{E}|} U_{e_j}^{stat}(s_i^{stat}) / cost_s$$

  for static sensors, and

  $$\mathcal{U}^{mob}(s_i^{mob}) = \sum_{j=1}^{|\mathcal{E}|} U_{e_j}^{mob}(s_i^{mob}) / (\mathcal{M}_{tc}[s_i^{mob}, e_j] * cost_m)$$

  for mobile sensor *Insertion/Extraction* points, where $\mathcal{M}_{tc}[s_i^{mob}, e_j]$ is the minimum number of mobile sensors needed to be deployed from s_i^{mob}.
- At the end of each iteration, we choose the sensor type and location pairing with the largest utility and add it to the placement set. We do this until either the budget is exhausted or the network has been covered.

The algorithm is guaranteed to complete since every detectable event location has at least one static sensor location that covers it. The worst case running time of the algorithm, $O((|\mathcal{S}^{stat}| + |\mathcal{S}^{mob}|)|\mathcal{E}|^2)$, occurs when at each iteration, only one new event location is covered. This would result in long runtimes for large scale water networks. We however use the concept of *submodularity* to significantly reduce the number of utility computations in each iteration, thus reducing the runtime [17].

Definition. *(Submodularity)* Let C be a finite set and f be a set function. For all subsets $C_s \subseteq C_r \subseteq C$, and elements $c_i \in C \setminus C_r$, f is submodular whenever $f(C_s \cup c_i) - f(C_s) \geq f(C_r \cup c_i) - f(C_r)$.

Theorem 1. *The utility functions $\mathcal{U}^{stat}, \mathcal{U}^{mob}$ are submodular.*

We provide the proof in the appendix □

Our greedy approach gives an approximation ratio of $(1-1/e)$ similar to the weighted set cover problem [17]. Hence, in a given iteration of the algorithm if $\mathcal{U}(s_1) \geq \mathcal{U}(s_2) \geq \mathcal{U}(s_3) \geq ...$, 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.

Once the locations of static sensors and mobile sensor *Insertion/Extraction* points have been determined, we need to identify the mobile sensor deployment locations. This depends on the direction and velocity of water flow in the network as well as the junctions that need to be localized. Given the junctions within

Algorithm 1 Network Planning Algorithm

Input: $\mathcal{S}^{stat}$, $\mathcal{S}^{mob}$, $\mathcal{E}$, $\mathcal{M}_{dc}$, $\mathcal{M}_{dt}$, $\mathcal{M}_{tc}$, $\mathcal{M}_{tt}$, $\mathcal{B}$, $cost_s$, $cost_m$, $\mathcal{P}^{stat}$, $\mathcal{P}^{mob}$, C^{stat}, C^{mob}, C^{cov}
Output: $\mathcal{P} = (\mathcal{P}^{stat} \cup \mathcal{P}^{mob})$
Initial Conditions: $\mathcal{P}^{stat} = \emptyset$, $\mathcal{P}^{mob} = \emptyset$, $C^{cov} = \emptyset$,
while $C^{cov} \neq |\mathcal{E}|$ **and** $\mathcal{B} > 0$ **do**
  for $i = 1 \rightarrow |\mathcal{S}^{stat} \cup \mathcal{S}^{mob}|$ **do**
    if $s_i^{stat} \notin \mathcal{P}^{stat}$ **then**
      for $j = 1 \rightarrow |\mathcal{E}|$ **do**
        $U_{ej}^{stat}(s_i^{stat})$=$\mathcal{I}_{e_j}/\mathcal{M}_{dt}[s_i^{stat}, e_j]$
      $\mathcal{U}^{stat}(s_i^{stat})$= $\sum_{j=1}^{|\mathcal{E}|} U_{ej}^{stat}(s_i^{stat})/cost_s$
    if $s_i^{mob} \notin \mathcal{P}^{mob}$ **then**
      for $j = 1 \rightarrow |\mathcal{E}|$ **do**
        $U_{ej}^{mob}(s_i^{mob})$=$\mathcal{I}_{e_j}/(\delta_{dt}[e_j] + \mathcal{M}_{tt}[s_i^{mob}, e_j])$
      $\mathcal{U}^{mob}(s_i^{mob})$= $\sum_{j=1}^{|\mathcal{E}|} U_{ej}^{mob}(s_i^{mob})/(\mathcal{M}_{tc}[s_i^{mob}, e_j]*cost_m)$
    $\mathcal{U}(s_i)$=$max(\mathcal{U}^{stat}(s_i^{stat}), \mathcal{U}^{mob}(s_i^{mob}))$
  $\mathcal{U}^{max}(s_v)$={$\mathcal{U}(s_i) : max(\mathcal{U}(s_i)), \forall i : 1 \rightarrow |\mathcal{S}|$}
  if $\mathcal{U}^{max}(s_v) = \mathcal{U}^{stat}(s_v)$ **then**
    $\mathcal{B} \leftarrow \mathcal{B} - cost_s$
    $C^{cov} \leftarrow C^{cov} \cup C_v^{stat}$
    $\mathcal{P}^{stat} \leftarrow \mathcal{P}^{stat} \cup s_v$
  else
    $\mathcal{B} \leftarrow \mathcal{B} - (\mathcal{M}_{tc}[s_v^{mob}, \mathcal{E}] * cost_m)$
    $\mathcal{P}^{mob} \leftarrow \mathcal{P}^{mob} \cup s_v$

the *regions of interest* $\mathcal{R}$, our objective is to identify the subset of (I/E) points determined by our planning algorithm at which mobile sensors need to be deployed (i.e.) $\mathcal{D} \subseteq \mathcal{P}^{mob}$. We describe our deployment algorithm (Algorithm 2) -

Algorithm 2 Mobile Sensor Deployment Algorithm

Input: $\mathcal{R}$, $\mathcal{M}_{tt}$, $\mathcal{M}_{tc}$, $\mathcal{P}^{mob}$
Output: $\mathcal{D} \subseteq \mathcal{P}^{mob}$
for all $e_j \in \mathcal{R}$ **do**
  $d = \emptyset$, $d^t = \emptyset$
  for $i = 1 \rightarrow |\mathcal{P}^{mob}|$ **do**
    if $\mathcal{M}_{tc}[\mathcal{P}_i^{mob}, e_j] > 0$ **and** $\mathcal{M}_{tt}[\mathcal{P}_i^{mob}, e_j] < d^t$ **then**
      $d^t = \mathcal{M}_{tt}[\mathcal{P}_i^{mob}, e_j]$
      $d = \mathcal{P}_j^{mob}$
  $\mathcal{D} \leftarrow \mathcal{D} \cup d$

• For every junction e_j in the *region of interest* $\mathcal{R}$, we identify the (I/E) points in the placement set $\mathcal{P}^{mob}$, from which mobile sensors can traverse to e_j (i.e.) $\mathcal{M}_{tc}[\mathcal{P}_i^{mob}, e_j] > 0$.
• We then determine the (I/E) point with the shortest traversal time and add it to the deployment set $\mathcal{D}$.
• We repeat this till the entire *region of interest* has been localized.

More complex models for mobile sensor deployment are possible that exploit existing control systems in the network such as pumps and valves to change the direction and speed of water flow that could result in fewer (I/E) points needed to ensure the reachability and coverage of mobile sensors. However, this requires detecting the state of the systems and estimating their levels of functionality in the aftermath of the event. Our proposed approach, though more conservative, results in deployment solutions unaffected by effects of the event on these systems.

5 EXPERIMENTAL RESULTS

In this section, we evaluate our proposed hybrid adaptive monitoring architecture for both failure and contamination events. We compare the performance of our approach to existing sensor deployment approaches using water networks of varying scale and validate our approach under multiple event scenarios. While our experimental studies are conducted using simulators, we intend to develop a testbed as part of our future work.

5.1 Experimental Setup

Water Networks. We evaluate our proposed architecture using three real-world water networks of varying scale - (1) a subzone of the Washington Suburban Sanitary Commission's (WSSC) water service area in Montgomery County, Maryland, (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 for (2) and (3) from [9]. A summary of the network layouts is shown in Figure 5. Figure 6 shows the layout of the WSSC network and its surrounding community and shows the placement of in-situ sensors and mobile sensor (I/E) points determined by the proposed planning algorithm for a subset of the network. The layouts of the Richmond and WCR networks are shown in the appendix.

Network	Length (km)	Demand 10^3 [m^3/day]	No. of Pipes	No. of Junctions
WSSC	32.46	1.57	316	299
Richmond	75.61	15.12	948	865
WCR	383.59	82.95	1985	1782

Figure 5: Experimental Network Layout Summary

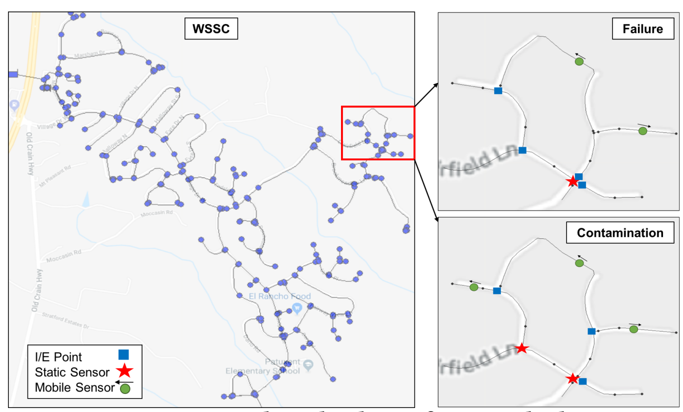


Figure 6: WSSC network and subset of sensor deployment for failure and contamination events

Comparison Approaches. We compare our proposed Hybrid Impact Driven planning and deployment approach *(HID)* to three existing sensor placement approaches in water networks - Static Coverage Driven *(SCD)* [20], Hybrid Coverage Driven *(HCD)* [19], and Static Impact Driven *(SID)* from our previous work [29]. Here, static and hybrid refer to the sensor types being used in the approach. The *SCD* approach iteratively selects static sensor locations based on the number of event locations covered and their ability to distinguish between pairs of events. The *HCD* approach uses a cross entropy based methodology to select a percentage of junctions to install static sensors followed by determining mobile sensor release points. The *SID* approach iteratively determines locations to install static sensors based on their achieved impact mitigation of the event on the community. Figure 7 shows the number of in-situ sensors and mobile sensors in the deployments determined by each approach. We observe that sensor deployments use fewer sensors to detect and localize failure events since the event propagation (movement of pressure wave) is faster as compared to the contaminant flow which is restricted by the flow velocity of the water in the network. We also see that the hybrid approaches result in the deployment of far fewer in-situ sensors thus reducing the costs incurred. However, the deployments resulting from impact driven approaches use more sensors than their coverage driven counterparts since they attempt to quickly localize high impact events resulting in more sensors being deployed to cover vulnerable regions.

	Failure (in-situ/mobile)			Contamination (in-situ/mobile)		
	WSSC	Richmond	WCR	WSSC	Richmond	WCR
SCD	61/0	241/0	402/0	83/0	270/0	430/0
SID	89/0	310/0	489/0	110/0	367/0	512/0
HCD	11/54	104/126	147/261	41/54	141/144	154/288
HID	20/62	89/240	122/343	27/73	112/261	186/313

Figure 7: Number of sensors deployed (in-situ/mobile) by each approach for failure and contamination events

Event Scenarios. We compare the performance of the sensor deployments resulting from each of the approaches for the following event scenarios -

1) *Geo-correlated events:* We simulate the cascading effects of failure and contamination events by introducing them in spatially clustered locations ranging from 5% to 50% of the network's junctions where the number of events in each cluster is uniform.

2) *Critical events:* We then introduce failure and contamination events in the top 5% to 30% junctions ordered by impact.

5.2 Evaluating Effectiveness of Hybrid Sensor Deployments

We compare the effectiveness of the sensor deployments resulting from each approach using three metrics - (a) Detection and localization times, (b) Impact caused, (c) Cost effectiveness. In order to determine the impact of failure and contamination events as described in Section 3.3, we obtain community structure information by building the terrain elevation map ($\triangle^{ele}$) using elevation data from [31], obtain population density information ($\triangle^{pop}$) from census data [3], mine the coordinates of critical infrastructure ($\triangle^{inf}$) in the area using the OpenStreetMap service [7], and determine the demand of water at each junction ($\triangle^{dem}$) from the network model.

Evaluating Detection and Localization Times: For each introduced failure or contamination event in the above event scenarios, we determine the shortest time taken to detect and localize the event by the sensors deployed by each approach. We then compare the average of these shortest times for all the introduced events. Figure 10(a) shows the comparison of the average detection times by each of the approaches for geo-correlated failure and contamination events. We see that in general, the detection and localization of failure events ($98 - 398$ sec) is faster than contamination events ($1010 - 7035$ sec). Also, the detection and localization time of contamination events increases with the size of the network, while that of failure events remains consistent. This happens because the high speed pressure wave resulting from pipe failures can be detected quickly even in larger networks as compared to the slower moving contaminant flow. We observe that the coverage driven approaches (*SCD*,*HCD*) take a longer time on average to detect and localize events as compared to impact driven approaches (*SID*,*HID*) since their coverage based objective results in sparser sensor deployments. We also observe that the proposed hybrid *HID* approach takes approximately 9% longer to detect and localize events as compared to the in-situ based *SID* approach.

Extent of Impact Caused: We then compare the approaches based on the impact caused to the community by the failure and contamination event scenarios before their deployments can detect and localize the events. For each introduced event, we determine its impact (Section 3.3) caused as a function of the shortest time taken for each approach's sensor deployment to detect and localize it. We then compute the average normalized impact caused over all the introduced events (Figure 10(b)). We see that for geo-correlated events, the impact driven approaches (*SID*,*HID*) result in much lower impacts on average since they prioritize the quick detection of high impact events and the coverage of critical regions. We observe that the proposed *HID* approach results in upto nearly 30% lesser impact than the *SID* approach due to the faster localization of events using mobile sensors and upto nearly 79% lesser impact than the coverage based approaches.

Sensor type cost ratio: Here, we determine the influence of the cost ratio between mobile and static sensors on their proportion in the sensor deployment resulting from the proposed *HID* approach by varying the cost ratio of mobile to static sensors from 1:1 to 1:10. Figure 8 shows the proportion of static and mobile sensors for the WSSC network for (a) failure and (b) contamination events. We see that there is a stabilization in the proportions at a 1:5 cost ratio for the WSSC network. We observe that this ratio increases with an increase in the size of the network.

Comparison of coverage and cost of deployment: We then use the 1:5 cost ratio to compare the costs of the deployments resulting from each of the four approaches. We vary the number of sensors from 10% to 100% of the total number of sensors in each deployment and compare the costs incurred and the coverage of the network achieved at each step for the WSSC network. Figure 9 shows that approaches using only in-situ sensors (*SID*,*SCD*) incur much larger costs than the hybrid approaches (*HID*,*HCD*). We also observe that the coverage driven approaches achieve higher network coverage using lesser number of sensors.

Cost effectiveness of deployments: We evaluate the normalized cost effectiveness of the deployments as a function of the total impact caused and the total cost incurred. We observe from Figure 10(c) that for geo-correlated events, the proposed *HID* approach proves to be the most cost effective by upto nearly 52% over the *SID* approach and upto nearly 68% over the coverage based approaches. This would improve as the cost ratio between mobile and static sensors increases.

Performance under Critical Events: Due to space constraints, we present the results of the comparison of the four approaches for the critical events scenario in the appendix. We observe that the hybrid driven approaches detect critical events much faster than coverage based approaches and result in lower impacts of events. We also see that the proposed *HID* approach remains the most cost effective by upto nearly 40% over the *SID* approach.

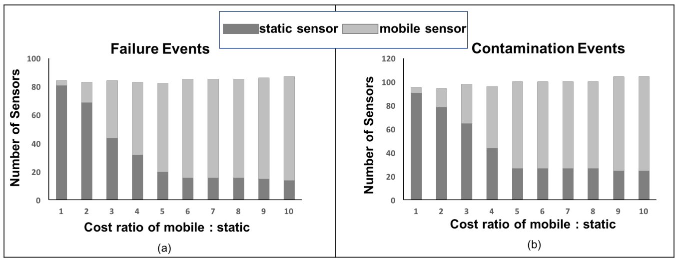


Figure 8: Proportion of in-situ and mobile sensors in proposed hybrid approach deployment with varying mobile:static sensor cost ratios for the WSSC network

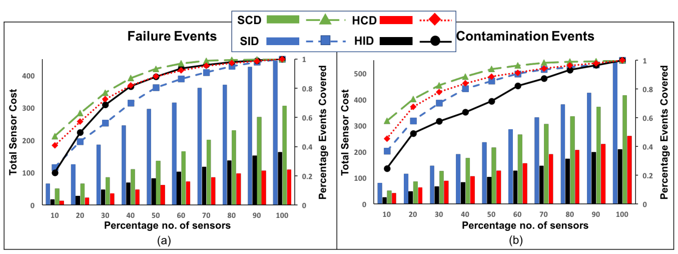


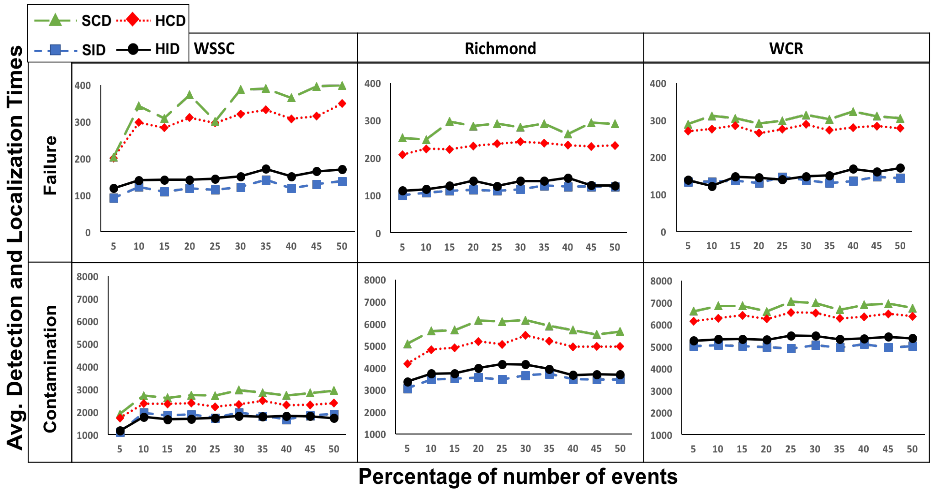
Figure 9: Progression of sensor deployment costs and achieved event coverage with increasing number of sensors by the approaches for the WSSC network

6 CONCLUSION AND FUTURE WORK

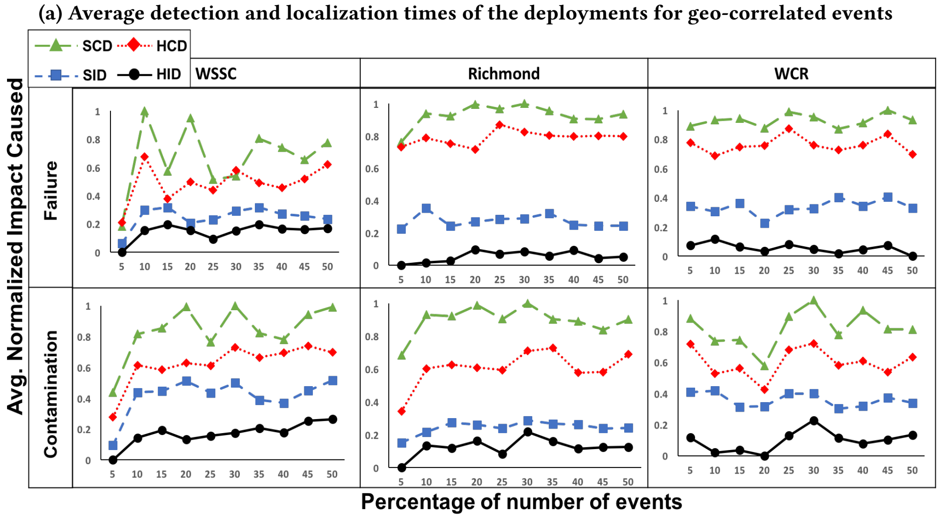
In this paper we presented a novel architecture for the adaptive monitoring of water distribution networks to quickly detect and localize adverse events like pipe breaks and chemical contamination. The architecture leverages in-situ and mobile sensing to provide a cost-effective solution that minimizes the impact of these events on the community. We presented a two-phase approach that incorporates information about the network, events, and the community to determine locations to install sensing infrastructure and to deploy mobile sensors and evaluate its effectiveness using real-world water networks of varying scale. While we demonstrate that a hybrid architecture provides improved cost-effectiveness, its biggest advantage lies in the flexibility provided by its adaptive monitoring capabilities. As part of our future work, we intend to provide more detailed models of the installation and operational costs incurred by in-situ and mobile sensors and determine methods to leverage existing control systems and infrastructure while maintaining reliability under sensor measurement uncertainties.

REFERENCES

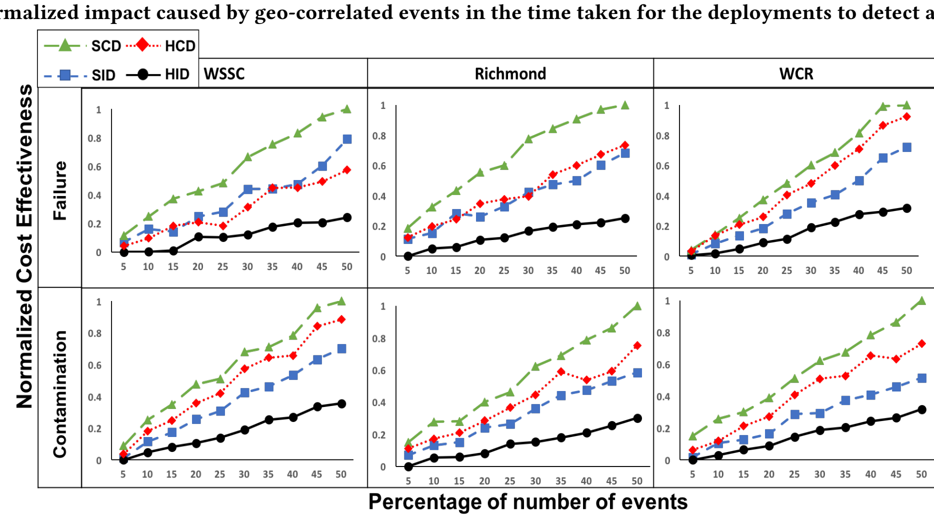
[1] American Water Works Association. [n. d.]. Buried no longer: confronting America's water infrastructure challenge. http://www.awwa.org/infrastructure.
[2] L. Begnudelli, B. F. Sanders, and S. F. Bradford. 2008. Adaptive Godunov-based model for flood simulation. *Hydraulic Eng.* (2008).
[3] T. Brinkhoff. 2005. City population. (2005).
[4] R. Du, C. Fischione, and M. Xiao. 2016. Flowing with the water: On optimal monitoring of water distribution networks by mobile sensors. In *INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications, IEEE.* IEEE, 1–9.
[5] Environmental Protection Agency, US. 2001. Potential contamination due to cross-connections and backflow and the associated health risks. *Office of Ground Water and Drinking Water, Washington DC.* (2001).
[6] R. Fletcher and M. Chandrasekaran. 2008. SmartBallâĎć: a new approach in pipeline leak detection. In *2008 7th International Pipeline Conference.* American Society of Mechanical Engineers, 117–133.
[7] M. Haklay and P. Weber. 2008. Openstreetmap: User-generated street maps. *Pervasive Computing* (2008).
[8] Q. Han, P. Nguyen, R. T. Eguchi, K-L. Hsu, and N. Venkatasubramanian. 2017. Toward an integrated approach to localizing failures in community water networks. In *Distributed Computing Systems (ICDCS), 2017 IEEE 37th International Conference on.* IEEE, 2505–2506.
[9] http://emps.exeter.ac.uk/engineering/research/cws/downloads/benchmarks/. [n. d.]. Centre of Water Systems University of EXETER. ([n. d.]).
[10] S. Kartakis, E. Abraham, and J. A. McCann. 2015. Waterbox: A testbed for monitoring and controlling smart water networks. In *CPS for Smart Water Networks.*
[11] B. Korte, J. Vygen, B. Korte, and J. Vygen. 2012. *Combinatorial optimization.* Vol. 2. Springer.
[12] T. T-T. Lai et al. 2013. Mapping hidden water pipelines using a mobile sensor droplet. *ACM Transactions on Sensor Networks (TOSN)* 9, 2 (2013), 20.
[13] A. Lambert. 2001. What do we know about pressure-leakage relationships in distribution systems. In *leakage control and water distribution system mgmt.*
[14] C-C. Liao et al. 2014. SAIS: Smartphone augmented infrastructure sensing for public safety and sustainability in smart cities. In *Proceedings of the 1st International Workshop on Emerging Multimedia Applications and Services for Smart Cities.* ACM, 3–8.
[15] A. Marjovi, A. Arfire, and A. Martinoli. 2015. High resolution air pollution maps in urban environments using mobile sensor networks. In *Distributed Computing in Sensor Systems (DCOSS), 2015 International Conference on.* IEEE, 11–20.
[16] D. Misiūnas. 2008. *Failure Monitoring and Asset Condition Assessment in Water Supply Systems.*
[17] G. L. Nemhauser et al. 1978. An analysis of approximations for maximizing submodular set functions. *Mathematical Programming* (1978).
[18] N. Oliker and A. Ostfeld. 2015. Inclusion of mobile sensors in water distribution system monitoring operations. *Journal of Water Resources Planning and Management* 142, 1 (2015), 04015044.
[19] L. Perelman and A. Ostfeld. 2013. Operation of remote mobile sensors for security of drinking water distribution systems. *Water research* 47, 13 (2013), 4217–4226.
[20] L. S. Perelman, W. Abbas, X. Koutsoukos, and S. Amin. 2016. Sensor placement for fault location identification in water networks: A minimum test cover approach. *Automatica* 72 (2016), 166–176.
[21] R. Pérez et al. 2009. Pressure sensor distribution for leak detection in Barcelona water distribution network. *Water science & tech: water supply* (2009).
[22] A. Rasekh et al. 2014. Operation of mobile sensors for monitoring municipal drinking water distribution systems. In *World Environmental and Water Resources Congress 2014.* 362–367.
[23] L. A. Rossman et al. 2000. EPANET 2: users manual. (2000).
[24] B. F. Sanders and L. Begnudelli. 2010. BreZo: A hydrodynamic flood simulation algorithm.
[25] J. R. Shewchuk. 1996. Triangle: Engineering a 2D quality mesh generator and Delaunay triangulator. In *Applied computational geometry towards geometric engineering.* Springer, 203–222.
[26] I. Stoianov, L. Nachman, S. Madden, et al. 2007. PIPENET: A wireless sensor network for pipeline monitoring. In *Sensor Networks.*
[27] M.A. Suresh et al. 2013. On Event Detection and Localization in Acyclic Flow Networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems* 43, 3 (2013).
[28] J. Thornton, R. Sturm, and G. Kunkel. 2008. *Water Loss Control* (2nd ed.). McGraw-Hill Professional.
[29] P. Venkateswaran, Q. Han, R. T. Eguchi, and N. Venkatasubramanian. 2018. Impact Driven Sensor Placement for Leak Detection in Community Water Networks. In *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems (ICCPS '18).*
[30] A. J. Whittle, L. Girod, A. Preis, et al. 2010. WATERWISE: A testbed for continuous monitoring of the water distribution system in singapore. In *Water Distribution Systems Analysis.*
[31] www.elevationmap.net. [n. d.]. Elevation Map. ([n. d.]).
[32] www.instrumart.com/categories/6123/level transmitters. [n. d.]. Water Level In-situ Sensor Price. ([n. d.]).
[33] Q. Zhu et al. 2016. Upload planning for mobile data collection in smart community Internet-of-Things deployments. In *2016 IEEE International Conference on Smart Computing (SMARTCOMP).* IEEE, 1–8.



(a) Average detection and localization times of the deployments for geo-correlated events



(b) Average normalized impact caused by geo-correlated events in the time taken for the deployments to detect and localize them



(c) Normalized Cost Effectiveness of the sensor deployments for geo-correlated events

Figure 10: Comparison of Approaches for Geo-correlated events

APPENDIX

A PROOF OF THEOREM 1

Proof - We see from the formulation of the event utility functions $\mathcal{U}^{stat}, \mathcal{U}^{mob}$ that they depend on the impact caused by the event and the time taken to detect and localize it. Since, the impact formulation derived from Section 3.3 is independent of the sensor deployment, the submodularity of the event utility functions depends on the detection time $\mathcal{M}_{dt}$ and traversal time $\mathcal{M}_{tt}$.

For the static event utility function $\mathcal{U}^{stat}$, consider two placement sets $\mathcal{P}^{stat} \subseteq \mathcal{Q}^{stat} \subseteq \mathcal{S}^{stat}$. Given an event $e_j \in \mathcal{E}$ that can be detected by a static sensor installed at location i such that $s_i^{stat} \in \mathcal{S}^{stat} \setminus \mathcal{P}^{stat}$. Depending on the time taken for s_i^{stat} to detect e_j, there are three cases -

(1) $\mathcal{M}_{dt}[s_i^{stat}, e_j] \geq min(\mathcal{M}_{dt}[\mathcal{P}^{stat}, e_j])$.
This implies $min\ (\mathcal{M}_{dt}[\mathcal{P}^{stat} \cup \{s_i\}, e_j]) = min(\mathcal{M}_{dt}[\mathcal{P}^{stat}, e_j])$ and $min\ (\mathcal{M}_{dt}\ [\mathcal{Q}^{stat} \cup \{s_i\}, e_j]) = min(\mathcal{M}_{dt}[\mathcal{Q}^{stat}, e_j])$.
Hence, $\mathcal{U}^{stat}(\mathcal{P}^{stat} \cup \{s_i^{stat}\})$ - $\mathcal{U}^{stat}(\mathcal{P}^{stat}) = \mathcal{U}^{stat}(\mathcal{Q}^{stat} \cup \{s_i^{stat}\}) - \mathcal{U}^{stat}(\mathcal{Q}^{stat}) = 0$.

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

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

Hence, we show that $\mathcal{U}^{stat}$ is submodular. Similarly we can show that $\mathcal{U}^{mob}$ is submodular.

B NETWORK LAYOUT OF RICHMOND AND WOLF CORDERA RANCH

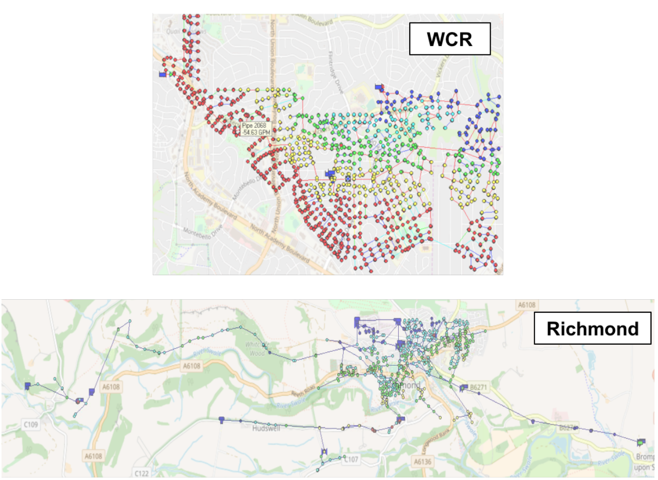
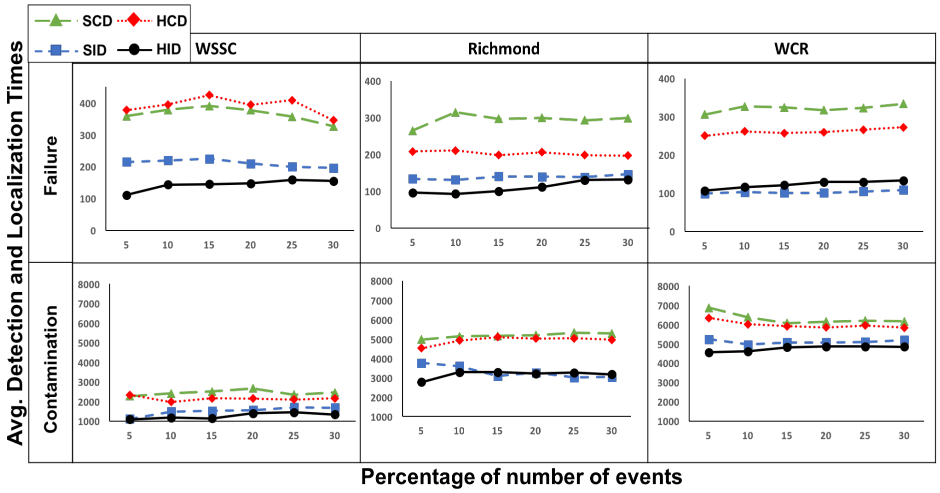
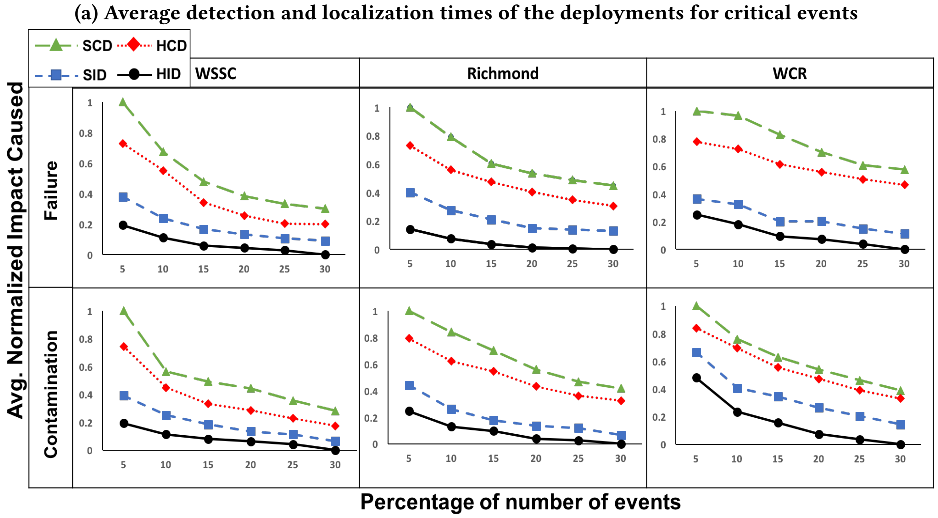


Figure 11: Network Layouts of WCR and Richmond

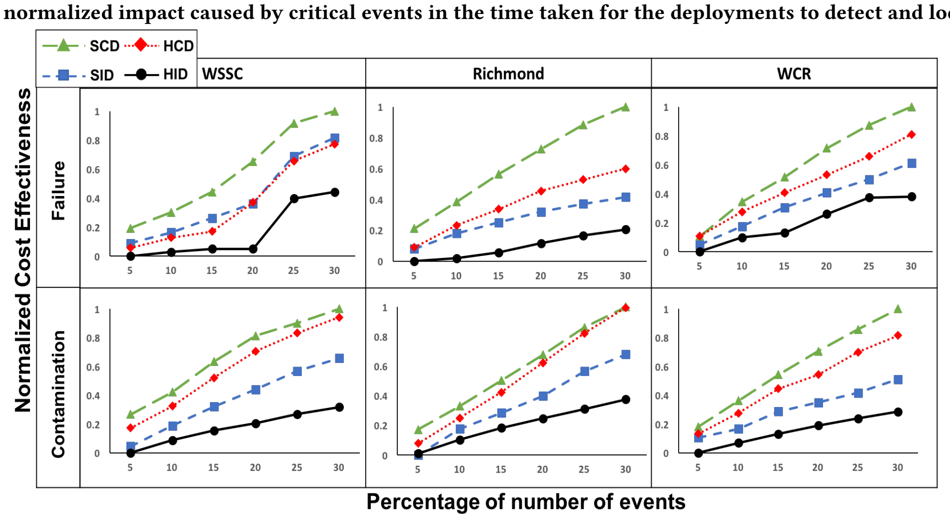
C CRITICAL EVENTS EXPERIMENTAL STUDY



(a) Average detection and localization times of the deployments for critical events



(b) Average normalized impact caused by critical events in the time taken for the deployments to detect and localize them



(c) Normalized Cost Effectiveness of the sensor deployments for critical events

Figure 12: Comparison of Approaches for Critical events