UNIVERSITY OF CALIFORNIA,
IRVINE

Enabling Resilience in Cyber-Physical-Human Water Infrastructures

DISSERTATION

submitted in partial satisfaction of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

in Networked Systems

by

Qing Han

Dissertation Committee:
Professor Nalini Venkatasubramanian, Chair
Professor Nikil Dutt
Professor Marco Levorato
Professor Sharad Mehrotra

2019

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

Page

List of Algorithms

ACKNOWLEDGMENTS

First, I would like to thank my advisor, Professor Nalini Venkatasubramanian, who offered me a precious opportunity to join her group to grow in research and participate in various wonderful conferences to gain experiences beyond our research fields for my personal and career developments, and most importantly, for all the encouragement and guidance throughout the years

I am also grateful for my various mentors from my Ph.D. studies: my dissertation committee – Professor Nikil Dutt, Professor Marco Levorato and Professor Sharad Mehrotra from UCI; Daniel Hoffman from Montgomery County, MD; Lingyi Zhang from WSSC, DC.

I would like to thank my co-authors: Ronald Eguchi, Phu Nguyen and Kuo-Lin Hsu, and various other peers and friends I worked closely with during my Ph.D. study: Praveen Venkateswaran, Guoxi Wang, Qiuxi Zhu, Nailah Alhassoun, Kyle Benson, Hang Nguyen, Fangqi Liu, Yusuf Sarwar, and Georgios Bouloukakis, for their valuable feedback and support. I further thank everyone in the Distributed Systems Middleware (DSM) group and the Information Systems Group (ISG) at the University of California, Irvine.

Last but not least, I would like to express my deepest gratitude toward my family and friends for their unconditional love and patience to be alongside me on this journey, and endless support in the most challenging times.

My Ph.D. research was supported in part by the National Science Foundation (NSF) award No. CNS-1143705, CNS-0958520 and CNS-1450768, and the Donald Bren School of Information and Computer Sciences (ICS) at the University of California, Irvine.

CURRICULUM VITAE

Qing Han

EDUCATION

Doctor of Philosophy in Networked Systems **2019**
University of California, Irvine *Irvine, California*

Master of Science in Communications and Signal Processing **2013**
Imperial College London *London, UK*

Bachelor of Science in Telecommunications Engineering **2012**
Queen Mary University of London *London, UK*

Bachelor of Science in Telecommunications Engineering **2012**
Beijing University of Posts and Telecommunications (BUPT) *Beijing, China*

RESEARCH EXPERIENCE

Graduate Student Researcher **2015–2019**
University of California, Irvine *Irvine, California*

Graduate Student Researcher **2012–2013**
Imperial College London *London, UK*

PROFESSIONAL EXPERIENCE

Software Engineer Intern **Summer 2018**
Facebook, Inc. *Menlo Park, California*

Software Engineer Intern **Summer 2017**
Facebook, Inc. *Menlo Park, California*

Innovation Fellow **Summer 2016**
Montgomery County Innovation Program *Montgomery County, Maryland*

Network Engineer **Summer 2014**
Ericsson *Beijing, China*

LEADERSHIP EXPERIENCE

President of Student Union **2010-2011**
International School, BUPT *Beijing, China*

TEACHING EXPERIENCE

Teaching Assistant **2017–2019**
University of California, Irvine *Irvine, California*

Graduate Reader **2015–2017**
University of California, Irvine *Irvine, California*

REFEREED CONFERENCE PUBLICATIONS

AquaEIS: Middleware Support for Event Identification in Community Water Infrastructures **Dec 2019**
ACM Middleware

Enabling State Estimation for Fault Identification in Water Distribution Systems under Large Disasters **Oct 2018**
IEEE International Symposium on Reliable Distributed Systems (SRDS) (*Best Paper*)

Toward An Integrated Approach to Localizing Failures in Community Water Networks **June 2017**
IEEE International Conference on Distributed Computing Systems (ICDCS)

ABSTRACT OF THE DISSERTATION

Enabling Resilience in Cyber-Physical-Human Water Infrastructures

By

Qing Han

Doctor of Philosophy in Networked Systems

University of California, Irvine, 2019

Professor Nalini Venkatasubramanian, Chair

Rapid urbanization and growth in urban populations have forced community-scale infrastructures (e.g., water, power and natural gas distribution systems, and transportation networks) to operate at their limits. Aging (and failing) infrastructures around the world are becoming increasingly vulnerable to operational degradation, extreme weather, natural disasters and cyber attacks/failures. These trends have wide-ranging socioeconomic consequences and raise public safety concerns. In this thesis, we introduce the notion of cyber-physical-human infrastructures (CPHIs) - smart community-scale infrastructures that bridge technologies with physical infrastructures and people. CPHIs are highly dynamic stochastic systems characterized by complex physical models that exhibit regionwide variability and uncertainty under disruptions. Failures in these distributed settings tend to be difficult to predict and estimate, and expensive to repair. Real-time fault identification is crucial to ensure continuity of lifeline services to customers at adequate levels of quality. Emerging smart community technologies have the potential to transform our failing infrastructures into robust and resilient future CPHIs.

In this thesis, we explore one such CPHI - community water infrastructures. Current urban water infrastructures, that are decades (sometimes over a 100 years) old, encompass diverse geophysical regimes. Water stress concerns include the scarcity of supply and an increase

in demand due to urbanization. Deterioration and damage to the infrastructure can disrupt water service; contamination events can result in economic and public health consequences. Unfortunately, little investment has gone into modernizing this key lifeline.

To enhance the resilience of water systems, we propose an integrated middleware framework for quick and accurate identification of failures in complex water networks that exhibit uncertain behavior. Our proposed approach integrates IoT-based sensing, domain-specific models and simulations with machine learning methods to identify failures (pipe breaks, contamination events). The composition of techniques results in cost-accuracy-latency tradeoffs in fault identification, inherent in CPHIs due to the constraints imposed by cyber components, physical mechanics and human operators. Three key resilience problems are addressed in this thesis; isolation of multiple faults under a small number of failures, state estimation of the water systems under extreme events such as earthquakes, and contaminant source identification in water networks using human-in-the-loop based sensing. By working with real world water agencies (WSSC, DC and LADWP, LA), we first develop an understanding of operations of water CPHI systems. We design and implement a sensor-simulation-data integration framework **AquaSCALE**, and apply it to localize multiple concurrent pipe failures. We use a mixture of infrastructure measurements (i.e., historical and live water pressure/flow), environmental data (i.e., weather) and human inputs (i.e., twitter feeds), combined and enhanced with the domain model and supervised learning techniques to locate multiple failures at fine levels of granularity (individual pipeline level) with detection time reduced by orders of magnitude (from hours/days to minutes). We next consider the resilience of water infrastructures under extreme events (i.e., earthquakes) - the challenge here is the lack of apriori knowledge and the increased number and severity of damages to infrastructures. We present a graphical model based approach for efficient online state estimation, where the offline graph factorization partitions a given network into disjoint subgraphs, and the belief propagation based inference is executed on-the-fly in a distributed manner on those subgraphs. Our proposed approach can isolate 80% broken pipes and 99% loss-of-service to end-users during an

earthquake.

Finally, we address issues of water quality - today this is a human-in-the-loop process where operators need to gather water samples for lab tests. We incorporate the necessary abstractions with event processing methods into a workflow, which iteratively selects and refines the set of potential failure points via human-driven grab sampling. Our approach utilizes Hidden Markov Model based representations for event inference, along with reinforcement learning methods for further refining event locations and reducing the cost of human efforts.

The proposed techniques are integrated into a middleware architecture, which enables components to communicate/collaborate with one another. We validate our approaches through a prototype implementation with multiple real-world water networks, supply-demand patterns from water utilities and policies set by the U.S. EPA. While our focus here is on water infrastructures in a community, the developed end-to-end solution is applicable to other infrastructures and community services which operate in disruptive and resource-constrained environments.

Chapter 1

Introduction

In this chapter, we motivate this thesis by first introducing the current status of our community services and urban infrastructures that are aging and failing, and the notion of cyber-physical-human infrastructures (CPHIs) that will be used throughout this thesis. A major issue is to enable quick and accurate fault identification in such CPHI systems prone to unforeseen failures and external malicious attacks. We illustrate key challenges in fault identification and describe our efforts towards addressing these problems using community water distribution infrastructures as a driving use case. Because water as a growing scarce natural resource is a critical lifeline to our daily life. Indeed, there are few things more important to a functioning society than access to clean water [106]. In particular, to enhance resilience of water systems, we propose a sensor-simulation-data integration platform and strive to fully exploit the potential of this platform by addressing three key resilience problems.

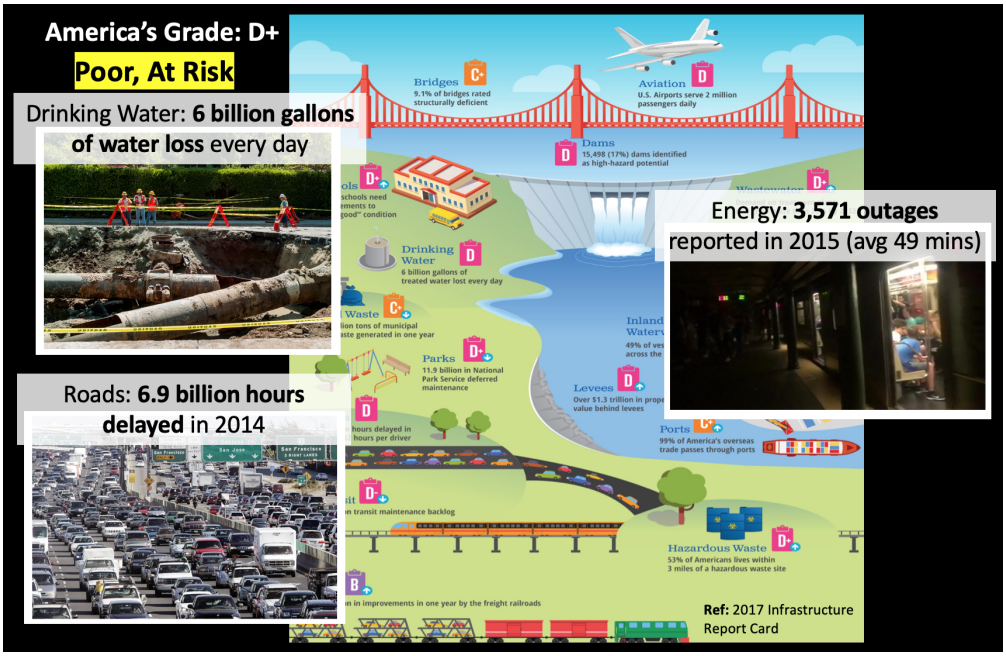


Figure 1.1: America's vulnerable infrastructures in 2017 (grade - $D+$).

1.1 Cyber Physical Human Infrastructures (CPHIs)

More than half of the World's population now lives in urban areas [29]. The United Nations estimates that this shift from a primarily rural to a primarily urban population is projected to continue for the next couple of decades (68% projected by 2050) [11]. With the surge in urban populations and the rapid development of the global economy, community-scale infrastructures (e.g., water, natural gas, power distribution systems and transportation network) are being forced to operate at their capacity limits. As an example shown in Fig. 1.1, America's infrastructure scored a $D+$ based on the 2017 infrastructure report card [9]. It is reflecting that the infrastructure is in poor to fair condition and mostly below standard, with many components approaching the end of their service life. As a mainstay of the national economy, security and societal functioning, these lifelines involve the production, transport and management of essential goods and services. In recent years, these infrastructures are subject to stress caused by a variety of factors - aging, rapid urbanization and environmental change. As a result, they have become increasingly complex and often vulnerable to failures

due to natural (e.g., hurricane and earthquake), technological (e.g., cyber attacks/failures) and man-made (e.g., human errors) events.

In this thesis, we introduce the notion of cyber-physical-human infrastructures (CPHIs) - smart community-scale infrastructures that are instrumented, networked, controlled and regionally distributed with human embedded in their operations. CPHI systems bridge technologies with physical infrastructures and people, and they must operate dependably, safely, securely, efficiently and in real time [104]. As engineered systems, CPHIs are highly dynamic stochastic systems characterized by complex physical models that exhibit region-wide variability and uncertainty under disruptions. Unfortunately, failures in these distributed settings tend to be difficult to predict and estimate, and expensive to repair, which could result in community disruptions ranging from temporary interruptions in services to extended loss of business and mass relocation of residents. Modernizing these infrastructures is the key to a resilient and sustainable future, which entails revolutionizing their design, deployment and operation with new architectures and added intelligence. The American Society of Civil Engineers estimates that investment over $2 trillion are required by 2025 to improve the resilience to critical deficiencies in the nation's roads, railroads, drinking water distribution, wastewater management, electrical grids, etc [8]. Resilience is, in general, defined as "the ability to prepare and plan for, absorb, recover from, and successfully adapt to adverse events" by the National Academies of Science [37]. Specifically in the context of urban infrastructures, it is the ability to withstand disruption events with little loss in function, and rapidly and efficiently restore functionality if loss incurred, according to the department of homeland security. Emerging smart community technologies for sensing, computing and communication are being embedded in a multitude of objects and structures in the physical environment - this has shown great potential to transform our failing infrastructures into robust and resilient future CPHIs [104].

As infrastructure systems become more and more autonomous with advances in cyber tech-

nologies, humans play critical roles in their operations. Leveraging citizens working together with cyber and physical worlds as a part of the execution event is a key component of efficiently running community-scale systems [28]. Urban infrastructures serve as lifelines to satisfy human needs, the human, in turn, can help a system make intelligent decisions and achieve its goals (ultimately also citizens' goals) [117]. Human knowledge and inputs are indispensable in settings where sensing needs are uncertain and expensive. For example in firefighting, an expert understanding of fire protection engineering and spread model will help first responders implement execute suppression and rescue operations. During power and water outage, technicians are dispatched to deal with issues that can only be solved with human intervention.

Designing, operating and maintaining CPHI systems for a particular service in a safe manner during its entire lifetimes has been recognized a critical issue worldwide [130]. Their regional scope, system wide dynamics and long-term nature distinguishes them from more standalone cyber-physical systems (CPSs), e.g., medical devices, autonomous vehicles and manufacturing components. This has presented a multitude of unique challenges including operation - optimization under multiple operational timescales is critical to enable (near) real-time control and provisioning decisions; architecture - new architectural principles are required to mediate new supply-demand dynamics driven by the growing urbanization; and adoption - human, organizational and ecosystem constraints should be considered as an integral part of the design and operational process. They can only be successfully addressed through a multidisciplinary approach, integrating together techniques from engineering, computer science, and social and behavioral sciences.

1.2 Fault Identification in CPHI Systems

Cyber-Physical-Human Infrastructure systems interact with the physical world and are ubiquitous in our daily life. They must operate reliably in the face of unforeseen failures and external malicious attacks. Examples of CPHI systems include transportation, utility (power, water and gas distribution), and communication networks. Many of these systems are prone to damages in their physical infrastructures, and cyber attacks on their data management and communication layers. CPHIs, as large-scale, aging and deteriorating systems, suffer from unique vulnerabilities for which appropriate fault identification techniques need to be developed. Today, the detection of anomalous events in CPHIs is time consuming and often takes hours or even days [118]; these failures can cause irreparable harm to people that depend on them. For example, an electric transformer fire caused massive blackout in Manhattan (U.S.) in 2019. The utility company spent 5 hours to identify the exact cause of the operational failure and weeks to fully recover the system, which left $72,000$ customers in the dark on that day and more than $50,000$ customers facing a second power outage one week later [12]. A failed insulator paralyzed the D.C. Metro system (U.S.) in 2015. The entire subway system was shut down for a whole day in order to perform system-wide inspections and repair [6]. A SCADA (supervisory control and data acquisition) system breach released 1 million liters of untreated sewage into local waterways at Maroochy Water Services (Australia) in 2000. It took over 3 months to discover that someone was hacking into their SCADA system and deliberately causing the problems [116].

The promise of cyber-physical system (CPS) is *pushed* by several recent trends: the proliferation of low-cost, low-power and increased-capability sensing platform (e.g., Internet of Things (IoT)); the availability of high-speed, high-capacity and flexible computing techniques (e.g., machine learning, big data, deep learning); the revolution of wireless communication (e.g., 5G); continuing improvements in energy management (e.g., renewable energy sources). The need of CPS innovations is also being *pulled* by increasingly failure-prone infrastructures

revealing that the technology base to enhance large-scale lifeline CPHI is seriously lacking [104]. CPHI systems are complex distributed interconnected networks. This web of interconnecting distribution/transmission links makes the system robust and reliable under normal operation. However a disruption in one location can quickly propagate across the system in complex ways leading to cascading failures and widespread service outages. It is of utmost importance to quickly and accurately identify and contain the failures to avoid escalation of problems, which ensure the health, safety and well-being of citizens.

1.3 Key Challenges

Today's urban infrastructure has the potential to evolve into a future cyber-enabled physical systems with human-in-the-loop, that can leverage sensing, networking, computing and domain knowledge to enable real-time fault identification. Such real-time fault identification frameworks are required for supporting reliable and timely decisions in the presence of faults, and maintaining adequate levels of quality and quantity of flows through critical infrastructures. This would achieve significant reductions in productivity losses and socioeconomic costs resulting from abnormal events. Nevertheless, to date, no city has deployed a comprehensive community-scale fault identification solution for its infrastructures. The actual implementation of real-time fault identification has been hindered by the following challenges.

(C1) Uncertainty and variability: Physical damage (e.g., bridge collapse, transformer overheating and water main break) and their impacts on physical-world infrastructures are unpredictable. At the same time, the cyber- assets (e.g., SCADA and IoT sensors) used for monitoring, networking and controlling are also subject to anomalous variability in communications and operational fidelity. Generally speaking, faults can originate at any point throughout an infrastructure network at any time. In order to identify the failures, the

search space can therefore be very large (roughly number of combinations of network nodes times number of historical time steps considered), making it intractable or even prohibitive to solve the problem uniquely. Furthermore, the availability of sensing devices is limited because of high deployment/maintenance costs and measurement precision. The lack of spatial coverage increases the uncertainty, where the limited observations make it hard to distinguish between events and can sometimes result in unavoidable ambiguous events.

(C2) Complex engineered system: CPHIs are complex distributed systems, driven by complex supply-demand models subject to strong operational constraints. They must obey the underlying physical mechanisms that are non-linear and non-normal, and in many cases these systems are looped which can provide better continuity of service under normal operations, yet render them prone to cascading failures under stress conditions. CPHI systems have, therefore, exhibited highly dynamic behavior and complex dependencies within the same infrastructure or between different infrastructures, creating computational challenges in isolating and identifying failures. A key concern is to understand and capture/model how the system behaves in the presence of faults. Any methods that do not exploit the compatibility of the observations with the underlying physical process, then, will be necessarily ineffective against cyber-physical disruptions that change the system performance.

(C3) Human engagement for decision supports: Cyber technology alone cannot transform the urban infrastructures; necessary strategies must be included to optimally accommodate human and community-wide engagement in decision supports [28]. Additional sensing in the form of human operators can help reduce the aforementioned uncertainty; this process involves cost and time. Therefore, to effectively build CPHI systems with human-in-the-loop, it is important to be able to describe what a person can do, when and where their engagement are needed, as well as to address issues related to unpredictability, accuracy and privacy of human inputs.

(C4) Cost-accuracy-latency tradeoffs: Fault identification in CPHI systems entails a

complex tradeoff between the operational costs, accuracy bounds of identifications and execution latencies. The aforementioned challenges, **C1** and **C2**, draw into question the effort of spending great computational resources to identify the exact failure locations. Rather, it may be important to acknowledge the limitations inherent in this complex distributed setting, and consider methods that produce a usable, approximate result with a set of tractable number of solutions in time. Human capabilities can be leveraged to help decision-making, however the cost of human involvement should be considered (Challenge **C3**). This dictates that a trade-off analysis must be included to develop practical solutions and enable real-time fault identification.

1.4 Driving Use Case - Community Water Infrastructures

The rapid population growth and climate change are straining our limited water resources and services. Freshwater consumption is expected to rise 25% by 2030. In addition, not only are these resources limited in supply, they are often poorly managed where almost 60% of the water is wasted due to leaks and inefficient techniques [134]. Across the global (Fig. 1.2), many of the water infrastructures have been deployed over long periods (often decades, centuries), and the condition of these infrastructures is deteriorating with time and becomes inadequate to provide services to growing communities. For example, in U.S., drinking water infrastructure is nearing the end of its intended lifespan, with much of the infrastructure dating back to the 1940s; the intended lifespan however was 75-100 years. Furthermore, they encompass diverse geophysical regimes and are subject to extreme geologic conditions (e.g., in floodplains, along fault lines and proximate to urban areas vulnerable to malicious attack).

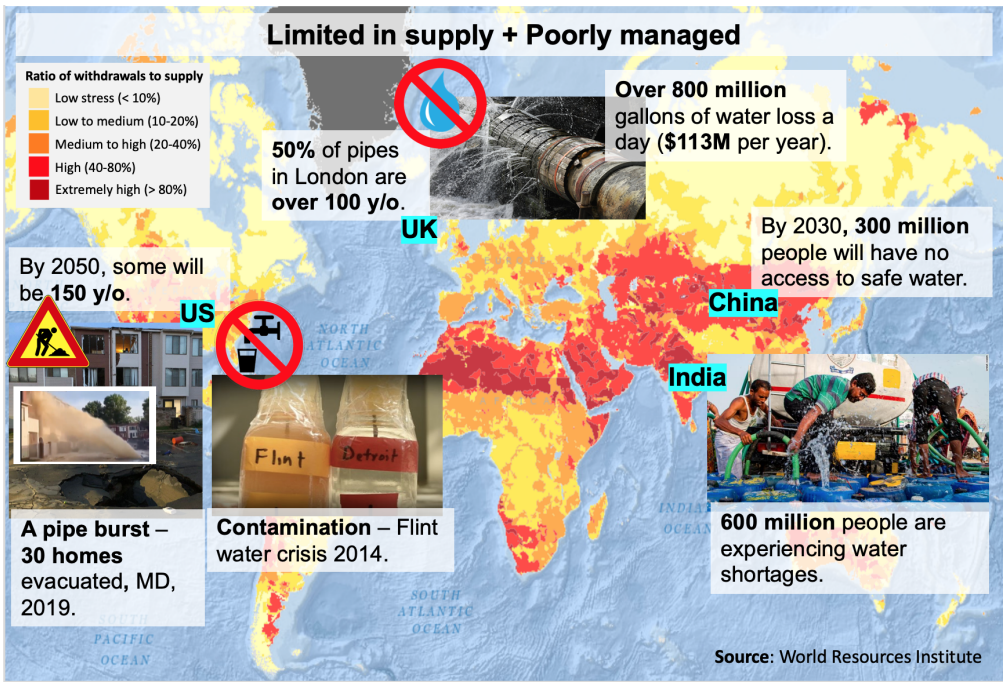


Figure 1.2: Water is an important but vulnerable lifeline; they are not only limited in supply, but also poorly managed [2].

Water distribution systems in communities today consist of large and diverse networks with many different components (pipes, junctions, pumps, valves, tanks and reservoirs); they are inherently vulnerable to small failures, extreme events and demand/supply variabilities [44]. Physical damage to water infrastructures (e.g., pipe breaks) can disrupt water service to citizens; contamination events could result in economic and public health consequences and long-lasting psychological impacts. Many cities are currently experiencing an increasing number of failing services, where water outages and interruptions have become a daily occurrence. Due to the scarcity of resources and failure-prone infrastructures, it is imperative to implement smarter ways in which water is conserved. Despite many significant innovations applied to public infrastructures, water infrastructure has not received the same level of attention and investment. For example, in U.S. during the period from 2016 through 2025, there are an estimated $2 trillion investment for transit infrastructures, $934 billion for electricity, yet $150 billion for water and wastewater [85].

Current status of water facilities: Today, cyber instrumenting of water systems is

severely lacking and sparse at best; this is due to the fact that such buried assets were deployed decades ago. Water meters are instrumented primarily for billing, SCADA systems, when available, are deployed at pump stations above ground, and water quality monitoring is mainly present at water storage and treatment plants. The network of hundreds or thousands of pipelines is not actively monitored since devices and networks are expensive to install/deploy and maintain. Furthermore, many cities do not yet have the much needed data (e.g., water demand history, population distribution and risk analysis) to compute an optimal sensor placement [43, 24]. Current sensing technologies also present limitations. While water quality sensors can detect different types of chemical/microbial contaminants, they often yield binary (yes/no) indications of possible contaminant presence and cannot provide concentration measurements of arbitrary contaminants [44]. Lab examinations on water samples are required to obtain detailed information - the realistic response times could vary from 6 to 24 hours [90].

Fault identification in water infrastructures - challenges: These underground infrastructures with buried assets have presented the aforementioned challenges (**C1** - **C4**); in fact, these challenges are exacerbated on such infrastructures due to inaccessibility. A deeper inspection of water infrastructures points to some inherent systemic complexities. Water systems are large-scale complex networks, where flow patterns are driven by time-varying water demands and distribution systems are looped, resulting in complex interdependencies. These interdependencies may be especially pronounced in the presence of failures, e.g., the mixing and dilution of contaminants. Mathematically, fault identification problem is the canonical "inverse problem", where we aim to infer the root cause from a set of observations. Inverse problems are often ill-posed, where multiple solutions or near-solutions may exist [120]. Under such dynamic unpredictable environmental changes (challenges **C1** and **C2**), we argue that one cannot necessarily solve the problem as a formal inverse problem and spend great computational effort for a solution as being the most likely. Instead, we focus on identifying a collection of possible solutions, with the recognition that even partial knowledge can inform

decision-making. This identified set could then be considered further using other methods or data. We also believe that humans will be more than just users of CPHI systems and play critical roles as active agents in the operation and evolution of these systems. How those roles play out in narrowing down the set of possible solutions is a prime challenge on the identification of failures (challenge **C3**).

All of the above point to the need for an integrated view of the state of CPHI systems to support cost-accuracy-latency tradeoffs inherent in fault identification, which enables improved real-time event identification frameworks for managing failure-prone water infrastructures of the future. Such a framework will provide real-time situational awareness and proactively detection of anomalies and concerns, which would trigger timely feedback control and operational planning.

1.5 Thesis Contributions and Organization

This thesis aims to address the above-mentioned challenges (Section 1.3) in rapidly identifying failures in mission-critical urban infrastructures, using community water systems as a driven use case. We propose an integrated approach, **AquaSCALE**, to fuse multiple (incomplete) sources of information for fast fault identification in these complex distributed settings. AquaSCALE integrates multiple sensing modalities (devices, human-as-a-sensor), computing (ML analysis) and domain knowledge (simulations and physics-based models) to enable an observation, analysis and adaptation loop in the system. This is required since the limited sensing devices in today's lifelines only capture phenomena spread (e.g., pressure drop due to pipe leaks, contamination due to illegal wastewater discharge), not the fault source, in the networked system. Furthermore, sensor data exhibits inaccuracies; techniques to process the data add uncertainties. Additional sensing in the form of human operators can help reduce these uncertainties by strategic sampling at selected points in the network - this

process involves cost and time. Holistically, this thesis has produced a suite of discussions and analytics to explore resilience of CPHI systems in the face of disruptions and failures from different causes. While our focus here is on water infrastructures in a community, the developed end-to-end solution will be portable to other infrastructures and community services which operate in disruptive and resource-constrained environments.

The following is the overall organization and research contributions of the thesis.

• Chapter 2 surveys related work from both practical and theoretical aspects.

• Chapter 3 proposes our AquaSCALE approach in greater detail. We present the architecture of the AquaSCALE system and leverage it to address the cross-layer concerns at multiple levels of CPHI systems (devices, networks, data and compute). AquaSCALE is designed as a workflow based system comprised of multiple components, which help enhance understanding of underlying physical dynamics, and enable a holistic analysis on heterogeneous data using a mixture of techniques.

• Chapter 4 describes our integrated approach to localizing multiple simultaneous failures in a CPHI system. It can isolate failures at fine level of granularity (individual pipeline level) with high level of accuracy with detection time reduced by orders of magnitude (from hours/days to minutes). Our key contributions include a two-phase process using a profile generated offline for quick estimation and an online live data integration for further localization; a plug-and-play analytic engine for the selection and integration of modeling and learning techniques; extensive evaluations under diverse failure scenarios using real-world water networks.

• Chapter 5 considers the resilience of a CPHI system under extreme events (e.g., earthquake), where threats to infrastructures are highly unpredictable. It focuses on real-time state estimation that produces estimates of current operating states and helps detect, locate and prevent secondary failures. Our key contributions include network topology processing to model the complex interdependence using factor graph representation; a multi-phase approach for improving the speed and accuracy of state estimate on the constructed graph;

design of a series of experiments to explore the performance of the proposed approach.

• Chapter 6 explores the incorporation of human-in-the-loop for fault source identification in a CPHI system, with the support of inherent cost-accuracy-latency tradeoffs. It can significantly reduce the number of human-driven sampling cycles, while ensuring localization accuracy. Our key contributions include event-driven profile generation to capture the physical nature of phenomena spread; online iterative event processing to iteratively select and refine potential failures via human-driven grab samples; real-world performance evaluation using a complete ensemble of anomalous events.

• Chapter 7 concludes the dissertation with lessons learned and looks forward to future research problems we must address to enable our vision of resilient CPHI systems.

Chapter 2

Related Work in Water Infrastructure Resilience

In this chapter, we introduce relevant work to provide appropriate background for this thesis. We survey the related work on enabling water infrastructure resilience during its lifecycle. We categorize this lifecycle into three phases including planning, operation and adaptation to events (Fig. 2.1). We first describe several existing tools that have been widely used for modeling water infrastructures.

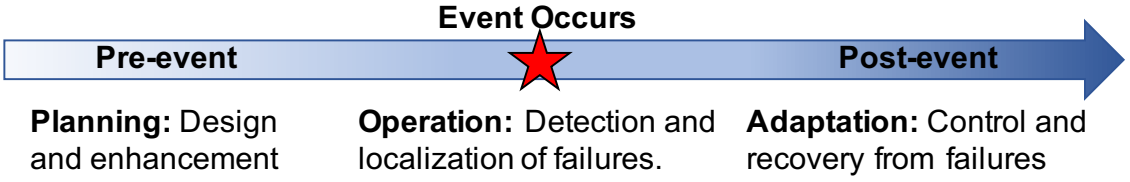


Figure 2.1: Related work is categorized into planning, operation and adaptation phases based on the lifecycle of services.

2.1 Existing Tools for Today

In large city-scale infrastructures, effective modeling and understanding of lifeline operations is key to enabling fast fault identification. Agencies such as U.S. Environmental Protection Agency (EPA) have been focused on the design of tools to model water distribution systems and to understand the movement and fate of drinking water constituents.

EPANET (Fig. 2.2) is a commercial-grade simulator used throughout the world to model drinking water distribution systems, developed by the U.S. EPA [108]. It supports hydraulic modeling, water quality modeling, and water security and resilience modeling in pressurized pipe networks. Today, engineers and consultants use EPANET for various applications, e.g., design and shape new infrastructure, optimize operations of tanks and pumps, identify small and silent pipe breaks, reduce energy consumption, and investigate water quality problems. There are also a number of EPANET based extensions, such as EPANET-MSX specific for modeling contamination threats [113] and EPANET-RTX for real-time analytics [45].

However, EPANET and its extensions are not designed to handle large or massive failures that can result in inadequate pressure or rapid changes in the operations. They use demand-driven (DD) hydraulic model as the simulation engine, assuming that customer demands are always met event if the pressure is insufficient to deliver potable water. In the context of extreme events (e.g., earthquakes), this assumption usually does not hold. Large events often cause increasing number and/or severity of pipe failures, which leads to low pressure conditions and end-users may lose the access to sufficient water supply. WNTR (water network tool for resilience) is a tool for assessing the resilience of drinking water systems to disasters, recently developed by the Sandia National Laboratories and U.S. EPA [68]. WNTR uses the pressure-driven demand (PDD) model, where the amount of water supplied

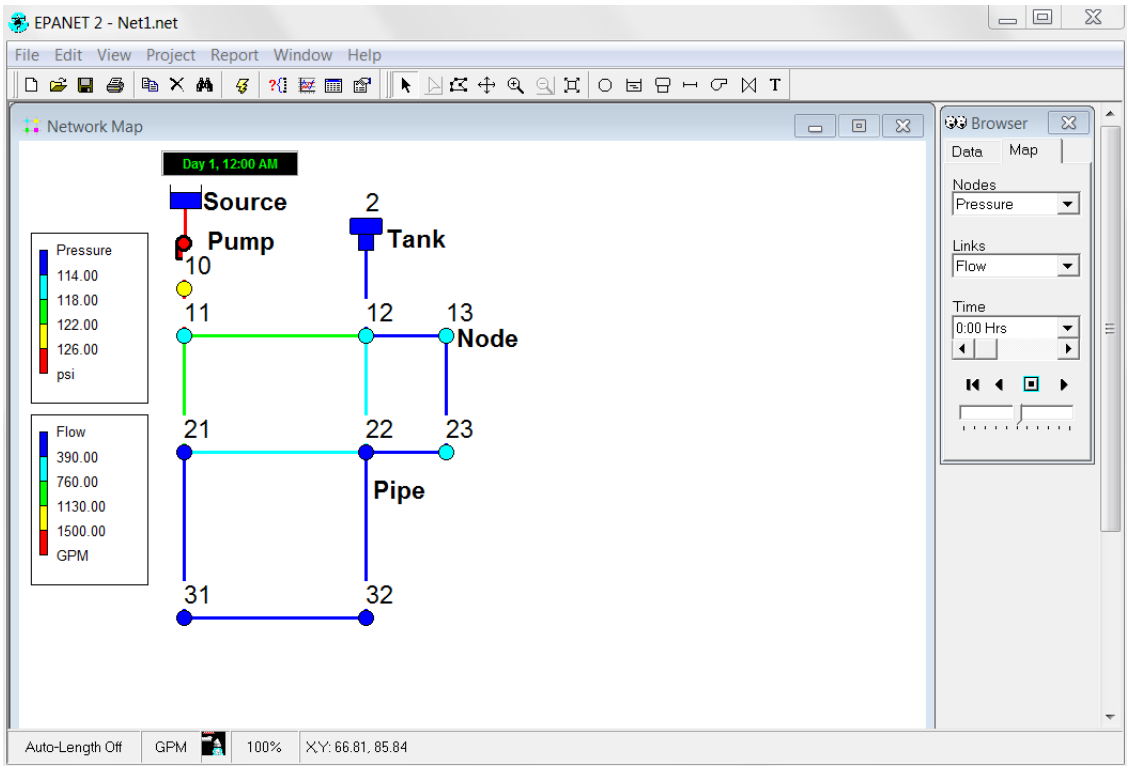


Figure 2.2: EPANET hydraulic modeling tool user interface.

to the end-users is a function of the pressure at that node:

$$d = \begin{cases} 0 & p \leq P_0 \\ D_f \sqrt{\frac{p-P_0}{P_f-P_0}} & P_0 \leq p \leq P_f \\ D_f & p \geq P_f \end{cases} \tag{2.1}$$

where d is the actual volume delivered to the end-users (m^3/s), D_f is the expected demand (m^3/s), p is the gauge pressure inside the pipe (Pa), P_f is the pressure above which the customer receives the expected demand (Pa), and P_0 is the pressure below which they cannot receive any water (Pa). While WNTR includes some features of EPANET to model and analyze water distribution system resilience, it extends these capabilities by providing a flexible platform for modeling a wide range of disruptive incidents, repair strategies, and hydraulic behaviors during low pressure conditions given the PDD model.

2.2 State-of-the-Art in Deployment and Maintenance of Community Water Systems

We start with the design and enhancement of water CPHI systems in terms of network structure analysis and optimal cyber instrumentation. We then discuss a wide range of methodological approaches for detection and localization of failures. Research efforts in these works motivated and inspired our approaches on fault identification under diverse conditions, which is our focus of the thesis. Lastly, we briefly overview the study on control and recovery of critical infrastructures to mitigate and limit cascading negative impacts, which is important to minimize public health and safety consequences.

2.2.1 Planning: Design and Enhancement of CPHI Systems

The growing demand for clean water has pushed water infrastructures to grow into large-scale networks which carry fluid under high pressure. Consequently, water networks have evolved into complex systems with thousands of branched and interconnected lines [144]. This has attracted increasing interests on the design and enhancement of water CPHI systems to reduce the likelihood and impact of asset failures.

Physcial Network Structure Analysis

Although many efforts are underway to improve the resilience of water infrastructures, many of the world's cities are currently served by old and poorly managed networks with consequential problems in terms of both water quality and quantity reliability [135]. The prime concerns for practitioners responsible for the design and operation of the water distribution networks are to improve system reliability and reduce system susceptibility to damage and perturbation. [145] proposes to enhance service resilience through the incorporation of strategic network redundancy. They study the structure of water networks using graph

theory as the underlying analytical tool, and introduce a simplified network-based approach that allows for a comparison of connectivity patterns for alternative design. In their case studies, the proposed approach is capable to provide useful insight on where and to what extent the redundancy to avoid or tolerate bottleneck should be allocated in the network structure. The authors of [127] focus on how network properties can influence performance in both hydraulic (e.g., minimum pressure, maximum hourly unit headloss) and water quality contexts. By leveraging strengths from graph theory and statistical inference, they develop a methodology for characterizing the relationship between the topological structures and the system-level performances. They also study the tradeoffs between two network structures: loops for reliability and branches for cost savings. The approach is evaluated on $10,001$ simplified and functional lattice-like pipe networks. This cost-effective proxy at a structural level can serve to complement traditional, complicated physics-based domain model of water networks. In [114], the authors study cascade-based attacks and the nodal vulnerability of water networks under cascading failures. They evaluate the network vulnerability with the consideration of two aspects: (1) topological analysis from complex networks; and (2) hydraulic analysis from water systems. Because, at each time, a water network is required to meet the equilibrium of water supply and demand. The proposed method is validated by comparing with methods from betweenness load, flow entropy model and minimum cut-sets on the same data set.

Cyber Instrumentation

Instrumenting infrastructures with sensors for monitoring is, clearly, the starting point to enable the quick detection and identification of disruptions. Over the last decade, water utility companies have aimed to make water distribution networks more intelligent to improve quality of service and reduce water loss and maintenance costs. In order to achieve these goals, water networks have been augmented with sensor nodes and data loggers for monitoring by transmitting the network state periodically. In parallel, civil engineers advocate that next generation of water distribution networks will not be passive water delivers, but active

highly-distributed event-based control systems [7]. Such a dynamical system will heavily rely on sensing and actuation.

However, devices and networks are expensive to install/deploy and maintain, which is especially pronounced for underground infrastructures due to inaccessibility. The inherent scale of todays water networks means that instrumenting the entire network would be prohibitively expensive, thus calling for selective placement solutions. In practice, SCADA systems, when available, are deployed at the edges of the network (e.g., pump stations above ground). With the advent of such sophisticated monitoring systems, it is possible to monitor pressures, flows and water quality at certain key points within the network in real time. Nonetheless, the network of hundreds or thousands of pipelines is not actively monitored, making it prohibitive to identify failures at pipeline level.

There have been several efforts towards instrumenting water networks with in-situ (static) sensors to detect and localize events in a timely manner. PIPENET [119] focuses on the use of wireless sensor networks (WSNs) for monitoring water transmission pipelines by collecting hydraulic and acoustic/vibration data at high sampling rates. WATERWISE [141] develops generic WSN capabilities to enable real-time monitoring of a water distribution network in Singapore. In [64], the team designs and develops a small-scale testbed, WaterBox, to allow researchers to test control algorithms in a fail-safe environment. This is necessary because software simulators are insufficient to reproduce issues that arise in real systems [108], and contemporary large-scale testbeds are limited due to safety concerns. Although these high-end static sensors, designed as low-cost devices, can provide continuous measurements, wide sensing ranges and good accuracy, the instrumentation of civic water infrastructures at large requires significant investments (millions of dollars). Systems like SmartBall [47] and PipeProbe [72] drop mobile sensors into the network which traverse and monitor pipelines by moving with the water flow. Mobile sensors allow for adaptive sensing on-demand at significantly reduced cost when compared with in-situ sensors. Because they can be deployed

at different locations, at different times, and with different sensing capabilities based on the need. However, they only provide a single measurement in time and have low sensing ranges.

These emerging sensing technologies can help enable the real-time monitoring and measurement platform for water infrastructures. Identifying optimal locations to instrument/deploy in-situ and/or mobile sensors remains a challenge. In practical setting, the most widely studied sensor placement formulation for water networks is to minimize the expected impact on communities of an ensemble of anomalous events given a sensor budget [21]. In [23], the authors present a mixed-integer programming (MIP) formulation that incorporates information about the temporal characteristics of an event occurring in water distribution networks, as obtained from standard domain models. The objective is to minimize the expected impact of a set of incidents, subject to the given resource constraints. They propose to use GRASP heuristic to solve this MIP model, which is robust and scalable for large-scale real-world settings. This work can be viewed as optimizing one particular statistic (i.e., minimization of exposure of populations), [137] suggests that other statistics may provide more "robust" solutions under different conditions. For example, they argue that the design objective of sensor placement should include, other than expected impact minimization, in particular the minimization of the worst-cast impact. Anomalous events are not necessarily uniformly distributed, therefore the robust sensor placement is required to address the issue of how to mitigate against high-consequence events (e.g., 9/11-style attacks). While [136] proposes that some performance objectives cannot be simultaneously optimized, and therefore the sensor placement problem must carefully consider the trade-off between multiple, disparate objectives. Their approach can support the analysis of these trade-offs, by characterizing the interdependencies among a range of design requirements (e.g., population exposed, time to detection, number of failed detections). The aforementioned methods make the implicit assumption that sensors work perfectly. In practice, sensors can fail to detect an anomaly. [22] addresses this issue by supporting a formulation that models sensor failures, by assigning each sensor an associated probability of failure.

In recent years, researchers have worked on combining mobile and in-situ sensor deployments for resilient water infrastructures [98, 122, 105, 95, 39]. The problem is challenging due to the large size of the networks and the probabilistic movement of the mobile sensors. In [39], a novel optimization approach is posed to provide a better coverage of the water networks using both mobile and in-situ sensors. They partition the objective function into a set of optimization problems, and show that these problems exhibit submodular properties. Therefore, a greedy mobile nodes release algorithm is proposed. However, this work, in fact most of existing work, typically assume the priori placement of static infrastructure - either in-situ sensors or sink nodes/beacons that support the communication with mobile sensors. [132] argues that it is essential to consider the deployment of both types of sensors simultaneously, because the placement of one type can affect the decision of another. Furthermore, [132] distinguishes events based on their impact on the community, instead of assuming a uniform distribution. Build on the previous work [131], they present a cost-effective hybrid monitoring architecture to minimize the impact of adverse water events on the community, combining the benefits of mobile and in-situ sensing with various geo-social factors.

2.2.2 Operation: Detection and Isolation of Failures

CPHI systems are becoming common in multiple domains (smart grids, intelligent transportation, natural gas); however, gathering and processing data for operational purposes in such settings have cross-layer concerns at multiple levels of the system (devices, networks, data, compute etc.). This section focuses on data-driven methods to isolate faults as applied to large scale distributed CPHI systems, e.g., water infrastructures; key topics include modeling the underlying systems and their operations as well as enabling efficient data gathering, data analysis and event processing to support decision making. Integrating human agents in the sensing, analysis and actuation tasks is a viable path to deployment.

Current Practices: We start with the introduction of current practices that have been applied by water utilities for fault identification in water networks. In reality, fault identification defines a complex combinatorial optimization problem where finding a solution often requires extensive computations depending on the network size and capacity, and supply-demand model driven by terrain specifications and population density of the served area, to say the least. One current practice used by water utilities for leak detection and isolation is acoustic inspection. It uses acoustic instruments to listen for variations in reflected signals caused by the leak. Although this "hear-and-repair" method has also been widely applied in other fluid networks e.g., nuclear power plants [58], the effectiveness is mainly valid within an area immediately adjacent to the failure and performing such measurements can be very expensive [50]. Another practice that has been adopted by utilities is to use a calibrated hydraulic simulator to enumerate possible leaky points for a best match between the simulation results and the meter data [92]. This appears plausible and is also proposed in other work e.g., [99, 109, 100]. However, it can be computationally expensive or even prohibitive for leak localization in community-scale water networks. Because the location and severity of a leak jointly affects the hydraulic behavior, and, if any, multiple leaks interact with each other, making it difficult to enumerate a one-to-one match.

Techniques for data and knowledge representations using graph-based techniques, ontologies and semantic web approaches have been used to model community data. Workflow processing systems (e.g., Ptolemy) and simulation techniques have been utilized to capture operational aspects and reason about the state of systems. In the realm of Civil Engineering, there have been numerous mechanisms used to detect pipe failures using intrusive sensing techniques, such as in-pipe sensors or even mobile sensors that roam within the pipeline network [65]. However, the limitations are also well-known in terms of the high cost of sensing devices and deployment, and the risk of water contamination. Methods studied in [75, 33, 76, 121] are based on fluid transient signal processing, since a sudden break often causes a pressure change followed by a transient wave traveling along the pipe. These approaches are evaluated

on the identification of a single pipe failure. Its scalability to the water network is uncertain due to the difficulty of obtaining an accurate transient model for a pipeline network.

Alternatively, instead of solving it as a formal optimization problem, the use of stochastic (rather than deterministic) formulations to identify pipe leakage have been around for some time, but their uptake has not been prolific [65]. [100] integrates domain simulator with SCEM-UA (shuffled complex evolution metropolis) algorithm to estimate the posterior probability distribution over potential leak locations. The measurements are assumed to be available at all nodes in the network, which is however not realistic due to the inaccessibility and costly sensor installation and maintenance. Similar techniques are also proposed in [99, 109], where they apply the Bayesian network for diagnosing water leakage while considering limited number of sensors and measurement errors. However, both methods lead to the issue of computational complexity on community-scale water networks. Other approaches using current-flow centrality analysis [91, 13] and state estimation [49] have also been investigated on water networks with small size and simple topology, yet the performance in real-world settings is not evaluated. Over the last decade, there are increasing number of machine learning (ML) based techniques [14, 84, 58, 61, 88, 133] proposed for failure localization in water networks. Unfortunately, only a paucity of methods have explored the physical mechanism inherent in the network topology to support their analysis. Figure 2.3 is the summary of limitations and advantages of these existing techniques for fault localization in water networks, and we aim to design and implement an approach that can quickly and accurately isolate multiple failures given that only a limited number of sensing devices are available in such underground infrastructures.

There is a research group at Imperial College London, who has presented several work on localization using graph theory techniques [146, 147, 138]. In [65], they extend their prior work and introduce an end-to-end leak localization platform, which exploits the edge computing and enables the use of low-resourced sensor nodes. The proposed platform is

	Approach	Fault Localization	Deployment Penetration	Detection Speed	Multi-fault Localization
Practical	**Hear & Repair**	Yes	---	Slow	Yes
Practical	**Hydraulic Model**	Yes	Medium	Slow	No
Theoretical	**WaterBox** [S.Kartakis et al. 2015]	No	---	---	---
Theoretical	**Fluid Transient** [J. Sun et al. 2016]	Yes	---	Fast	No
Theoretical	**Current-flow Centrality** [I. Narayanan et al. 2014]	Yes	Low	Fast	No
Theoretical	**Min. Test Cover** [W. Abbas et al. 2015]	Yes	Low	Fast	No
Theoretical	**State Estimation** [F. Fusco et al. 2012]	Yes	High	Fast	Yes
	Our Approach	**Yes**	**Limited**	**Fast**	**Yes**

Figure 2.3: Comparisons of fault localization techniques applied in water networks.

a highly distributed lightweight scheme that combines data compression techniques and graph-theory based localization algorithms on the edge to identify leak events. This edge anomaly localization approach can produce a timely and accurate results and reduce the communication by 99% compared to the traditional periodic communication. However, these works overlook the underlying causal dynamics of cascading events/failures in the physical systems, which captures the spatial-temporal dependency between consecutive failures.

Recently, approaches utilizing HMM [103], that captures the underlying physical dynamics, have been successfully applied in multiple domains, e.g., social media [148], transportation [101] and power grids [59]. Particularly, [59] has shown that HMM outperforms the conventional algorithms in identifying the cause of transmission line outages. HMM is a probabilistic state transition model, which is commonly used to model systems with the events we are interested in are hidden: we do not observe them directly [103]. It can capture both observed events and hidden events that are thought of as causal factors in the model,

and thus it is useful when we need to discover the best hidden state sequence given a time series observations. A HMM contains a hidden Markov and an observed processes. In hidden process, a set of hidden states follow a Markov process with the property that the states before the current state have no impact on the future except via the current state. Given this, the observed process is a sequence of conditionally independent variables and depends on the hidden process only through the corresponding hidden state.

Integrating human agents in the sensing, analysis and actuation tasks is a viable path to deployment. Human-interactive systems have leveraged human capabilities to complex systems to make intelligent decisions for a while. For example, [150] uses crowd-sourced information for timely data delivery and [71] models and analyzes customer behavior for efficient energy management. ML approaches today are becoming available as generalized services and the role of ML such as RL for rapid analysis of complex events is invaluable. The basic idea of RL is to have a learning agent interacting over time with its environment to achieve a goal by capturing the most important aspects of the real problem [124] - this has emerged as a powerful tool for solving sequential decision-making problems. It is well known for its great success in online games - AlphaGo Zero [115], and recently has been successfully applied in natural language processing [46], software development [143], recommendation systems [125] and ridesharing order dispatching [77].

2.2.3 Adaptation: Control and Recovery from Failures

Should the presence of failures is detected, consequence management must be employed. Decision-making tools that assist in evaluating and planning various response strategies are needed to support fast recovery of infrastructures to return to normal operating conditions.

[44] introduces a common operational approach that water utilities use to mitigate water quality concerns. That is the purging of water from the distribution network using a fire

hydrant or blow-off port, namely flushing. Flushing can remove the sources of poor water quality, and it can be made more efficient through the strategic selection of where to implement flushing activities. This problem is formulated as selecting a set of locations to flush that minimizes the human exposure to contaminants, and solved using an iterative optimization process that requires numerous simulations of the network hydraulics and water quality. A major limitation is that computational run time grows exponentially as the size of the network, the number of possible incidents and the number of feasible intervention locations. Several techniques are therefore studied to decrease the computational complexity, including parallelization, stop time criteria and skeletonization.

Most of the literature focus on recovering the individual infrastructure as fast as possible with the consideration of limited available resources. Nonetheless, only a paucity of work that have been done on recovery planning include the interdependencies between the infrastructure networks. Such consideration is imperative, given that the interdependence between urban infrastructures increases the vulnerability of individual network, while imposing additional constraints to the restoration of each of them [40]. In [53], the authors aim to optimize the recovery of interdependent networks by considering their functional coupling. Possible simultaneous recovery jobs in corridors of colocated components are identified such that the cost of reconstruction is reduced. They also study a simulation-optimization framework to take into account uncertainty in the physical system by nature, enabling the analysis of the expected behavior of a system.

Chapter 3

An Integrated Approach to Resilient CPHI Systems

We now provide a high-level overview of our proposed approach **AquaSCALE** - an integration framework to enable resilience in cyber-physical-human water infrastructures. We first motivate and study the water CPHI system resilience using two community-scale water networks in the east coast and west coast. Then we present our approach, its system architecture and design decision, that can model and explore the improved performance of water infrastructures in diverse communities with different structures, scales and concerns. We further explore our integrated approach to address resilience concerns at multiple levels of a CPHI system.

3.1 AquaSCALE: Executable Computational Framework to Explore Resilience of a CPHI System

Water is essential for sustaining the economic and social viability of a community; water infrastructures are considered an important lifeline to communities worldwide [5]. Recent events such as water supply disruptions caused by Hurricane Sandy in 2012 and multiple pipeline ruptures caused by Ridgecrest earthquake in 2019, and the looming California drought crisis clearly indicate society's dependence on critical lifeline services and the far-reaching impacts that disruption can cause. In the current setting, the generation, distribution and maintenance of lifeline services (e.g., water) is managed in a distributed way by local agencies, who have encountered different concerns. We engage with two cities Montgomery County (MC), Maryland and the City of Los Angeles (LA) as our focus testbeds, because they represent diverse communities with different structures, scales and vulnerabilities. In MC, extreme cold weather and heavy rainfall (e.g., El Niño 2016 and La Niña 2017) can stress already weakened pipes to the point of causing major pipe breaks or excessive leaks [10]; while 88% of the water supply to LA cross the San Andreas Fault such that a big earthquake on the San Andreas could cut off the water services for more than 22 million people [3]. What is lacking - and what defines the future - is an underlying common platform allowing to model and explore resilience of water CPHI systems in real-world communities with different concerns [28].

Designing robust water CPHI systems involves a clear understanding of the structure, components and operations of this system, how community infrastructure dynamics (e.g., varying customer demands, extreme weather and small/large disruptions) impact lifeline service availabilities, and how service level decisions impact infrastructure control. The ability to view water service flows as a community wide CPHI system with multiple levels of observation/control and diverse players presents new possibilities. Built on our experiences

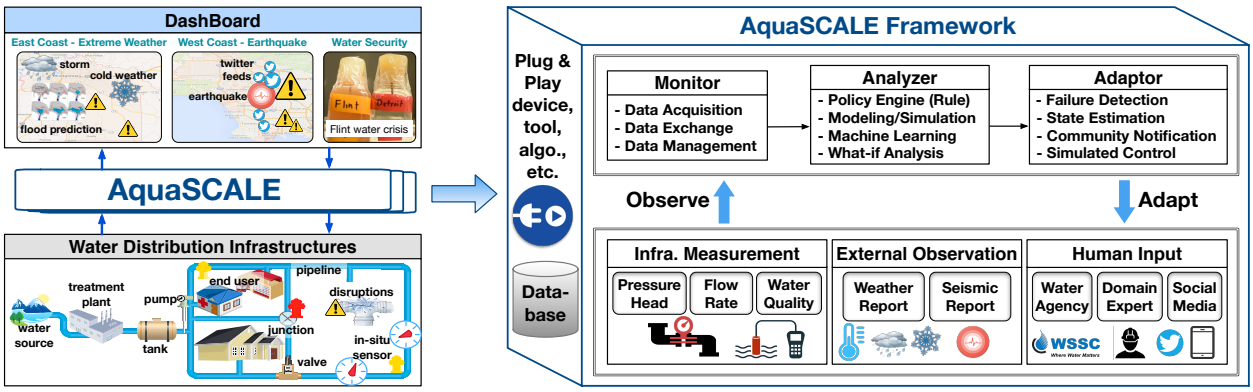


Figure 3.1: AquaSCALE system architecture. It bridges the infrastructure/application gap by transforming raw sensor data into higher level semantic streams through a logic observe-analyze-adapt loop.

on SCALE project [19, 129, 20, 149], we design and develop AquaSCALE, an executable computational framework and tools, to model, simulate and explore community water infrastructures at two layers - (a) a higher layer to capture the availability of features and services to end users and (b) a lower built infrastructure layer to model components, interconnections and flows. The framework represents spatial and temporal aspects of water lifelines operating under normal and abnormal conditions, and estimate their vulnerability to varying levels of shock intensity. It is designed to prevent water service failures by identifying operational degradation in aging infrastructures (e.g., pipe leaks), improve speed and accuracy of damage estimation in larger hazards (e.g., earthquakes, floods and ice storms), and improve service restoration times in the presence of disruptions.

3.2 System Architecture

AquaSCALE (Fig. 3.1) is designed as a sensor-simulation-data integration platform to integrate multiple information sources and technologies. An integrated view of a CPHI system is required to enable resilient and scalable urban infrastructures due to the following observations. Measurements from underlying infrastructures (e.g., flow rate, pressure head) reflects

the physical behavior of a network that is dynamically altered by supply-demand variations, operations designed to conserve resources, as well as cyber-physical disruptions. The installation/deployment and maintenance of sensing devices is expensive and time consuming, and many cities do net yet have the much needed data (e.g., demand history, population distribution and risk analysis) to compute an optimal sensor placement; sensor data has inaccuracies due to the anomalous variability in devices and communications. This is especially pronounced for underground infrastructures (e.g., water networks). Furthermore, techniques to process the data have uncertainties, creating computational challenges on abstracting correct information from limited measurements in a timely manner. External observations can provide additional information for decision-making. For example, sub-freezing temperatures add stress to water pipes; this can - and often does - result in breaks [4]. In the context of earthquakes, the knowledge of seismic waves can help estimate the possible extent of impact on e.g., electricity and water delivery. Humans and human inputs play a vital role in decision-making in settings where sensing needs are expensive and uncertain. For example, the damage to buried infrastructures is often hidden and most pipe leaks are silent until noticed by people.

As shown in Fig. 3.1, the core of AquaSCALE framework is a data-driven simulation engine that executes a logical observe-analyze-adapt loop. The input to the analyzer is derived from *Observations* gathered from diverse data sources and stored in the data management module. The *Analytics* module that subsumes models and techniques developed by domain experts will operate on the near real-time data to generate higher level awareness for specific application tasks. The awareness will trigger corresponding logical *Adaptations* within the framework that will again generate a new set of observations.

To realize this observe-analyze-adapt loop, AquaSCALE is designed as a workflow based system comprised of multiple modules (Fig. 3.2).

• The **Scenario Generation module** enables water managers and analysts to provide

meaningful and diverse water contexts to the framework by generating a range of situations. A user of the tool can start defining a situation by choosing a geographic region, entity elements of interest in that region, and using additional modules to identify hazard, vulnerability, restoration and impact of the hazard at a temporal and spatial scales of choice. In this thesis, we particularly address three resilience problems: isolation of multiple faults, state estimation under extreme events, and fault source identification. We will explain each individual work in great detail in the following chapters.

• The **Sensor Data Acquisition module** enables gathering of real-time field information for predefined scenarios by projecting the effects of new updates from the field on simulation outcomes. This module is not merely a collection of sensing data throughout a CPHI system. Fundamentally, it is information acquisition and management. Beyond each individual data source, coordination among different data streams is needed to establish a comprehensive view of the current system status and provide specific, reliable and timely actions in response to the incident.

• Understanding how a CPHI system will perform during a range of threats and being able to measure how well an approach would adapt to such threats require an **Integrated Simulation and Modeling Engine**. Simulation tools can help explore the capacity of a system to handle disruptive events and guide the planning necessary to make systems more resilient over time [69]. To explore the resilience of water CPHI systems, we integrate EPANET hydraulic and water quality model, WNTR water network tool, NOAA's hydrometeorological data, TAS tweet acquisition system and BreZo flood model.

• A **Plug and Play Analytics module** is used to plug and unplug specific information, such as data sets and algorithms, at will depending on the specific context of applications, and to understand the advantages and limitations of diverse strategies in isolation and combination.

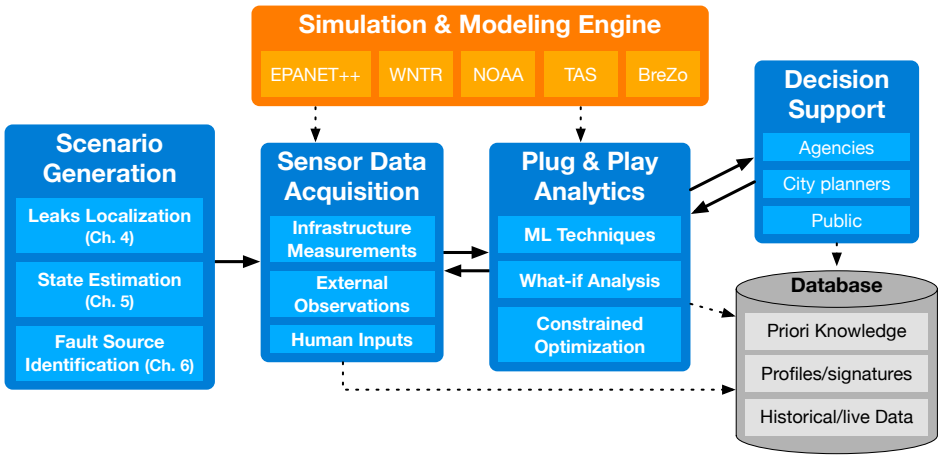


Figure 3.2: AquaSCALE prototype implementation. It is designed as a workflow based system comprised of multiple modules.

• Users/operators/analysts interact with AquaSCALE using the **Decision Support module** to manage devices at runtime as they identify vulnerable spots and address cost-accuracy-time tradeoffs, and to optimize sensor placement for a better performance.

3.3 AquaSCALE for Cross-layer Concerns

Gathering and processing data for operational purposes in CPHI systems have cross-layer concerns. As shown in Fig. 3.3, AquaSCALE enables a higher level knowledge abstraction from raw sensor data collected at lower infrastructure layer, and resilience techniques to deal with limitations/disruptions in a CPHI system at multiple levels.

Community-scale sensor placement is important for operational management during normal conditions and fault detection/localization during disruptive events. Current sensor placement techniques use heuristics that focus mainly on enabling network coverage while treating all events uniformly. We argue that identifying the needs of the community and the impact of the event is also important, because not all events are equally impactful. Our previous work [131] developed a structured geo-social approach for sensor placement that is

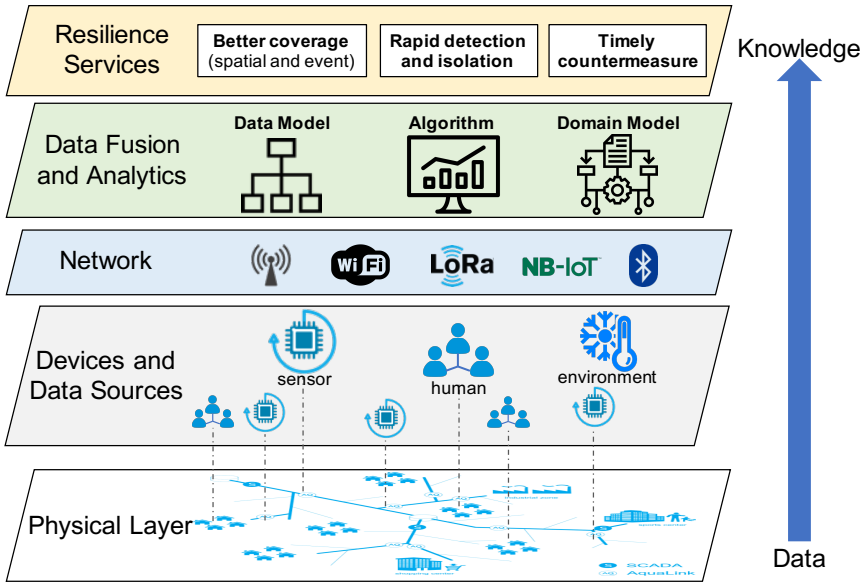


Figure 3.3: AquaSCALE addresses cross-layer concerns at multiple levels of the CPHI system including devices, networks and data analytics.

contextualized to community-scale water distribution networks. We partition the terrain of the community area into regions using Delaunay triangulation, and determine the criticality of these regions based on the population density as well as the critical infrastructures (e.g., hospital, school, airport) present within the region. Our approach models and quantifies the real-world impact of a failure on a community using various geospatial, infrastructural and societal factors. We then integrate failure impact, IoT sensing data and simulation based analytics to drive sensor placement methods with the objective of reducing community-scale impact. Build on this work, [132] introduced a general adaptive monitoring framework leveraging the strengths of both in-situ and mobile sensing, and provided cost-effective planning and deployment solutions to mitigate the impact of adverse events. The proposed 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 first integrates network topology and community information to determine the placement of in-situ sensors and mobile sensor insertion infrastructure, and then incorporate network flow dynamics to determine mobile sensor deployment locations with the goal of reducing costs

and minimizing adverse impact while maintaining localization accuracy.

Reliable delivery of sensor data to distributed cloud service instances is crucial consideration to create dependable awareness and consequently improved decision making. However, this is challenging in the face of massive geo-correlated communication network outages caused by large events (e.g., earthquakes). In [20], we described a centrally-controlled Geographically Correlated Resilient Overlay Network (GeoCRON) to ensure more reliable collection of data at a cloud service despite geo-correlated infrastructure failures throughout the sensed region. GeoCRON achieves reliable message delivery by exploiting geographically-redundant network routes to avoid failures. Particularly, it gathers the underlying routing infrastructure topology and locations of both IoT nodes in the overlay and routers in the underlay. It uses this information to establish multiple geo-diverse routes in the overlay according to the number requested by an applications resilience requirements. Sending multiple copies of each message along these geo-diverse routes improves the chances of successful message delivery (to the cloud data exchange) during large-scale geo-correlated failures.

Data-driven analytics for community resilience is key to understanding the behavior of CPHI systems before, during and after disruptions, and reducing their vulnerability to and enhancing their recovery after a disruption. An efficient approach should effectively translate (possibly incomplete) data to meaningful information for timely decision-making by optimizing tradeoffs between cost, accuracy and time. To better fuse and interpret a multiplicity of sources, we apply our integrated approach to solving three major problems in regard to fault identification in a CPHI system, which is the focus of this thesis. We explore the resilience under small failures (§4) and extreme events (§5), as well as human-enhanced analytics for resilience (§6). Through these work, we show that AquaSCALE can mitigate the significant costs associated with various types, timescales and intensities disruptions to water CPHI systems.

Chapter 4

Infrastructure Resilience under Small Failures

In this chapter, we begin our exploration of resilient water infrastructures under multiple concurrent small failures. We apply our proposed AquaSCALE framework for gathering, analyzing and localizing anomalous operations of increasingly failure-prone community water services. Today, detection of pipe breaks/leaks in water networks takes hours to days. AquaSCALE leverages dynamic data from multiple information sources including IoT sensing data, geophysical data, human input and simulation/modeling engines to create a sensor-simulation-data integration platform. It can accurately and quickly identify areas of potential water leakage. We propose a two-phase workflow that begins with robust simulation methods using the commercial-grade hydraulic simulator EPANET, which is then enhanced with the support for IoT sensors and fault events modelings. The platform generates a profile of anomalous events using diverse plug-and-play ML techniques, which then incorporates with external observations (NOAA weather reports and twitter feeds) to rapidly and reliably identify the locations of damaged water pipes. We evaluate the efficacy of proposed methodology using canonical and real-world water networks under different scenarios. Our results indicate

that the proposed approach with offline learning and online inference can improve the process of locating multiple simultaneous pipe failures at a useful level of granularity (pipeline segments) with shorter detection times (from hours/days to minutes).

4.1 Chapter Overview

Pipe leaks are ubiquitous in aging community water infrastructures [126, 78]. Recent repair reports from the Los Angeles Department of Water & Power (LADWP) and the Washington Suburban Sanitary Commission (WSSC) indicate that communities are experiencing an unusual increase in pipe breaks, mainly in old pipes that are susceptible to corrosion problems or pipe joints that are experiencing displacements caused by surface deformations. Like many other urban infrastructures, water distribution networks consist of multiple interconnected components, whose individual or simultaneous pipe failures can cause disruption to water services. Although many problems due to old and poorly maintained infrastructure are individually small, they can quickly add up. For example, numerous small leaks have caused some municipal water systems to lose up to 20% of their water during transmission [60]. Approximate 14 - 18% of treated water in the U.S. is lost because of leaks or breaks in faulty pipelines [91, 66]. The quality of water supplies can also be compromised via contaminant propagation through a leaking pipe. A large-pipe break can cause severe flooding and impact other lifelines, e.g., the water main break on Sunset Boulevard flooded the UCLA campus on July 2014. Extreme weather and heavy rainfall (e.g., Hurricane Sandy 2012, El Niño 2016, and La Niña 2017) can also stress already weakened pipes to the point of causing major pipe breaks or excessive leaks. Additionally, large-scale disasters, such as earthquakes, can cause pipe failures that can drain vital water supplies when most needed, e.g., extinguishing ensuing fires.

All of the above highlight the need for quick and accurate identification of pipe failures, such that timely countermeasures can be performed to preserve precious water supplies and prevent negative impacts on other interdependent lifeline services. The operations of water infrastructures are subject to complex, highly non-linear temporal and spatial processes, making it challenge to differentiate between faults and stochastic network behaviors. This makes multiple-leak identification a non-trivial task, leading us towards a solution based on

integrating information from multiple sources. IoT sensing data from underlying infrastructures can track variations in the network and inform pipe breaks based on the reduction in pressure/hydraulic heads and increase in flow rates at failure points [99]. Pressure head is essential piece of information that is required for determining water service availability, which is defined as the mechanical energy per unit weight of fluid. External conditions can be used as the additional information for failure detection. According to our experience with WSSC (a water agency at east coast of the U.S.), extremely cold temperature is likely to cause pipe breaks due to ice blockage - this knowledge can be used to capture patterns of changes in pressure heads (increase first due to pipe freeze and decrease due to pipe break). Human leak-related report, in some cases, is the only indicator of the presence of failures due to the sparse sensor deployment on such underground infrastructures where failures can be silent until noticed by people. The aggregation of these external observations can help improve our assessment of pipe failures.

Given that most of the existing work focuses on identifying single pipe failure on small-scale water networks (Fig. 2.3), our study addresses a more realistic case where the available measurement is limited by the type and number of sensors and the objective is to localize multiple concurrent leaks of a real-world water network in seconds/minutes (instead of hours/days). To capture the dynamics of complex water networks, we apply AquaSCALE to enable the fusion of multiple different data sources, robust simulation engines and plug-and-play ML techniques. We show that AquaSCALE, as a CPHI-enabled service, can model community water distribution infrastructures and pipe failures, and support the integration of various information for identifying potential failures.

Key Contributions of This Chapter:

• Leveraging AquaSCALE computational framework to integrate multiple data sources and techniques for localizing leaks in community water networks - (Sec. 4.2).

• A novel two-phase process for leak identification using an offline profile generation for quickly identifying potential faulty pipes and online live data integration for accurately localizing damaged pipelines - (Sec. 4.3/4.4).

• A plug-and-play analytic engine that enables selection/integration of statistical ML techniques for fault identification and transforms low level pipeline information into higher level impact (e.g., floods) - (Sec. 4.4).

• Extensive evaluations of the proposed approach under diverse failure scenarios using real-world water networks - (Sec. 4.5).

4.2 Approach Overview

To quickly identify leak events in real-world water networks, we argue that an integrated approach to fusing multiple (possibly incomplete) sources of information is necessary. The approach allows to explore solutions to problems in cyberspace before instantiating them into a physical infrastructure.

In the paper, we leverage AquaSCALE for pipe leak identification. AquaSCALE supports a novel two-phase approach for managing water workflows at multiple levels of observation and control. In the first phase, statistical approaches are used to drive the offline creation of a profile model of faults and their impact to help rapid identification of the problem in near real-time. While this initial phase significantly reduces the online detection time, the second phase exploits the availability of dynamic data and compensates for the limitation of the offline model to improve accuracy and efficiency. To support a flexible suite of methods for leak events detection, we incorporate a plug-and-play analytic engine that enables the selection/integration of statistical techniques for improved identification of faults. Statistical based data integration algorithms are used to incorporate IoT measurements with additional

observations. This analytic engine facilitates the discovery of an efficient composition of techniques for failure localization in a given water network. In our prototype, robust simulations using an enhanced version of a commercial grade hydraulic simulator EPANET (with added support for IoT sensors and failure modelings) are used offline to train a profile model of anomalous events. The profile and multiple information sources are then used for online rapid coarse fault isolation and fine-grained fault localization (i.e., leak detection).

4.3 Modeling Resilience In Water Infrastructure

Pipe leaks or breaks, as one of the most frequent types of failures, represent a very high cost vulnerability and is associated with public health implications and wastage of limited resources. It is often caused by operation degradation of pipelines, extremely cold temperature, and large-scale disasters (e.g., earthquake). Leak events may be identified through diverse information sources - an unexpected reduction on pressure heads; an abnormal increase in flow rates; leak-related messages posted on social media platforms. Thus we introduce multiple information sources into AquaSCALE, and evaluate its efficacy in the treatment of pipe leaks.

This section introduces the modeling of leak event, IoT measurement in water infrastructures, weather information, and human input. Our experience indicates that IoT measurement alone may work well to identify a single leak event, but, as we explain later in Sec. 4.3.1, it is not sufficiently accurate to isolate multiple concurrent failures. The combination of diverse information sources provides new possibilities for enabling detection of multiple pipe failures. In the real world, extremely low temperatures can cause ice forming in a pipe that leads to complete ice blockage, and continued freezing and expansion inside the pipe increase water pressure heads that leads to pipe breaks. Thus ambient temperature, though is coarse-grained (city-level information), can provide an additional pattern of pressure changing for

leak localization. The damage to underground infrastructures is often hidden, and most pipe failures are silent until noticed by people. In the case where IoT measurement is unavailable, human reports on leak events provide indispensable information. Such weather temperature and human input when integrated with IoT measurement can help improve the detection outcome with a higher accuracy in a shorter amount of time.

4.3.1 Modeling Leak Events

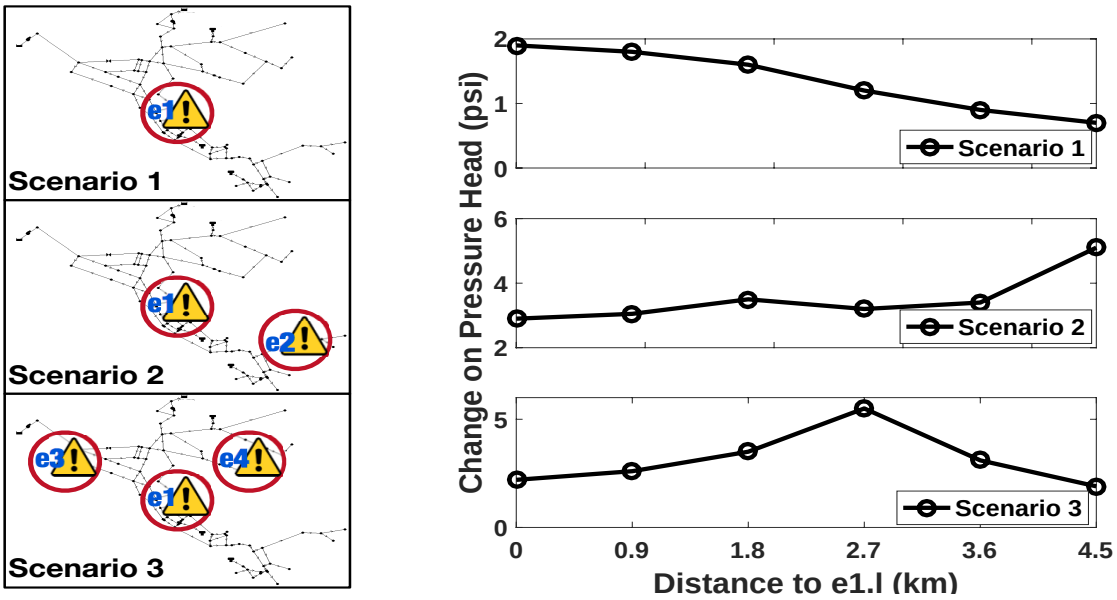
A water system is represented as an undirected graph $G(\mathcal{V}, E)$ (water can flow in both directions) with vertices $\mathcal{V}$ that represent nodes (the joint of pipes), and edges E that represent pipelines. $|\cdot|$ denotes the cardinality of a set. The leak event is denoted as $\mathbf{e} = \{e\}$, where an event $e = (l, s, t)$ is identified by location $e.l$, size $e.s$, and starting time slot $e.t$. The goal is to locate $e.l$ for $\forall e \in \mathbf{e}$. We use and enhance EPANET with the support for IoT devices and failure modelings, named EPANET++. As introduced in Section 2.1, EPANET is a software that models water distribution system and helps to improve our understanding of the movement of water within distribution systems. In EPANET++, pipe failures are simulated by emitter that is device associated with node to model the flow through a nozzle or orifice that discharges to the atmosphere [108]. Leakage continuously increases with pressure, and it is often computed using Eq.4.1 in civil engineering domain [89, 52, 51, 100, 84, 92]. More details refer to [73]. The pipe leak is modeled by

$$Q = EC \cdot p^{\beta} \tag{4.1}$$

where Q is discharge flow rate at the leak point, EC is effective leak area depending on the discharge coefficient and leak area, p is current pressure head at the leaky node and β is pressure exponent. β typically varies between 0.5 and 2.5 depending on the leak type, and we set it to 0.5 for general purpose [73]. EC indicates the leak size, (i.e., $e.s$) and the greater

the value of EC the higher the severity of a leak event. In single leak context, a node will be assigned as an emitter with a EC and a timestamp where the node is leak location ($e.l$), EC is leak size ($e.s$), and time stamp is leak starting point ($e.t$). In the multi-leak case, one or more nodes will be assigned as emitters with different EC and identical time stamp, to simulate multiple concurrent failures.

Compared with single leak identification, multiple pipe failures become much more complex to detect and locate. By executing EPANET++, our empirical results show that the changes on pressure head and flow rate are easy to be captured in single failure case (Fig. 4.1). In scenario 1 where there is a single leak, the total change in pressure values of nodes in a certain distance range of e_1 decreases with increasing distance to $e_1.l$ (Fig. 4.1b - Scenario 1), and similarly for flow rates that are not shown in the paper for saving space. Here, the distance refers to the shortest path between two nodes, and the distance between two adjacent nodes is the length of the connection pipeline. This is because a sudden pipe leak often causes a pressure decrease and a flow rate increase which is followed by a transient wave traveling along the pipe [87, 75]. This pattern can be learnt and captured to identify a leak event. However it is hard to follow a certain changing pattern when multiple failures occur simultaneously, as shown in Scenarios 2 and 3 in Fig. 4.1. Multiple leak events interact with each other and jointly affect the hydraulic behavior, resulting in a set of highly correlated observations that makes it difficult to extract correct message in a timely manner. In this case, external data sources and a hybrid of ML based techniques are used to compensate for the limitation of individual information and improve the localization performance. It is worth noting that multiple failures refer to multiple concurrent leak events where the interval between the occurrence of any two events is less than the sampling frequency of IoT devices. The problem then cannot be reduced to a single failure detection because leak events cannot be separated by time series.



(a) Scenarios (b) Changes on pressure head

Figure 4.1: Failure scenarios with corresponding changes on pressure head. (a) Three failure scenarios with a single leak event $\boldsymbol{e} = \{e_1\}$; two events $\boldsymbol{e} = \{e_1, e_2\}$; three events $\boldsymbol{e} = \{e_1, e_3, e_4\}$. (b) The sum of changes on pressure heads of nodes within a certain range of the location of e_1 along with increasing distance to $e_1.l$ for each scenario.

4.3.2 Modeling IoT Measurements

The variation in pressure heads and flow rates due to pipe leaks can be used to obtain critical information on which parts of the system are suffering from the effects of water leaks. To model IoT measurements, a set of pressure and/or flow rate sensors $\mathcal{A}$ are simulated using EPANET$_{++}$, where $\mathcal{A} \subseteq \mathcal{V} \cup E$ since pressure head is measured on node while flow rate is measured on pipeline. The hydraulic time step, time interval between re-computation of system hydraulics, is used to simulate the sampling frequency of IoT devices. The IoT observations are filtered out based on the pre-defined sensor set $\mathcal{A}$ from the computed results of all nodes and links during the simulation time period.

We consider X as a set of IoT measurements collected from sensors, and Y as a set of event variables, i.e., the states of each node (leak or not) that we wish to identify. An arbitrary assignment to X is denoted by a vector $\mathbf{x} = \{x_a : a \in \mathcal{A}\}$. Similarly for Y, an assignment

$\mathbf{y} = \{y_v : v \in \mathcal{V}\}$ is a vector of labels taking from the label set $\mathcal{L} = \{0, 1\}$ where $y_v = 1$ indicates a leakage at node v. Note that the leak event is assumed to occur at node (the joint of pipes), since the interconnect points are more risky than others [5]. In our implementation, leak locations are arbitrarily assigned meaning that the structure of labels is independent, therefore the conditional probability $p(y_v|\mathbf{x})$ can be modeled and trained by using supervised ML based techniques [123]. This is a multi-output classification problem since the dimension of the output is more than one. Due to the mutual independence of labels, the problem is then transformed into multiple binary classifications where a binary classifier is trained for each node independently [128]. The goal is to maximize the number of correctly classified labels by learning a set of classifiers that maps $\mathbf{x} \rightarrow \mathbf{y}$, which is

$$\hat{y}_v = \arg\max_{y_v \in \mathbf{y}} f_{v \in \mathcal{V}}(\mathbf{x}, y_v) \tag{4.2}$$

Each f_v is a compatibility function indicating how well y_v fits the input $\mathbf{x}$, and $\hat{y}_v$ is the prediction for $\mathbf{x}$ that maximizes the compatibility. Given the knowledge of a water network, ML based techniques can be used to find a solution to (4.2), however the prediction capability may be limited by the uncertainty of IoT measurements due to noise and interference, incomplete observations due to inaccessible locations and high cost, and highly overlapped observations due to tightly interconnected network structures.

4.3.3 Modeling Weather Information

When the ambient temperature falls to 20 degrees F or below, pipes may be subject to freezing in the event of extremely cold weather [1]. According to the general manager's report from WSSC and weather report from National Oceanic and Atmospheric Administration (NOAA), pipes become more brittle during the winter and the chance of water main breaks rises significantly as the temperature drops (Fig. 4.2). Cold weather can be a root cause of

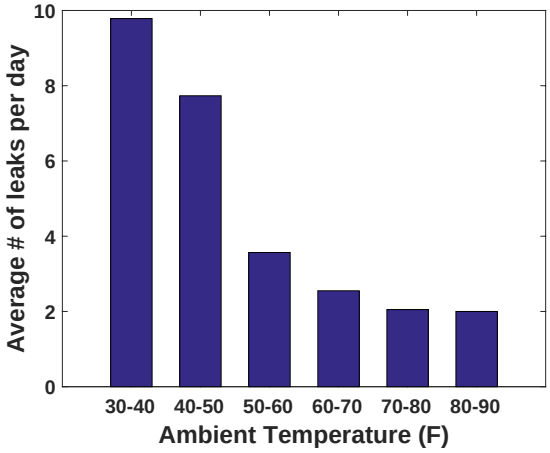


Figure 4.2: Average number of pipe breaks per day along with ambient temperatures in the regions of Prince George's and Montgomery County's for recent five years (2012-2016).

pipe breaks. The measurement based study in [1] addresses that frozen pipe itself does not typically cause a break. Instead continued freezing and expansion inside the pipe increase water pressure that can dramatically increase stress on a pipe and cause the pipe break following a pressure decrease. Therefore, the pattern of a pressure increase followed by a decrease can help isolate a faulty pipe. That is, if the ambient temperature is below $20°F$, a time series of pressure values will be processed that may provide additional information for a faulty pipe. As mentioned, the paper is to evaluate the effect on leak localization by integrating multiple information sources and techniques. Thus weather information is modeled straightforward using probability representation. For each node v, we define p_v(freeze) as the probability of freezing if the temperature is below $20°F$, and p_v(leak|freeze) as the conditional probability of leak due to freezing. We intend to model weather information using Markov chains as part of the future work.

4.3.4 Modeling Human Input

To leverage human input, we bring in social media, Online Social Network (OSN), for the incorporation of human sensing. OSN has become a major platform for information sharing

in which we can mine interesting patterns. Human reports on pipe leak events, such as leak messages posted on Twitter, can help identify the potential damaged region by extracting the associated geographic information. Compared with IoT measurements, human input is considered as deterministic information because it is highly likely to have pipe breaks in a region if people living around report it on OSN. The more reports, the higher the level of confidence about the event. Therefore, we incorporate human input with the predicted outcome from IoT measurements to improve the detection performance.

Data collected from Twitter represents a previously untapped resource for detecting a pipe break and locating the failure. Human input to AquaSCALE is enabled by integrating a novel Tweet Acquisition System (TAS) [110] developed at UC Irvine, which enhances the monitoring of tweets based on client/application needs in an online adaptive manner such that the quality and quantity of results improve over time. Given a group of interested patterns, TAS can extract related tweets that are then used to help track and locate leak. Twitter users are "sensors" and the posted message with a mention of water pipe break such as "*Pipe bursts @ Sunset Boulevard north of the UCLA campus.*" is an indicator of leak event. To model the human input, let $\mathcal{C} = \{c : c = \{v : |l_c - l_v| < \gamma \wedge v \in \mathcal{V}\}\}$ represent a set of subsets of $\mathcal{V}$ (i.e., a set of cliques) inferred from tweets. Here, a clique c is associated with the location l_c where people post the event, and $|l_c - l_v| < \gamma$ means that the distance between l_v (the location of node v) and l_c is less than threshold γ. The threshold γ is a pre-defined parameter indicating the coarseness of the collected Twitter data. For example, if γ is set to 1 km, nodes within 1 km distance to l_c are considered to be likely to leak and will be added into the clique c.

Although there is a high probability for a region to have a pipe break if leak message posted on Twitter, a tweet can be erroneously treated as an indicator of a leak event. For example, tweets like this "*LeakFinderST - innovative leak detection and location in water pipes.*" may be collected but it does not relate to a leak event that we wish to identify. Therefore,

we define a probability of false positive error as p_e, i.e., the likelihood of a tweet that is improperly considered to be relevant, where $0 < p_e < 1$. The confidence that there is a leak within a certain region is represented by

$$p_t = 1 - (p_e)^k \tag{4.3}$$

where k is the number of tweets collected over a period of time, and with more tweets collected the confidence p_t increases. To model the number of messages received along with the time, we use Poisson distribution that is popular for modeling the number of times an event occurs in an interval of time or space. The human input is assumed to arrive independently of the time. The average number of human reports received in a sampling interval (of IoT devices) is designated as λ that is called arrival rate. The probability of receiving k reports in n elapsed time slots is given by the equation

$$P(\mathsf{k} \text{ reports in } \mathsf{n} \text{ intervals}) = \frac{(n\lambda)^k e^{-(n\lambda)}}{k!} \tag{4.4}$$

where e is the Euler's number and $n \in \mathbb{N}$. Combine (4.3) and (4.4), the confidence that there is a pipe leak in an area can be computed.

4.4 A Composite Leak Identification Algorithm

To enable an accurate and timely leak events identification, we discuss a two-phase approach where the profile model is generated offline by learning an extensive amount of measurements in water infrastructure (Phase I) and the additional observations are integrated with the predicted results from the profile model when live data coming in (Phase II). The proposed composite algorithm reduces the detection time by orders of magnitude by generating a profile offline and improve the detection accuracy by incorporating multiple data sources.

4.4.1 Phase I: Training Profile Model Using Measurements In Water Infrastructure

In Phase I, the objective is to train a set of classifiers $f_{v\in\mathcal{V}}$ in (4.2) to generate a robust profile model f using a great amount of measurements collected in water infrastructure. Here we drop the subscript v and use f to represent a set of trained $f_{v\in\mathcal{V}}$ as the profile model. We first discuss the generation of training features and samples that are then input into the classifiers for a profile model generation. The enabling of plug and play ML based techniques allows us to explore the knowledge of which technologies work well in terms of speed and accuracy under different configurations.

Features of internal measurements in water infrastructure include the topology of the network and IoT observations. The basic topology information, denoted as T, includes node elevation, pipe length, diameter and roughness coefficient, which are static parameters for a given water network. Dynamic IoT measurements X collected from IoT sensors depend on the type and location of the devices. Techniques based on the measurements from pressure and flow rate devices allow a more effective and less costly search in situ [112]. Thus we use pressure transducers and flow meters in the paper. Water pipe leak identification is based on the premise that leakage in one or more locations of the network involves local liquid outflow at leaky points, which will change the pressure head and flow rate at monitoring points [112]. Therefore, we use the difference between the two sets of consecutive readings from IoT devices as the features of X. That is x_a is the change in pressure head or flow rate of sensor a. The dynamic IoT observations X aggregated with the static topology T are then the features of a training sample. AquaSCALE in conjunction with EPANET++ enables the selection of a sensor set $\mathcal{A}$ giving the type and number of IoT devices. It allows the study of sensor placement by evaluating different sensor configurations. The problem of identifying an optimal sensor placement for leak detection will be studied in future work. In this work, given the number of available devices, we use k-medoids algorithm to select a group of

locations as the sensor set. K-medoids is a clustering algorithm related to k-means, but it is more robust to noise and outliers [80, 81]. k-medoids partitions $|\mathcal{V}| + |E|$ potential sensor locations into certain number of clusters and assigns cluster centers as the sensor locations, based on the pressure head and flow rate read from nodes and pipes.

As discussed in Section 4.3.2, this multi-output classification problem is transformed into multiple binary classification problems where the classifier is trained separately for each potential leak location v using same datasets and its true labels denoted as Y_v. The profile model $f : T \cup X \rightarrow \mathbb{R}$ can be an ensemble of a set of linear/nonlinear predictors, decision trees, or weak learners, and the parameters of f can be learnt by ML based techniques on the basis of the analysis of pressure and flow rate variations produced by the leak. Note that the performance of specific techniques depends on the structure of water networks, the type and number of IoT devices and their deployment. AquaSCALE allows to test different techniques in isolation or combination, and a hybrid approach may improve the performance since it is thought of as a way to compensate the limitations of individual algorithms.

In the paper, we used scikit-learn package for data processing and analysis [97], and compared multiple well-known ML algorithms including Linear Regression (LinearR), Logistic Regression (LogisticR), Gradient Boosting (GB), Random Forest (RF) and Support Vector Machine (SVM). We proposed a hybrid approach named HybridRSL, a combination of RF and SVM via LogisticR, because RF and SVM remain robust with decreasing number of IoT sensors, and LogisticR has low variances and is less prone to overfitting. As shown in Fig. 4.3, the same dataset is trained and predicted by RF and SVM separately, and their predicted results, i.e., leak probabilities for each node, are then aggregated as a new feature set and input into LogisticR for further learning. Algorithm 1 shows how classifiers are trained and updated to generate the profile model for Phase II.

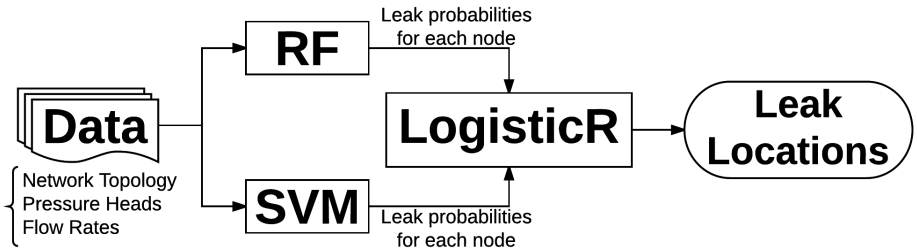


Figure 4.3: A sketch of the workflow of HybridRSL approach.

Algorithm 1 Training the Profile Model

1: **Input** water network topology T, IoT measurements X, true leak events Y_v and classifiers f_v for $v \in \mathcal{V}$
2: **Output** the profile model $f = \{f_v : v \in \mathcal{V}\}$
3: **Objective** update f_v to best fit training samples

4: **for** v in V **do**
    $f_v.\mathsf{fit}(T, X, Y_v)$
5: **end for**

4.4.2 Phase II: Inferring Leak Locations Using Live Ingress Data

In Phase II, we sequentially aggregate multiple data sources to infer the leak locations. Compared with human input, IoT measurements and ambient temperatures are relatively stable data sources. We can expect telemetry readings from these two sources at a certain interval once the sensing devices are deployed. Due to dynamic and complex social behavior, however, human reports on leak event may not be available. Therefore, we first use IoT and temperature streams for event prediction, and use additional human input for event tuning.

The live IoT observations $\mathbf{x} = \{x_a : a \in \mathcal{A}\}$ together with the topology information T are firstly learnt by the profile model f. It uses predict_proba and predict methods built in the scikit-learn package, whose outcomes are the score/probability of leak for each node, i.e. $P = \{p_v(i) = \mathsf{score}(y_v = i) : 0 \leqslant p_v(i) \leqslant 1 \wedge \sum_i p_v(i) = 1 \wedge i = \{0, 1\} \wedge \forall v \in \mathcal{V}\}$

with $y_v = 1$ indicating having a leak at node v, and a subset of $\mathcal{V}$ that are predicted to leak, i.e. $\mathcal{S} = \{v : p_v(1) > p_v(0) \wedge v \in \mathcal{V}\}$. If the ambient temperature is below $20°F$ and a location v is detected to be frozen, its predicted leak probability $p_v(1)$ will aggregate with $p_v(\text{leak}|\text{freeze})$ based on Bayes' theorem [32]. This is a well-known method to combine probability distributions from experts in risk analysis, and to apply it into AquaSCALE, we simply consider each information source as an expert. The updated leak probability at node v is

$$p_v^*(1) = \frac{q_v^*(1)}{1 + q_v^*(1)} \tag{4.5}$$

where

$$q_v^*(1) = \prod_{j=1}^{n} \frac{g_{1j}(p_j|q_v = 1)}{g_{0j}(p_j|q_v = 0)} \tag{4.6}$$

$q_v^*(1)$ is the posterior odds of the occurrence of leakage at node v; g_{1j} (g_{0j}) represents the probability of source j giving probability p_j conditional on the occurrence (non-occurrence) of leakage at node v. Here, the predicted probabilities come from two information sources, IoT measurements and weather data. In this manner (4.5), more sources of information means more certainty. For example, if the probability of leak is 0.6 predicted by both two sources, then $p_v^*(1)$ will tend to be much higher than 0.6. The aggregated results then updated P and S correspondingly. In set $\mathcal{S}$, potential faulty pipes are identified. However ML based techniques with noisy IoT sensing data work on the predictive perspective whose output is probabilistic. We use entropy to measure the uncertainty of a predicted event (leak or not) at node v on the basis of its leak probability, which is defined as

$$H(y_v) = -\sum_{i=0}^{1} p_v(i) \log p_v(i) \tag{4.7}$$

The corresponding uncertainty function is given by

$$E[\mathbf{y}] = \sum_{v \in \mathcal{V}} H(y_v) \tag{4.8}$$

In order to minimize (4.8), we integrate additional human input to help to enhance the knowledge of leaks and increase the determinacy of the predicted events.

Human reports on leak events as deterministic information are able to correctly reflect pipe failures within a certain region, but are unable to specify an exact damaged position due to various social behaviors. Therefore, the human input is used as an additional subzone-level information, working with the pipeline-level outcomes P and $\mathcal{S}$, to enforce the event consistency and improve the prediction results. The event consistency here refers to the consistency of the pipeline-level and subzone-level predictions. An inconsistent event means that none of pipes in the subzone identified by human input is currently predicted to leak. To leverage the human input, we apply the higher order potential concepts used in the image segmentation problems, which is used to enforce label consistency in image regions [70]. We define a higher order potential function $\Phi_c : \mathcal{L}^{|c|} \rightarrow \mathbb{R}$ on clique c to assign a cost to each possible configurations (or labelings) of $\mathbf{y}$. By incorporating human input, (4.8) can now be written as

$$E[\mathbf{y}] = \sum_{v \in \mathcal{V}} H(y_v) + \sum_{c \in \mathcal{C}} \Phi_c \tag{4.9}$$

that is the energy function to be minimized. Because the effects of human input is considered to be non-negative in the paper, Φ_c can assign a very high cost to clique c if none of nodes in c is currently predicted to leak, i.e. $\nexists v \in \mathcal{S}$ for $\forall v \in c$. In this case, the node in clique c with the highest entropy (uncertainty) will be selected for further processing.

The higher order potential used by us can be written as

$$\Phi_c = \begin{cases} 0 & if \quad \exists v \in \mathcal{S} \;\; for \;\; v \in c \\ 0 & else\, if \;\; H(y_v) < \Gamma \;\; for \;\; \forall v \in c \\ Inf & else \end{cases} \tag{4.10}$$

Here we introduce a threshold Γ to decide if a pipeline-level prediction is considered to be determinate enough to ignoring the subzone-level information. That is if the entropy for node v is less than the threshold Γ meaning that the current predicted event is likely to occur, then the leak information on node v will not be updated by human input. According to (4.9) and (4.10), an inconsistent event can push the energy to the infinity. In order to minimize (4.9), in Algorithm 2, a set of leak locations S is firstly identified by the profile model f and then updated based on clique c by adding a candidate v^* if $\Phi_c = Inf$ and $v^* = \arg\max_{v \in c} H(y_v)$. Correspondingly, $p_{v^*}(0)$ and $p_{v^*}(1)$ will be updated to 0 and 1, and $H(y_{v^*})$ will be 0. In this manner, the inconsistent events will be forced to change and the total energy will be reduced because the infinite potentials are eliminated and the entropy of certain nodes are reduced.

4.5 Experimental Study - Using AquaSCALE for Leak Event Identification

In this section, we evaluated the proposed identification approach on single- and multi-failure scenarios, tested multiple ML based techniques in isolation and combination, and examined the impact of incorporating IoT measurements and additional observations. We begin by describing the setup and datasets under which the experiments are conducted, and introduce the performance metrics and the results.

Algorithm 2 Inferring Leak Events

Input water network topology T, IoT measurements $\mathbf{x}$, profile model f, leak probability due to frozen $p_{v\in\mathcal{V}}(\text{leak}|\text{freeze})$ and human input $\mathcal{C}$
Output an updated set of leak locations $\mathcal{S}$
Objective minimize $E[\mathbf{y}]$ in (4.9)

/* Event Prediction */
$P = f.\text{predict\_proba}(T, \mathbf{x})$; $\mathcal{S} = f.\text{predict}(T, \mathbf{x})$
for v in $\mathcal{V}$ **do**
    if v is detected to be frozen **then**
        $q_v^*(1) = \frac{p_v(1)}{p_v(0)} * \frac{p_v(\text{leak}|\text{freeze})}{1-p_v(\text{leak}|\text{freeze})}$
        $p_v(1) = \frac{q_v^*(1)}{1+q_v^*(1)}$
        $p_v(0) = 1 - p_v(1)$
        $\mathcal{S} = \mathcal{S} \cup \{v\}$ **if** $p_v(1) > p_v(0)$
    end if
end for

/* Event Tuning */
$\mathcal{C} = \{c : c = \{v : |l_c - l_v| < \gamma \wedge v \in \mathcal{V}\}\}$; $\Phi_{c\in\mathcal{C}} = \text{Inf}$
for c in $\mathcal{C}$ **do**
    if $\exists v \in \mathcal{S}$ for $\forall v \in c$ **then**
        $\Phi_c = 0$, break
    end if
    if $\Phi_c \neq 0$ **then**
        $v^* = \arg\max_{v\in c} H(y_v)$
        if $H(y_{v^*}) > \Gamma$ **then**
            $p_{v^*}(1) = 1$, $p_{v^*}(0) = 0$, $\mathcal{S} = \mathcal{S} \cup \{v^*\}$
        end if
    end if
end for

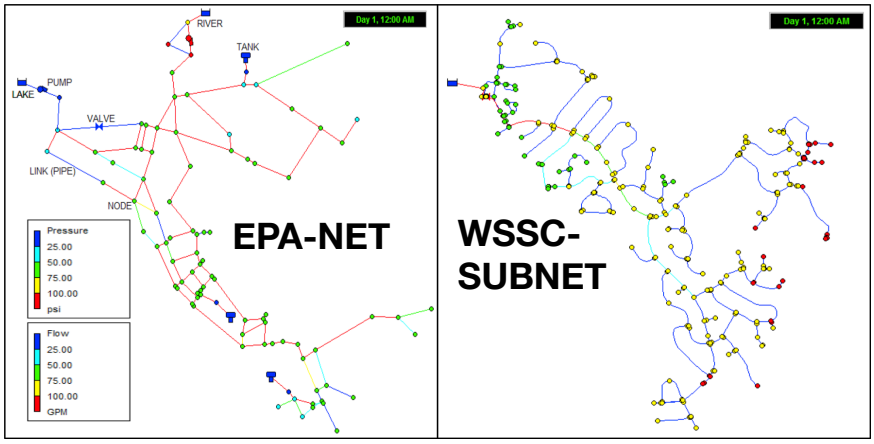


Figure 4.4: A graph representation of EPA-NET - a canonical water network provided by EPANET with 96 nodes, 118 pipes, 2 pumps, one valve, 3 tanks and 2 water sources, and WSSC-SUBNET - a subzone of WSSC service area with 299 nodes, 316 pipes, 2 valves and one water source.

4.5.1 Experimental Setup and Datasets Generation

Water Networks. AquaSCALE is evaluated using two water networks - a canonical water network provided by the EPANET (named EPA-NET) and a real subzone of WSSC water service area provided by WSSC (named WSSC-SUBNET). A graph representation of EPA-NET with $|\mathcal{V}| = 96$ and $|E| = 118$, and WSSC-SUBNET with $|\mathcal{V}| = 299$ and $|E| = 316$ is shown in Figure 4.4. The elevation of pipes varies with the topography, and each pipe has four attributes - length, diameter, roughness coefficient, and status (open or close controlled by a valve). Each node has a pattern of time variation of the demand (i.e. consumption), and leak events are simulated at nodes. EPANET++ is used to perform extended period simulation of hydraulic behavior, which computes pressure heads at nodes and flow rates at pipes.

IoT Sensing Data. Extensive simulations are run on these two networks using EPANET++. Given the number of devices, we first identify a set of sensor locations, and generate a great amount of IoT measurements for profile training. As mentioned, features of a training sample are the topology of a water network and changes on sensing values. The number of

training and testing samples are $20,000$ and $2,000$ respectively. For each simulation run, there is at least one and at most 5 leak events, and the number of events follows the uniform distribution i.e., $U(1,5)$. The leak events are generated with arbitrary locations and sizes but same starting time, since we aim to study concurrent failures that are harder to pinpoint. The sensor set $\mathcal{A}$ is selected using k-medoids algorithm based on the given information of IoT devices. The sampling frequency of IoT devices is 15 minutes. Since the goal is to identify leak locations, we assume that the leak starting time $e.t$ is known. The change in pressure heads and flow rates is then computed by taking the differences between the sensing values at $e.t-1$ and $e.t+n$, where n is the number of elapsed time slots after leaking, as in (4.4).

Human Sensing Data. From January 6, 2016 to April 1, 2016, the east coast of the US experienced extremely cold temperatures, while the west coast experienced high precipitation due to El Niño effects. We collected 30 million "leak-related" tweets posted in the US during this period using TAS system. Since this data contains significant noise, it was treated as described (Sec. 4.3.4). Based on the result statistics, the arrival rate λ of human input is set to 1 per 15 minutes, and the false positive error p_e is set to 0.3. The coarseness parameter γ, determining the clique c, is set to different values to test the impact of incorporating with human input. More in-depth analysis of those tweets, such as the distribution among different facilities and how soon after the leaks are the tweet posted, will be discussed in future work. In Algorithm 2, the node in c with the highest entropy will be considered as the most risky point, and it will be predicted to leak if the entropy is greater than threshold Γ. Here Γ is set to 0 to always consider human effect. For each simulation run, given an elapsed time slot n, a random number between 0 and 1 is generated for obtaining the number of received tweets k based on (4.4), and the confidence probability p_t can then be computed based on (4.3). With the lapse of time, more human reports can be collected to help identify pipe failures.

Environmental information - Ambient Temperatures. In the paper, the probability

p_v(freeze) and p_v(leak|freeze) are set to 0.8 and 0.9 respectively for all $v \in \mathcal{V}$. It might be different for every node since the vulnerability to low temperatures depends on a variety of factors, e.g., material, age, location, which will be studied in future work. For each simulation run, a random number between 0 and 1 is generated for each node and it will be used to decide if the connected pipe is frozen based on the pre-defined probabilities. It is likely to have more pipe failures under extreme cold temperatures, which will be used to drive failure scenarios.

Failure Scenarios. We evaluate the proposed composite algorithm of pipe leak identification through two-failure scenarios over different evaluation strategies. We generated $20,000$ single- and multi-failure scenarios separately for training and $2,000$ for each for testing. The single pipe failure represents that there is only one leak event, which is denoted as $e = \{l, s, t\}$. While multiple pipe failures represents that multiple leak events occur simultaneously, denoted as a set of events $\mathbf{e} = \{e_i : i = 1, ..., m\}$ where m is the number of leaky points and $e_i = \{l_i, s_i, t_i\}$. Multi-failure is often caused by the ice blockage in winter, thus *Pipe Failures due to Low Temperature* is considered as the use case of multiple leaks. The faulty pipes will be located by using different strategies - measurements in water infrastructures with diverse ML based techniques, weather information, and/or human input.

4.5.2 Performance Metrics

The effectiveness of the proposed algorithm is evaluated in terms of following metrics.

Hamming Score is defined as $\sum_{v \in \mathcal{V}} \frac{\mathbb{1}[\hat{y}_v = 1 \wedge y_v = 1]}{\mathbb{1}[\hat{y}_v = 1 \vee y_v = 1]}$ where $\mathbb{1}$ is an indicator function. It is the number of leak events correctly predicted divided by the union of predicted and true leak events. The score is bounded by 1 and the higher the score the greater number of leaks that are identified.

Precision is defined as $\sum_{v\in\mathcal{V}} \frac{\mathbb{1}[\hat{y}_v=1 \wedge y_v=1]}{\mathbb{1}[\hat{y}_v=1]}$, which is the number of leak events correctly predicted divided by the total number of events labeled as leak. It is the fraction of predicted leak events that are true. A perfect precision score of 1 means that every predicted leaky node does indeed leak but says nothing about the number of leaky nodes that are labeled correctly.

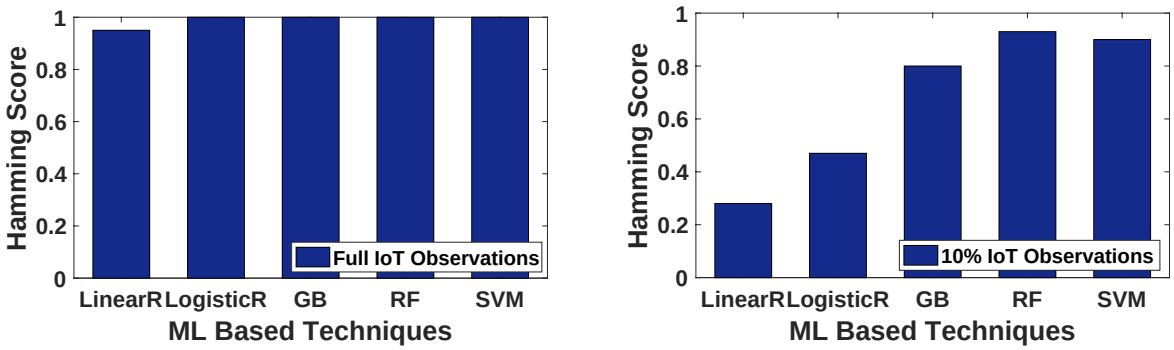
Recall is defined as $\sum_{v\in\mathcal{V}} \frac{\mathbb{1}[\hat{y}_v=1 \wedge y_v=1]}{\mathbb{1}[y_v=1]}$, which is the number of leak events correctly predicted divided by the total number of true leak events. In contrast with precision, recall is the fraction of leak events that are actually identified. A perfect recall score of 1 indicates that every leaky node is identified but says nothing about how many other nodes are incorrectly also labeled as leak.

Percentage of IoT Observations is the percentage of IoT deployment penetration. In practice, we want to reduce the number of devices since the installation and maintenance are very expensive. Here $\mathcal{A} = \mathcal{V} \cup E$ with $|\mathcal{A}| = |\mathcal{V}| + |E|$ refers to the full (100%) IoT observations.

Elapsed Time Slot is the number of time slots elapsing after the leak event, denoted as n. A time slot is a 15 minutes time interval, determined by the sampling frequency of IoT device. With n increasing, on one side, more observations including IoT data and human input will be collected, which may provide more information. On the other side, it may also waste more water, and increase the risk to public health due to water contamination and other infrastructures due to cascading events.

4.5.3 Experimental Results

In this section, the proposed approach for leak event identification is validated through a detailed simulation study. We begin by plugging and playing several ML based techniques for both single- and multi-failure scenarios using EPA-NET network, and apply Hybrid-RSL



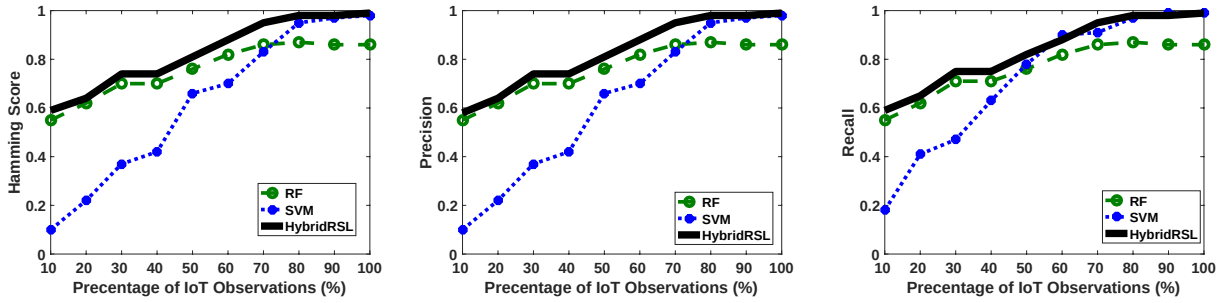
(a) Full IoT Observations (b) 10% IoT Observations

Figure 4.5: EPA-NET with *Single Failure* - Comparison of ML techniques for single leak identifications using (a) full and (b) 10% IoT observations.

technique that outperforms others for following experiments. The effectiveness of integrating diverse data sources for failure detection is evaluated by running extensive simulations on both EPA-NET and WSSC-SUBNET networks. Flood as a cascading event is modeled and predicted to help explore the impact of pipe failures.

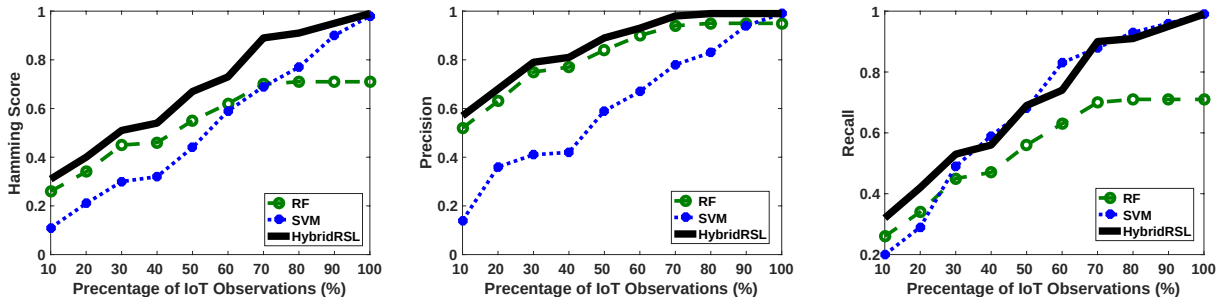
Figure 4.5 illustrates the comparison of different ML based techniques for single leak identification. Those techniques have similar high hamming scores as using 100% IoT observations (Fig. 4.5a), while RF and SVM can keep a better performance even with 10% IoT (Fig. 4.5b). Figure 4.6 and 4.7 show the comparison of RF, SVM and Hybrid-RSL in terms of hamming score, precision and recall for single- and multi-failure scenarios. With a lower percentage of IoT observations, RF yields a better performance compared with SVM. With more IoT data available, SVM outperforms RF as using around 70% IoT in multi-failure scenarios (Fig. 4.7). With the aggregation of RF and SVM, HybridRSL has the best performance in both single- and multi-failure cases. It also shows that multi-failure is much harder to locate. Other ML and data fusion techniques can also be plugged and tested using AquaSCALE.

In the following result, HybridRSL with the highest score will be used, and the integration of multiple data sources will be examined. Here the distance threshold γ for human input is set



(a) Hamming Score (b) Precision (c) Recall

Figure 4.6: EPA-NET with a Single Failure - Comparison of RF, SVM and Hybrid machine learning techniques in terms of hamming score, precision and recall for single leak identifications.



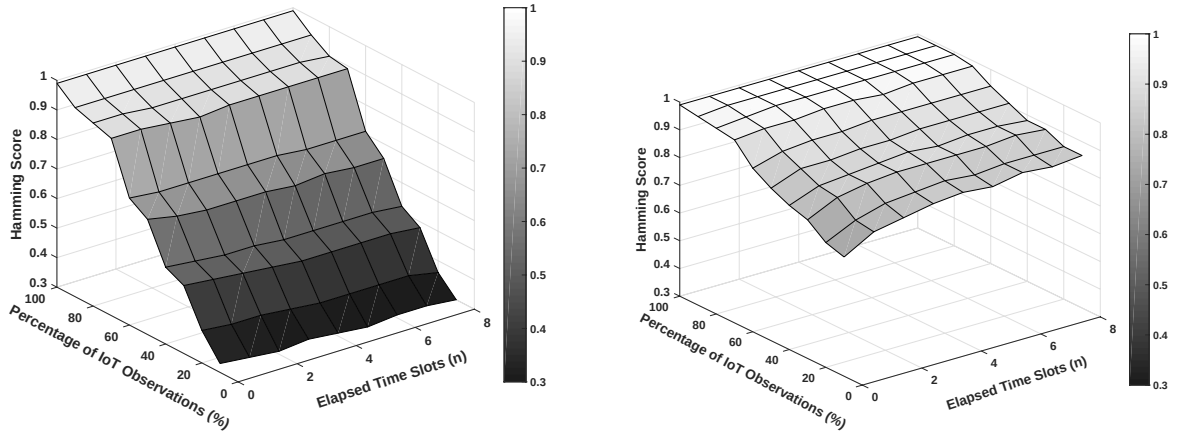
(a) Hamming Score (b) Precision (c) Recall

Figure 4.7: EPA-NET with Multiple Failures - Comparison of RF, SVM and HybridRSL techniques in terms of hamming score, precision and recall for multiple leaks identifications.

to 30 meters. Figure 4.8 and 4.10 describes how much do weather and human data together contribute toward identifying *Multiple Failures due to Low Temperature* using EPA-NET and WSSC-SUBNET water networks. In Fig. 4.8a and 4.10a, only IoT data is applied, and the result obtained by aggregating temperatures and human input is shown in Fig. 4.8b and 4.10b. The plotted surface shows how the hamming score varies with the percentage of IoT observations and the elapsed time slots. It clearly illustrates that the integration of weather and human information is robust for locating leak events even with limited IoT data. Incorporating with human input can increase the score, however, more human input as the time elapsing do not provide significant improvement. Because the false positive error of human data is set to a small number in our experiments. Figure 4.9 and 4.11 present the increment on hamming score by incorporating weather and human data, and the incrementation is more significant with less IoT data. Figure 4.12 shows that the efficacy of incorporating with human input decreases with the coarser Twitter data. Combined with temperature information, it can compensate for the impact of loose human data and yield a better performance with the higher score. In Fig. 4.13, detection using only IoT data is sensitive to the maximum number of leak events, but the aggregation of additional information can help locate failures and output a better result.

4.5.4 Exploring Impact - Flood modeling and prediction

To capture the impacts of pipe failures and improve post-event awareness, we incorporate flood modeling and prediction to study cascading events. We apply the BreZo simulator for flood prediction on WSSC-SUBNET water network. BreZo is a hydraulic model and has been successfully applied in simulation of dam breaks [17, 18] and floods [94, 93]. It can efficiently simulate water flows in varying shapes of the earth's surface. A detailed description can be seen in [25]. In BreZo, the flood is predicted based on the digital elevation map (DEM), interpolated from node elevations, shown in Fig. 4.14a. To feed leak information into the



(a) IoT (b) IoT + Weather + Human

Figure 4.8: EPA-NET with Multiple Failures due to Low Temperature - Average hamming score for multiple leaks identifications using (a) IoT observations and (b) the aggregation of multiple data sources.

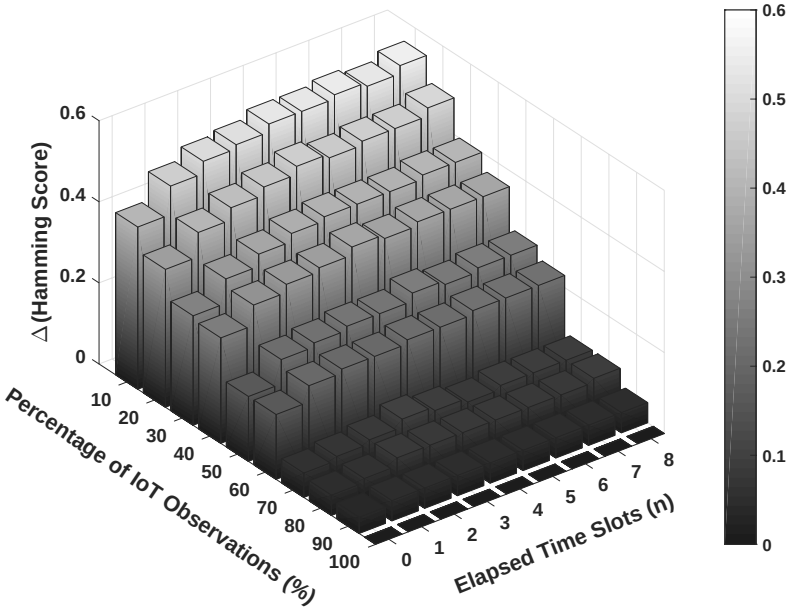
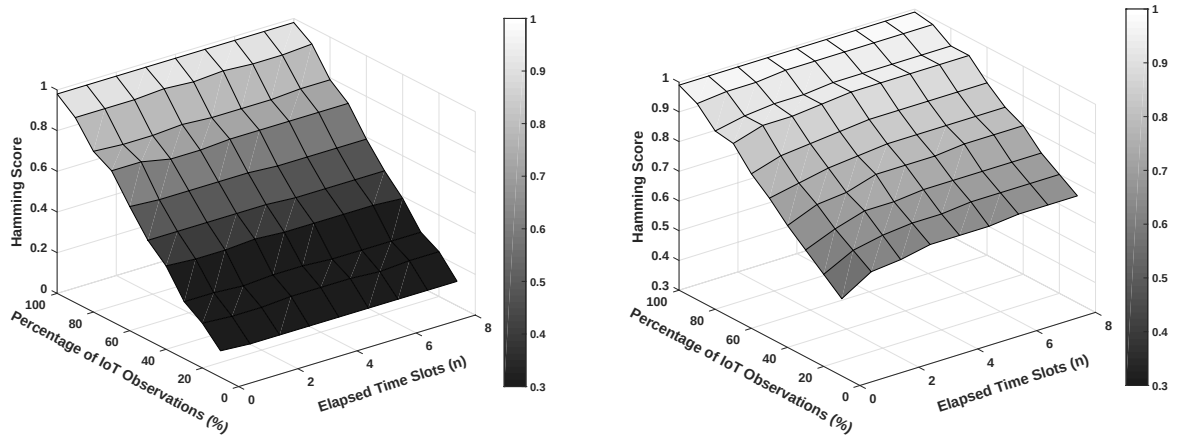


Figure 4.9: EPA-NET with Multiple Failures due to Low Temperature - Average increment on hamming score by adding weather and human input.



(a) IoT (b) IoT + Weather + Human

Figure 4.10: WSSC-SUBNET with Multiple Failures due to Low Temperature - Average hamming score for multi-leak identifications using (a) IoT and (b) multiple data sources.

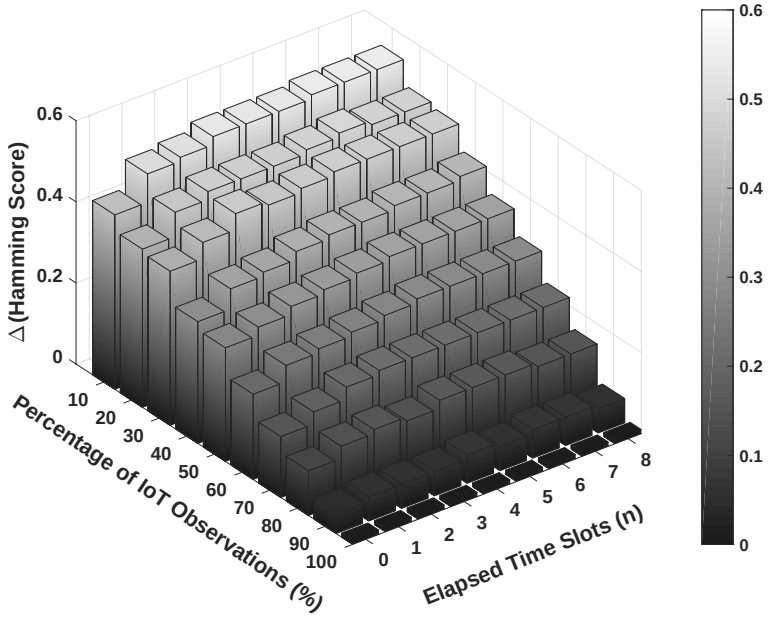


Figure 4.11: WSSC-SUBNET with Multiple Failures due to Low Temperature - Average increment on hamming score by adding weather and human input.

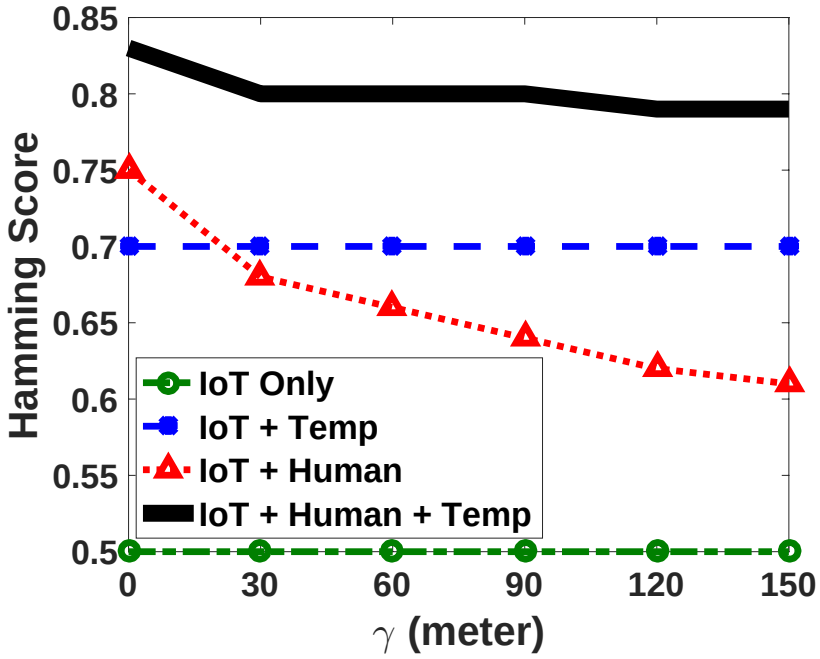


Figure 4.12: WSSC-SUBNET with *Multiple Failures due to Low Temperature* - Average hamming score with coarser twitter data using different information sources.

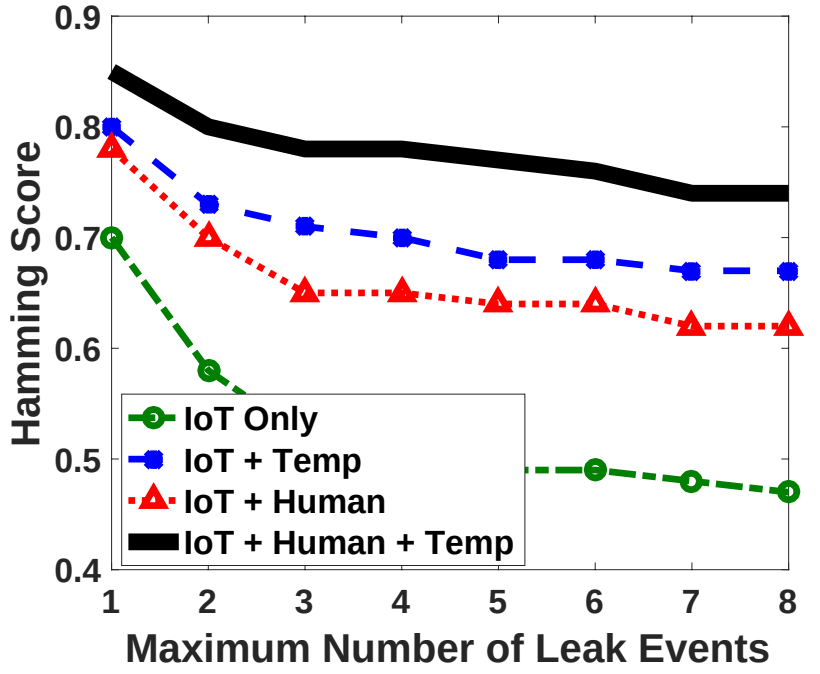
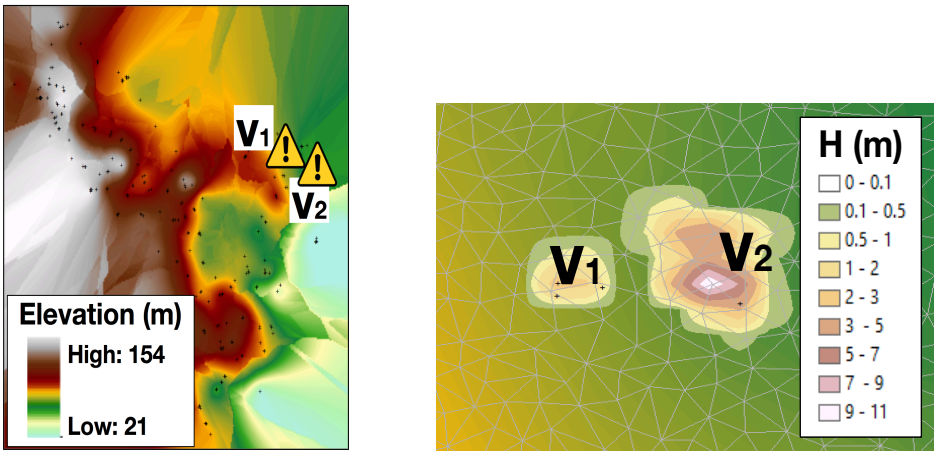


Figure 4.13: WSSC-SUBNET - Average hamming score for failure identifications with increasing number of leak events using different information sources.



(a) DEM (b) Flooding Map

Figure 4.14: Flood prediction based on (a) DEM of WSSC-SUBNET with leaks at v_1 and v_2. (b) Zoom-in flooding map overlaying over the DEM. Flood flows from the center to the outer. H represents the flood depth in meter.

flood model, we use (4.1) to calculate the outflow rate based on pressure readings, which is then input into BreZo for flood simulations. Two leak events are simulated at v_1 and v_2 with different leak sizes but same start time, and Fig. 4.14b shows that the flood spreads along the earth's surface. This information can be used by water agencies and city planners for damage control, community notifications and evacuation plans.

4.6 Chapter Summary and Discussion

In this chapter, we leverage the AquaSCALE framework to localize multiple concurrent pipe failures in community water infrastructures. We formulate multi-leak identification problem, develop an ML based integration mechanism for efficiently fusing information from multiple sources, and evaluate it using real-world water networks.

In this work, we first develop and evaluate a range of interventions in the AquaSCALE platform to understand the structure, components and operations of this CPHI system, which is the first step to design robust systems under different situations. Given that, we apply it to

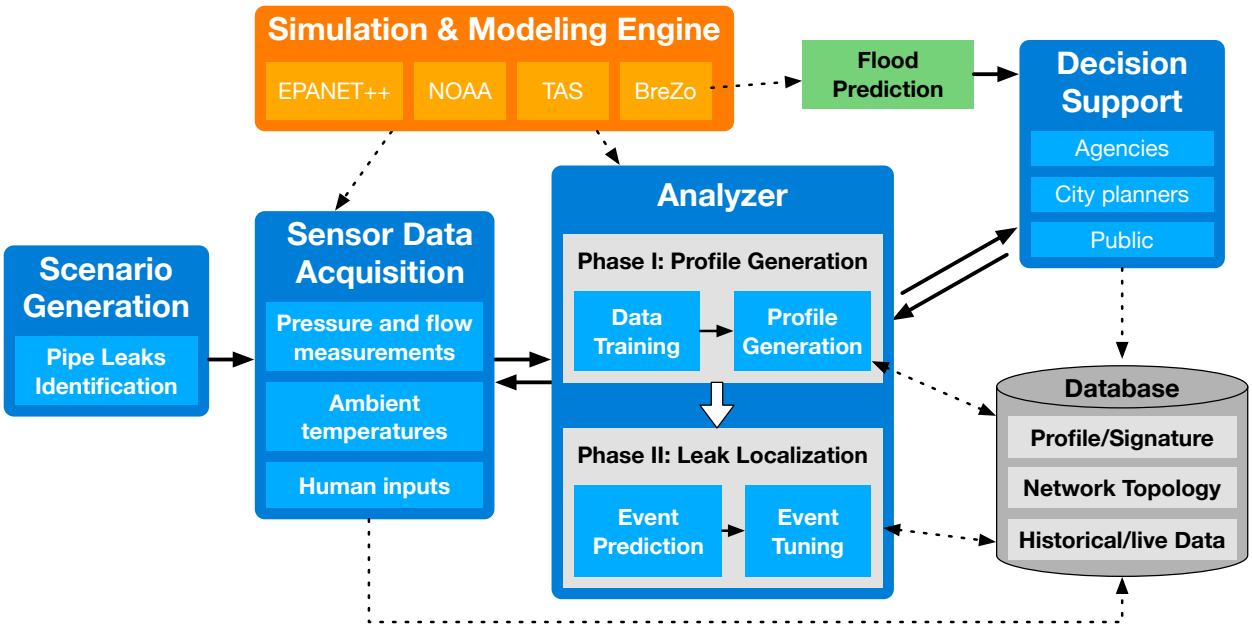


Figure 4.15: This chapter leverages the AquaSCALE system architecture for multi-leak isolation under a small number of failures.

isolate multiple faults under small number of failures (Fig. 4.15), where data from multiple information sources including infrastructure measurements, ambient temperatures, human inputs from social media, and domain models is gathered. In order to efficiently utilize and fuse these data with different characteristics (e.g., spatial granularity, sampling frequency and uncertainty), we propose a two-phase workflow that begins with robust simulation methods using a commercial grade hydraulic simulator, enhanced with the support for IoT sensor and failure modelings. It generates a profile of anomalies using a hybrid ML method. The profile then incorporates with live sensor readings, temperatures and human inputs (if available) to isolate broken pipelines. This joint analysis of heterogeneous data captures the spatial and temporal coherence at different levels of granularity, which can inform the decision-making. Our initial study shows that the integration of spatiotemporal data sources supported by simulation engines and ML techniques can generate near real-time information with high level of accuracy and detection time reduced by orders of magnitude.

Chapter 5

Infrastructure Resilience under Extreme Events

In this chapter, we explore the resilience of water infrastructures under extreme events (e.g., earthquakes), where massive failures can occur. In our previous work (Chapter 4), since we start with the consideration of failures caused by the operational degradation, the maximum number of pipe breaks is set to 8 (less than 3% of the number of network nodes). It draws into question efficient isolation of a large number of failures caused by e.g., earthquakes.

Water distribution infrastructures often exhibit extreme fragilities during large-scale disasters resulting in massive pipe breaks, water contamination and disruption of service. To monitor and identify potential problems, hidden state information must be extracted from limited and noisy data environments. This requires estimating the operating system states of the water infrastructures quickly and accurately. We therefore present a graphical model based approach for on-line state estimation of water distribution networks during large-scale disasters. We model the water network as a factor graph, characterizing the non-linearity of fluid flow in the network that is dynamically altered by leaks, breaks and operations de-

signed to minimize water loss. The approach considers a structured probabilistic framework that models complex interdependencies within a high-level network topology. The proposed two-phase approach, which begins with a network decomposition using articulation points followed by the distributed Gauss-Newton Belief Propagation (GN-BP) based inference, can deliver optimal estimates of the system state in near real-time. The approach is evaluated in canonical and real-world water networks under different levels of physical and cyber disruptions, using the Water Network Tool for Resilience (WNTR) recently developed by Sandia National Lab and EPA. Our results demonstrate that the proposed GN-BP approach can yield an accurate estimation of system states (mean square error 0.02) in a relatively fast manner (within 1s). The two-phase mechanism enables the scalability of state estimation and provides a robust assessment of performance of large-scale water networks in terms of computational complexity and accuracy. A case study on the identification of "faulty zones" shows that 80% broken pipelines and 99% loss-of-service to end-users can be localized.

5.1 Chapter Overview

Natural disasters, e.g., earthquakes, hurricanes and winter storms, and other types of hazards have resulted in different types of water service disruptions, and caused financial, social, environmental and human health consequences [37, 69]. The ability to maintain delivery of water supplies during and after catastrophic events is critical to ensure public safety and welfare. Additionally, fire-following hazards are of major concern, especially in highly-dense, urban areas. Water infrastructure resilience is important not only for individuals, but also for hospitals, fire stations, schools and for other industries that rely on water.

In this work, earthquakes are particularly concerning since buried water pipelines are extremely vulnerable to damage from earthquake-caused ground failures, e.g., liquefaction, landslide, surface faulting and other effects. Water infrastructures consist of large and diverse networks, constructed over long periods of time, crossing diverse geologic conditions, and as a result are expected to suffer damage after large earthquakes [41]. For example, the 1994 magnitude 6.7 Northridge Earthquake (US) damaged over 1,000 distribution pipes and caused 7 days water outages; the 2010 magnitude 8.8 Chile Earthquake damaged 3,000 distribution pipes and caused over a month of water outages; the 2011 magnitude 9.0 Tohoku Earthquake (Japan) damaged thousands of distribution pipes and caused several months of water outages. Furthermore, a recent earthquake report published by the Water Research Foundation indicates that the bulk of the damage to infrastructures as well as the resulting loss of access to facilities and supplies are mainly due to failure of hundreds/thousands of distribution pipelines in zones of infirm ground. Over the past 25 years or so, many U.S. water utilities in high seismic hazard zones have been upgraded with seismic-resistant pipes or joints at a rate of 0.1% to 0.3% per year (about a 100 to $1,000$ year replacement cycle) [41]. Although the level of damage can be controlled or minimized using proven seismic retrofit practices, we should never expect water infrastructures to be damage free after a significant earthquake.

Once pipeline networks are damaged, water service areas will immediately shutdown via closure of valves. In the meantime, potable water is distributed to customers using mobile water tankers, while service crews are dispatched to repair and restore the system to normal operating conditions. Without a robust and efficient approach to localize the damaged component or section, water agencies are forced to shutdown a greater area to minimize the loss of cascading failures, which may impact a greater population and cause significant financial losses due to long out-of-service periods. This scenario highlights the need for a holistic and efficient state estimation that produces estimates of the current operating states, and helps detect, locate and prevent possible secondary failures in the water infrastructure. An efficient hydraulic state estimation enables timely countermeasures that can mitigate and limit failure propagation, e.g., cascading failures such as release of waste, flooding and possible contamination.

Despite promising applications, the actual implementation of a real-time monitoring and measurement platform that adapts to perturbations caused by disruptive events is lacking. One reason is that, as mentioned in Sec.1.4, water flow and pressures are generally not monitored in real-time at an individual customer level (i.e., households). Water is a relatively inexpensive resource. Consequently, most water networks are metered only for billing purposes, and there is no intelligent supervisory control and data acquisition (SCADA) system on distribution pipelines. However, civil engineers advocate that the next generation water networks will not be passive water delivery systems, but active highly-distributed event-based control systems [7]. Such a dynamical system will heavily rely on an efficient operating state estimation to facilitate effective water management under dynamic and nondeterministic environmental changes.

Challenges: The analysis of hydraulic behaviors requires an accurate representation of network topology as well as real-time measurements of water flows and pressures. However, instrumenting the entire system of underground pipelines with sensing devices (pressure

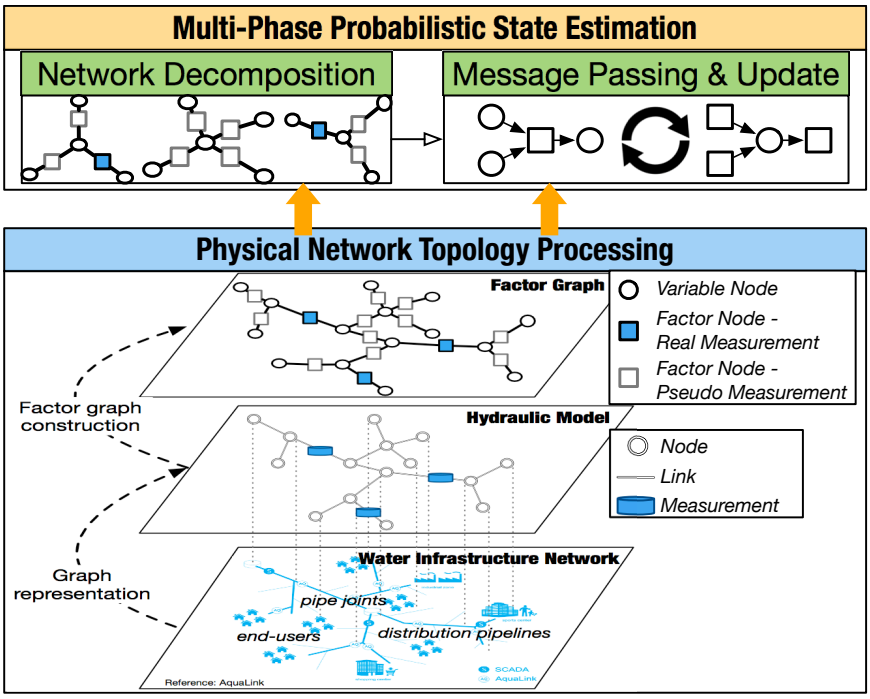


Figure 5.1: Design of probabilistic model based state estimation.

transducers and/or flow meters) is both unfeasible (inaccessibility of locations) and expensive. Also urban water infrastructures are densely connected and complex networks crossing diverse geologic conditions, where system performance measurements are highly correlated. It is non-trivial to infer operational states even with a complete observation. When limited numbers of metering devices are available, probabilistic state estimation can serve as a useful technique to "fill-in" missing performance data as well as "smooth-out" noisy measurements. Upon convergence, the optimization should reflect the current state of the water network which, in turn, should allow the prioritization of immediate responses and after-event repairs, and eventually restoration of the system.

In this chapter, we provide a systematic study of water CPHIs performance estimation under limiting conditions, i.e., less than fully instrumented network and noisy data environment, and combine this with distributed graphical model to identify failures caused by disasters. This is a key step to the aforementioned distributed event-based control system. Though inspired by water infrastructure resilience under earthquakes, the proposed methodology is

designed for generic state estimation and analysis for other pipe based CPHI systems and beyond.

Key Contributions of This Chapter:

• **Network topology processing (Fig.5.1):** Design of a methodology that formulates the water network as a hydraulic model with measurement configurations, and transfers it into a factor graph representation to incorporate non-linear hydraulic principles within a structured probabilistic framework - (Sec.5.2).

• **Probabilistic state estimation (Fig.5.1):** A novel two-phase approach for improving the speed and accuracy of state estimation on the constructed factor graph: (I) split the water network into conditional independent components using articulation points; (II) estimate the hydraulic states using a distributed Gauss-Newton Belief Propagation based approach - (Sec.5.3).

• **Evaluation in real-world water networks:** Design of a series of experiments to explore the performance with respect to the time complexity and accuracy of the proposed approach on real-world water networks provided by the EPA and WSSC - (Sec.5.4).

• Extensive evaluations under different levels of physical and cyber disruptions, and a case study on the faulty zones identification - (Sec.5.4).

5.2 Modeling Water CPHI System using Graphical Model

An efficient system state estimator requires to provide optimal estimations under dynamic and nondeterministic operational and environmental changes. In order to properly model the system stochastic properties and to conduct computationally-tractable inferences, we propose a graphical model description of the water system. It can discover and analyze

desired informative data by abstracting the physical nature into a cyber network of nodes and links, such that nodes interact with each other along their incident links in a distributed message-passing manner. Specifically, we model hydraulic heads at water nodes as state random variables on the graph vertices, and the edges of the graph determine the interaction of state variables according to the hydraulic physical law (i.e., Hazen–Williams equation [108]). Viewed together, the graphical model is specified by the joint density of hydraulic head random variables in the network for state estimation, subject to the constraints imposed by the fundamental fluid mechanics.

5.2.1 Graphical Model based Inference

We first briefly introduce the concept of graphical model and motivate a graphical model based approach for efficient state estimation. Graphical models are used to represent the conditional independence relationships among a set of random variables. It has been successfully deployed in many fields, such as computer visions [86], gene regulatory networks [140], medical diagnostics [140], communication systems [42], and recently power grids [139]. Belief propagation (BP) is an efficient message-passing algorithm that gives exact inference results in linear time for tree-structured graphs [74]. Though widely used, tree-structured models possess limited modeling capabilities, and many stochastic processes arising in real-world applications cannot be well-modeled by cycle-free graphs [62]. Loopy belief propagation (LBP) is an application of BP on loopy graphs, however, the convergence and correctness of LBP are not guaranteed in general. LBP has fundamental limitations when applied to graphs with cycles: local message-passing cannot capture the global structure of cycles, and thus can lead to convergence problems and inference errors. [79] presents a feedback message passing algorithm for an efficient inference in loopy models, which makes use of a special set of vertices whose removal results in a cycle-free graph. Inspired by the exciting results made available by graphical models, we consider a graph representing the water distribution

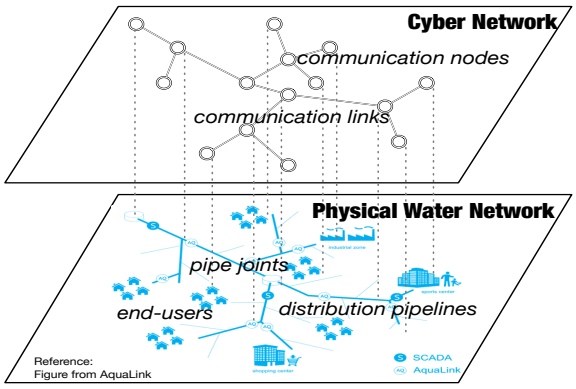


Figure 5.2: Water physical network and its cyber network with end-users and pipe joints as communication nodes, and distribution pipelines as communication links.

infrastructure as a probabilistic model, and apply BP based methods for state inference. The BP method has shown its capability to break the current centralized monitoring architecture that requires both large communication and computation overhead and tight data synchronization. It enables the estimation of unknown or uncertain parameters in a distributed message-passing manner.

5.2.2 Graphical Model of Water Systems

A water system is defined as an undirected graph $\mathcal{G}(\mathcal{V}, \mathcal{E})$ (water can flow in both directions) with vertices $\mathcal{V} = \{1, ..., n\}$ that represent nodes (end-users – nodes with demand, and junctions – pipe joints), and edges $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ that represent transmission/distribution pipelines (Fig.5.2). The set of measurements is defined as $\mathcal{M}$ that is connected to the graph $\mathcal{G}$. There are two kinds of measurements, real measurements and pseudo measurements denoted by $\mathcal{M}_{\mathrm{R}}$ and $\mathcal{M}_{\mathrm{P}}$ respectively, where $\mathcal{M}_{\mathrm{R}} \subseteq \mathcal{M}$, $\mathcal{M}_{\mathrm{P}} \subset \mathcal{M}$, $\mathcal{M}_{\mathrm{R}} \cup \mathcal{M}_{\mathrm{P}} = \mathcal{M}$ and $\mathcal{M}_{\mathrm{R}} \cap \mathcal{M}_{\mathrm{P}} = \emptyset$. Because the number of real measurements will be limited by the cost of installation and maintenance of sensing devices, and to initiate the state estimation algorithm, pseudo measurements will be added in order for the entire system to be "observable". The initial values of pseudo measurements are assigned based on the knowledge of real measurements,

and usually with large noise variances. It is worth noting that in the context of seismic hazards, there is no enough priori knowledge (e.g., historical data) that can be used for an appropriate initialization.

The probabilistic measurement model of hydraulic system state estimation is expressed as

$$\mathbf{z} = \mathbf{g}(\mathbf{x}) + \mathbf{u} \tag{5.1}$$

where the vector $\mathbf{x} = (x_1, ..., x_n)$ represents the probabilistic water system states; the vector $\mathbf{u} = (u_1, ..., u_k)$ where u_i is the additive measurement noise assumed to be independent Gaussian random variable with zero mean, i.e. $\mathbf{u} \sim \mathcal{N}(\mathbf{0}, \Sigma)$, Σ is a diagonal matrix with the i^{th} diagonal element σ_i^2; $\mathbf{z} = (z_1, ..., z_k)$ is the vector of measurement readings such as flow rate and hydraulic head; and $\mathbf{g} = (g_1(\mathbf{x}), ..., g_k(\mathbf{x}))$ is the vector of non-linear functions associated with each measurement following hydraulic physical laws. Each measurement $\mathcal{M}_i \in \mathcal{M}$ is associated with measured value z_i, measurement noise u_i, and measurement function $g_i(\mathbf{x})$.

The probabilistic state estimator aims to find an estimated $\hat{\mathbf{x}}$ of the true states $\mathbf{x}$ that achieves the maximum a posteriori probability (MAP), given the measurement set $\mathbf{z}$ and the priori state information of $\mathbf{x}$ according to the measurement model in (5.1). It is mathematically expressed as

$$\max_{\mathbf{x}} p(\mathbf{x}|\mathbf{z}) = \frac{p(\mathbf{x})p(\mathbf{z}|\mathbf{x})}{p(\mathbf{z})} \tag{5.2}$$

where $p(\cdot)$ represents the probability density function. Assuming that the prior probability distribution $p(\mathbf{x})$ is uniform, and given that the measurement probability distribution $p(\mathbf{z})$ does not depend on $\mathbf{x}$, MAP solution of (5.2) reduces to maximization of the likelihood

function $\mathcal{L}(\mathbf{z}|\mathbf{x})$, which is defined via likelihoods of k independent measurements:

$$\hat{\mathbf{x}} = \arg\max_{\mathbf{x}} \mathcal{L}(\mathbf{z}|\mathbf{x}) = \arg\max_{\mathbf{x}} \prod_{i=1}^{k} \mathcal{N}(z_i|\mathbf{x}, \sigma_i^2) \tag{5.3}$$

One can find the solution of (5.3) by weighted least squares (WLS) estimator [34]:

$$\hat{\mathbf{x}} = \arg\min_{\mathbf{x}} \sum_{i=1}^{k} \frac{(z_i - g_i(\mathbf{x}))^2}{\sigma_i^2} \tag{5.4}$$

To obtain the WLS estimate in (5.4), we need to first obtain a proper formulation for $\mathbf{g}(\mathbf{x})$ (Sec.5.2.3), and employ an efficient algorithm to conduct marginalization over $p(\mathbf{x}|\mathbf{z})$ in (5.2) with respect to $\mathbf{x}$ (Sec.5.3).

5.2.3 Hydraulic Network Model

The hydraulic system model is defined using the non-linear measurement functions $\mathbf{g}(\mathbf{x})$ that follow the physical laws to connect measured variables with state variables. This model takes hydraulic head denoted by $\mathbf{h}$ as state variables $\mathbf{x}$ (i.e., $\mathbf{x} \equiv \mathbf{h}$), since hydraulic head measurements are essential pieces of information that are required for determining water service availability. Hydraulic head represents the mechanical energy per unit weight of fluid in the system, and is defined on water node i as $h_i = p_i + e_i$, for $i \in \mathcal{V}$, where p_i is the pressure head and e_i is the elevation head at node i. The typical set of measurements $\mathcal{M}$ in water systems includes: the status of valves V_{ij} (open or closed) and flow rates Q_{ij} (cubic meter per second, cms or m^3/s)) at pipes $(i, j) \in \mathcal{E}$, and the hydraulic head h_i (meter, m) at special nodes $i \in \mathcal{V}$ (e.g., reservoir, pump and tank). That is $\mathcal{M} = \{\mathcal{M}_{V_{ij}}, \mathcal{M}_{Q_{ij}}, \mathcal{M}_{h_i}\}$ for $(i, j) \in \mathcal{E}$ and $i \in \mathcal{V}$, where $\{\mathcal{M}_{h_i}\}$ is referred to as the direct measurement $\mathcal{M}_{\text{dir}}$ since it measures state variables directly, and $\{\mathcal{M}_{V_{ij}}, \mathcal{M}_{Q_{ij}}\}$ is referred to as the indirect measurement $\mathcal{M}_{\text{ind}}$. Noted that real/pseudo measurements ($\mathcal{M}_{\text{R}}$ and $\mathcal{M}_{\text{P}}$) and direct/indirect measurements

($\mathcal{M}_{\text{dir}}$ and $\mathcal{M}_{\text{ind}}$) are just two different classifications of measurements that are defined for the convenience to present the proposed approach. For reasons of completeness, a short elaboration of the hydraulic background is given following. Reader may safely skip this part. The measurement functions used in the PDD model are specified based on the measurement types and readings.

For flow-rate measurement $z_{Q_{ij}}$:

$$g_{Q_{ij}}(\cdot) = (1/R_{ij})^{\frac{1}{1.852}} \cdot |h_{L_{ij}}|^{\frac{1}{1.852}} \qquad \text{if } z_{Q_{ij}} > 0.0004 \tag{5.5a}$$

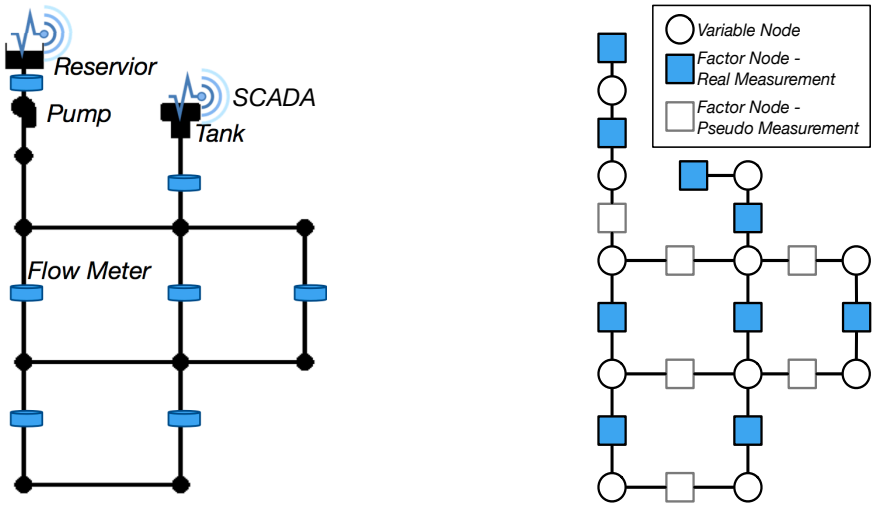
$$g_{Q_{ij}}(\cdot) = (1/(R_{ij} \cdot m)) \cdot h_{L_{ij}} \qquad \text{if } z_{Q_{ij}} < 0.0002 \tag{5.5b}$$

$$g_{Q_{ij}}(\cdot) = a'(h_{L_{ij}}/R_{ij})^3 + b'(h_{L_{ij}}/R_{ij})^2 + c'(h_{L_{ij}}/R_{ij}) + d' \quad \text{if } 0.0002 \leq z_{Q_{ij}} \leq 0.0004 \tag{5.5c}$$

For hydraulic head measurement z_{h_i}:

$$g_{h_i}(\cdot) = h_i \tag{5.6}$$

Here $h_{L_{ij}} = |h_i - h_j|$ is the headloss in the pipe (m), and $R_{ij} = 10.667C^{-1.852}d^{-4.871}L$ is the pipe resistance coefficient (unitless) [108] where C is the Hazen-Williams roughness coefficient (unitless), d is the pipe diameter (m) and L is the pipe length (m). Constant m=0.001 in (5.5b), and constants a'=$1.524 \cdot 10^{15}$, b'=$-2.530 \cdot 10^{9}$, c'=$1.830 \cdot 10^{3}$, d'=$-7.695 \cdot 10^{-5}$ in (5.5c), which are calculated using polynomial curve fitting. In (5.5), different functions are used according to the values of flow-rate measurements. Because when $Q_{ij} \approx 0$, it can cause the Jacobian of the set of hydraulic equations to become singular, and [68] proposed to split the domain of Q into several segments to create a piecewise smooth function.



(a) Hydraulic network (b) Factor graph

Figure 5.3: Transformation of (5.3a) the hydraulic network with measurement configuration into (5.3b) the corresponding factor graph with variable nodes and factor nodes.

5.2.4 Factor Graph Construction

To solve the optimization problem in (5.2), we instead need to find an optimal solution of (5.3) in an efficient manner. We first construct a factor graph to describe a factorization of the likelihood function $\mathcal{L}(\mathbf{z}|\mathbf{x})$. Factor graphs comprised of the set of variable nodes and factor nodes have been widely used to represent factorization of a probability distribution function, enabling efficient computations [82].

As shown in Fig.5.3a, a factor graph can be formed from the hydraulic model, where the variable node characterizes the probability distribution of the hydraulic head at nodes, and the factor node is determined by the set of measurements. The pseudo measurements will be filled in based on the real measurement readings, to make the entire system "observable". That is, the vector of state variables $\mathbf{h}$ defines the set of variable nodes $\mathcal{X} = \{h_1, ..., h_n\}$, while the set of measurements $\mathcal{M}$ defines the set of factor nodes $\mathcal{F} = \{f_1, ..., f_k\}$. A factor node f_i connects to a variable node $x_s \in \mathcal{X}$ if and only if the state variable h_s is an argument of the corresponding measurement function $g_i(\mathbf{x})$ according to (5.5) and (5.6). In this manner, hydraulic head and flow rate are modeled separately, and their correlations can be captured

in the corresponding factor nodes.

5.2.5 Leak Event Model

Leaks can cause large changes in network hydraulics, and we use WNTR to model the pipe breaks [68]. A new junction is added onto each pipeline to model the event node, and if a pipe breaks, its event node will be used as the leaky point. In WNTR, the mass flow rate of fluid through the hole, d^{leak}, is expressed as:

$$d^{\text{leak}} = C_d A \sqrt{2\rho} p^{\alpha} \tag{5.7}$$

where C_d is the discharge coefficient (unitless) with default value 0.75, A is the area of the hole (m^2), ρ is the density of the fluid (kg/m^3), p is the pressure (Pa) computed using elevation and hydraulic head (m), fluid density and acceleration due to gravity (m/s^2), and α is set to 0.5 for large leaks out of steel pipes.

5.3 A Multi-phase Probabilistic State Estimation

The industry paradigm is shifting from the traditionally deterministic model based centralized monitoring architecture to probabilistic model based highly distributed interactive data and resource management. Therefore, a multi-phase, distributed implementation of the state estimator is likely to be the preferred approach, which enables a fast and accurate hydraulic behavior assessment.

The belief propagation algorithm efficiently calculates the marginal distribution for each state variables by passing messages (a) from a variable node x_s to a factor node f_i and (b) from a factor node f_i to a variable node x_s. Variable and factor nodes locally process the incoming

messages and calculate outgoing messages in a distributed manner. Under the assumption that measurement errors $\{u_i\}$ follow a Gaussian distribution, the probability density function of $\{x_s\}$ and $\{f_i\}$ are Gaussian. The passing-message can then be characterized by mean and variance. The marginal inference provides marginal probability distributions $p(\mathbf{x}|\mathbf{z})$ that is used to find an estimate $\hat{\mathbf{x}}$ of the true states $\mathbf{x}$. It is well-known that the key assumption of the BP algorithm converging to the optimal solution is that the applying graph has no cycle, i.e., tree-structured [82]. Such assumption often does not hold for most water distribution networks, which are locally dense and contain loops. In addition, due to the non-linearity of the measurement functions, the BP based approach will be sequentially applied over the factor graph until the stop criterion is satisfied, which increases the time complexity of convergence. Our experience shows that the inference on a large-scale water systems can take more than 30min to converge, which is too slow for an on-line state estimation, especially under seismic events. Ideally, state estimation should run at the scanning rate or at least less than the sampling rate of industrial metering devices (15min) [15], to handle the new measurement as soon as it is delivered from telemetry to the computational unit. To overcome these limitations, we proposed a two-phase approach: (I) decomposes the hydraulic network into conditional independent connected components, and (II) performs the GN-BP based inference on each of them. It is worth noting that Phase II can be done in parallel for all components that can further reduce the time complexity.

5.3.1 Phase I: Network Decomposition by Articulation Points

Phase I aims to generate several disjoint connected components where each one has a moderate size and they are conditional independent given specific nodes being observed. Articulation points (APs) are vertices in an undirected connected graph, whose removal along with the removal of their incident links disconnects the graph. The APs can divide a graph into several biconnected components, where a biconnected component is a connected and

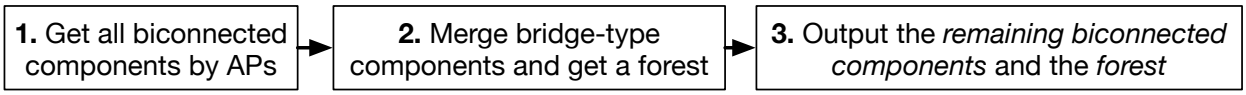


Figure 5.4: The work flow of Phase I - network decomposition.

"nonseparable" subgraph, meaning that if any one of vertices is removed, the subgraph will remain connected. Noted that a biconnected graph has no APs. In the context of the water system, it is important to know the relatively structural prominence of nodes or links to identify key elements in the network. A water node that is an articulation point often has higher information centrality and will be considered as the critical location - a single point whose failure would cause network disconnection [26]. A water network subzone that is a biconnected graph is considered as more resilient and less susceptible to damage and perturbation [145]. The existing of the alternative supply paths provides a two-fold redundancy and improve system robustness and resilience by avoiding critical locations and network bottlenecks. This redundancy, in turn, may improve the performance of state estimations, since the incorrect measurement on one path may be compensated by measurements from alternative paths, and an estimate on the variable node can be derived by the cooperation of messages from multiple incident links. The work flow of Phase I is shown in Fig.5.4, and summarized in following:

1. Find the articulation points and biconnected components of the water network. For example, in Fig.5.5a, this is the hydraulic network of the water system operated by North Marin Water District, and nodes that are labeled by stars are APs, which are used to generate all biconnected components.

2/3. One observation on branched water networks is that there are many bridge-type biconnected components, that consist of a single edge. A bridge, in graph theory, is defined as an edge whose deletion increases the number of connected components, and a bridge-type biconnected components is any biconnected components that consists of a single edge.

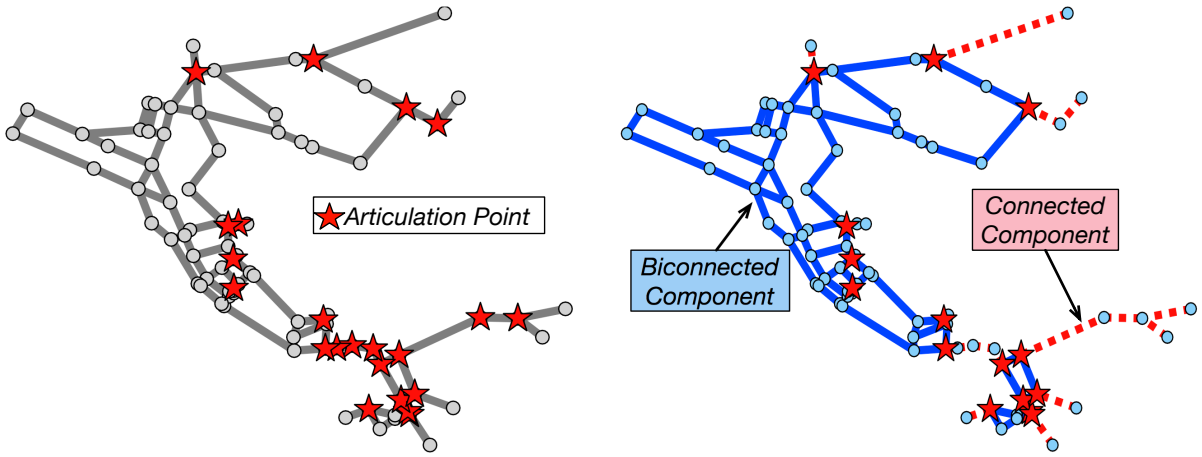
The purpose of the network decomposition is to reduce the time complexity while obtaining the optimal state estimates. This will require to use as many sensing devices as the number of split points to make disjoint components conditionally independent. Without using many sensors, we do not want to split the network into many small parts (components with a single edge). Thus, we merge those bridge-type biconnected components and obtain a disjoint union of trees, which can be proved by contradiction as shown in Theorem 5.1. For example, in Fig.5.5, the network decomposition outputs 3 biconnected components that are not bridge-type, and 9 (bi)connected components after merging.

Lemma 5.1. *For any graph G, all edges in a cycle will be in the same biconnected component.*

Theorem 5.1. *The subgraph G_{sub} generated by merging bridge-type biconnected components of a specific graph G is a forest, i.e. a disjoint union of trees.*

Proof. To prove it by contradiction, assume that there is a cycle on the subgraph G_{sub}. Since $G_{sub} \subseteq G$, it means that this cycle is also in G. According to Lemma 5.1, all edges along with this cycle will be in the same biconnected component, which contradicts the initial statement that they belong to different biconnected components. Therefore the assumption must be false. □

Without losing the structural information after decomposition, the node that is the intersection of 2 or more generated disjoint connected components needs to be observable. In Fig.5.5b, nodes labeled by stars are intersection points and will be observed. That is $\mathcal{M}_{h_i} \in \mathcal{M}_{\mathrm{R}}$ if node i is such a point. The paper focuses on efficient state estimation by utilizing limited and noisy measurements. The problem to find a minimum amount of measurements that are required to deliver optimal system states is out of the scope. The corresponding study is referred to as observability analysis.



(a) Novato water network (b) Network decomposition

Figure 5.5: Decomposition of (a) Novato water network into (b) 3 biconnected components (solid lines) and 9 (bi)connected components (dashed lines).

5.3.2 Phase II: Hydraulic State Inference

In Phase II, we describe an efficient Gauss-Newton Belief Propagation (GN-BP) based hydraulic state estimation algorithm, which converges to the optimal inference results for all water nodes in a reasonable time. To speed up the convergence time on loopy networks, we then introduce a feedback vertex set (FVS) selection criterion to break all loops, and a modified version of the proposed algorithm to use FVS.

GN-BP based on-line inference

The BP based algorithm allows the state of end-users to be estimated in a distributed, message-passing manner with the neighboring end-users where flow meters are located. Then the aggregated information is communicated in a bottom-up way to the backbone system operator for driving system control. However, due to the non-linearity of measurement functions $\mathbf{g}(\cdot)$ in (5.5), the closed-form expressions for certain classes of BP messages cannot be obtained. Therefore we integrate Gauss-Newton (GN) method with BP based inference to solve the WLS problem in (5.4). The GN algorithm is used to solve non-linear least squares

problem by minimizing a sum of squared function values and it has advantage that second derivatives, which can be challenging to compute, are not required.

Based on k number of measurements $\mathcal{M}$, the solution of (5.4), which is a vector of n state variables $\hat{\mathbf{x}} \equiv \hat{\mathbf{h}}$, can be found using the GN method [34]:

$$[\mathbf{J}(\mathbf{x}^{(\nu)})^T \mathbf{W} \mathbf{J}(\mathbf{x}^{(\nu)})] \cdot \Delta \mathbf{x}^{(\nu)} = \mathbf{J}(\mathbf{x}^{(\nu)})^T \mathbf{W} \mathbf{r}(\mathbf{x}^{(\nu)}) \tag{5.8a}$$

$$\mathbf{x}^{(\nu+1)} = \mathbf{x}^{(\nu)} + \Delta \mathbf{x}^{(\nu)} \tag{5.8b}$$

where $\nu = \{1, 2, 3, ...\}$ is the iteration index, and at each iteration step ν, $\Delta \mathbf{x}^{(\nu)} \in \mathbb{R}^n$ is the vector of increments of state variables $\mathbf{x}$, $\mathbf{J}(\mathbf{x}^{(\nu)}) \in \mathbb{R}^{k \times n}$ is the Jacobian matrix of measurement functions $\mathbf{g}(\mathbf{x}^{(\nu)})$, $\mathbf{W} \in \mathbb{R}^{k \times k}$ is a diagonal matrix containing inverses of measurement variances, i.e. $\mathbf{W} = \Sigma^{-1}$, and $\mathbf{r}(\mathbf{x}^{(\nu)}) = \mathbf{z} - \mathbf{g}(\mathbf{x}^{(\nu)})$ is the vector of residuals, i.e., the difference between measured and estimated values. The Jacobian expressions corresponding to $g_{Q_{ij}}(\cdot)$ and $g_{h_i}(\cdot)$ can be computed based on (5.5) and (5.6). The Jacobian expressions corresponding to $g_{Q_{ij}}(\cdot)$ are:
If $z_{Q_{ij}} > 0.0004$, based on (5.5a):

$$\frac{\partial g_{Q_{ij}}(\cdot)}{\partial h_i} = (\frac{1}{1.852}) \cdot (1/R_{ij})^{\frac{1}{1.852}} \cdot |h_{L_{ij}}|^{\frac{1}{1.852}-1} \tag{5.9a}$$

$$\frac{\partial g_{Q_{ij}}(\cdot)}{\partial h_j} = -(\frac{1}{1.852}) \cdot (1/R_{ij})^{\frac{1}{1.852}} \cdot |h_{L_{ij}}|^{\frac{1}{1.852}-1} \tag{5.9b}$$

If $z_{Q_{ij}} < 0.0002$, based on (5.5b):

$$\frac{\partial g_{Q_{ij}}(\cdot)}{\partial h_i} = 1/(R_{ij} \cdot m) \tag{5.10a}$$

$$\frac{\partial g_{Q_{ij}}(\cdot)}{\partial h_j} = -1/(R_{ij} \cdot m) \tag{5.10b}$$

If $0.0002 \leq z_{Q_{ij}} \leq 0.0004$, based on (5.5c):

$$\frac{\partial g_{Q_{ij}}(\cdot)}{\partial h_i} = 3a'(h_{L_{ij}}^2/R_{ij}^3) + 2b'(h_{L_{ij}}/R_{ij}^2) + c'/R_{ij} \tag{5.11a}$$

$$\frac{\partial g_{Q_{ij}}(\cdot)}{\partial h_j} = -3a'(h_{L_{ij}}^2/R_{ij}^3) - 2b'(h_{L_{ij}}/R_{ij}^2) - c'/R_{ij} \tag{5.11b}$$

The Jacobian expression corresponding to $g_{h_i}(\cdot)$, based on (5.6), is:

$$\frac{\partial g_{h_i}(\cdot)}{\partial h_i} = 1 \tag{5.12}$$

Consider the GN method in (5.8) where, at each iteration ν, the algorithm returns a new estimate $\hat{\mathbf{x}}$, and (5.8a) represents the minimization problem:

$$\min_{\Delta \mathbf{x}^{(\nu)}} ||\mathbf{W}^{1/2}[\mathbf{r}(\mathbf{x}^{(\nu)}) - \mathbf{J}(\mathbf{x}^{(\nu)})\Delta \mathbf{x}^{(\nu)}]||_2^2 \tag{5.13}$$

Hence, the probability measurement model (5.1) can be re-defined as a group of linear equations:

$$\mathbf{r}(\mathbf{x}^{(\nu)}) = \phi(\Delta \mathbf{x}^{(\nu)}) + \mathbf{u} \tag{5.14}$$

where $\phi(\Delta \mathbf{x}^{(\nu)}) = \mathbf{J}(\mathbf{x}^{(\nu)})\Delta \mathbf{x}^{(\nu)}$ comprises linear functions. The MAP solution of (5.2) can be reduced to maximum likelihood problem, and the equation (5.3) can be re-defined as an

iterative optimization problem:

$$\begin{aligned}\Delta\hat{\mathbf{x}}^{(\nu)} &= \arg\max_{\Delta\mathbf{x}^{(\nu)}} \mathcal{L}(\mathbf{r}(\mathbf{x}^{(\nu)})|\Delta\mathbf{x}^{(\nu)}) \\ &= \arg\max_{\Delta\mathbf{x}^{(\nu)}} \prod_{i=1}^{k} \mathcal{N}(r_i(\mathbf{x}^{(\nu)})|\Delta\mathbf{x}^{(\nu)}, \sigma_i^2) \qquad (5.15a)\end{aligned}$$

$$\mathbf{x}^{(\nu+1)} = \mathbf{x}^{(\nu)} + \Delta\hat{\mathbf{x}}^{(\nu)} \qquad (5.15b)$$

Next, we show that the solution of (5.15) can be efficiently obtained using BP based algorithm applied over the underlying factor graph introduced in Sec.5.2.4. The factor graph constructed by the factorization of the likelihood function in (5.15a) is slightly different from the one in (5.3). The set of variables nodes is defined as the increments of state variables instead of the state variable itself, i.e. $\mathcal{X} = \{\Delta h_1, ..., \Delta h_n\}$, while the set of factor nodes is defined as before based on the measurements $\mathcal{M}$, i.e. $\mathcal{F} = \{f_1, ..., f_k\}$. The factor node f_i connects to a variable node Δx_s if and only if Δx_s is an argument of the corresponding function $\phi_i(\Delta\mathbf{x})$, that is if the state variable h_s is an argument of the measurement function $g_i(\mathbf{x})$. The BP algorithm on factor graphs proceeds by passing two types of messages along the edges: from variable nodes to factor nodes and from factor nodes to variable nodes. BP messages represent "beliefs" about variable nodes, thus a message that arrives or departs a variable node is a probability distribution of the random variable associated with this node. The "beliefs" will be iteratively updated by incoming messages and propagated by outgoing messages until the stopping criterion is satisfied.

Message from a variable node to a factor node: Since we consider the Gaussian graphical model, the message from a variable node Δx_s to a factor node f_i at iteration step

τ can be characterized by mean $r^{(\tau)}_{\Delta x_s \to f_i}$ and variance $\sigma^{2\ (\tau)}_{\Delta x_s \to f_i}$:

$$r^{(\tau)}_{\Delta x_s \to f_i} = \left(\sum_{f_a \in \mathcal{F}_s \backslash f_i} \frac{r^{(\tau-1)}_{f_a \to \Delta x_s}}{\sigma^{2\ (\tau-1)}_{f_a \to \Delta x_s}} \right) \cdot \sigma^{2\ (\tau)}_{\Delta x_s \to f_i} \tag{5.16a}$$

$$\frac{1}{\sigma^{2\ (\tau)}_{\Delta x_s \to f_i}} = \sum_{f_a \in \mathcal{F}_s \backslash f_i} \frac{1}{\sigma^{2\ (\tau-1)}_{f_a \to \Delta x_s}} \tag{5.16b}$$

where $\mathcal{F}_s$ is a set of factor nodes incident to Δx_s, and $\mathcal{F}_s \backslash f_i$ is a subset by excluding the factor node f_i. The incoming messages used for calculation are obtained in previous iteration $(\tau - 1)$.

Message from a factor node to a variable node: Similarly, the message from a factor node f_i to a variable node Δx_s can be characterized by mean $r_{f_i \to \Delta x_s}$ and variance $\sigma^2_{f_i \to \Delta x_s}$:

$$r^{(\tau)}_{f_i \to \Delta x_s} = \frac{1}{C_{i,\Delta x_s}} \left(r_i - \sum_{\Delta x_b \in \mathcal{X}_i \backslash \Delta x_s} C_{i,\Delta x_b} \cdot r^{(\tau)}_{\Delta x_b \to f_i} \right) \tag{5.17a}$$

$$\sigma^{2\ (\tau)}_{f_i \to \Delta x_s} = \frac{1}{C^2_{i,\Delta x_s}} \left(\sigma^2_i + \sum_{\Delta x_b \in \mathcal{X}_i \backslash \Delta x_s} C^2_{i,\Delta x_b} \cdot \sigma^{2\ (\tau)}_{\Delta x_b \to f_i} \right) \tag{5.17b}$$

where $\mathcal{X}_i$ is a set of variable nodes incident to f_i, and $\mathcal{X}_i \backslash \Delta x_s$ is a subset by excluding the variable node Δx_s. $C_{i,\Delta x_p}$ for $\Delta x_p \in \mathcal{X}_i$ are Jacobian elements of the measurement function $g_i(\cdot)$ associated with f_i as shown in (5.9, 5.10, 5.11, 5.12):

$$C_{i,\Delta x_p} = \frac{\partial g_i(\cdot)}{\partial x_p} \tag{5.18}$$

Marginal inference: The marginal of the state variable is the estimated value of the

increment, which will be calculated when $r_{f_i \to \Delta x_x}$ and $\sigma^2_{f_i \to \Delta x_x}$ converge:

$$\Delta \hat{x}_s = \left(\sum_{f_i \in \mathcal{F}_s} \frac{r_{f_i \to \Delta x_x}}{\sigma^2_{f_i \to \Delta x_x}} \right) \cdot \left(1 \Big/ \sum_{f_i \in \mathcal{F}_s} \frac{1}{\sigma^2_{f_i \to \Delta x_x}} \right) \tag{5.19}$$

The GN-BP based inference subroutine is summarized in Algorithm 3. To present the algorithm precisely, we define different types of factor nodes based on the measurements $\mathcal{M}$. The factor nodes that correspond to real measurements $\mathcal{M}_{\mathrm{R}}$ are real factor nodes $\mathcal{F}_{\mathrm{R}} \subseteq \mathcal{F}$, and similarly pseudo factor nodes $\mathcal{F}_{\mathrm{P}} \subset \mathcal{F}$ are associated with pseudo measurements $\mathcal{M}_{\mathrm{P}}$. The direct/indirect measurements $\mathcal{M}_{\mathrm{dir}}$ and $\mathcal{M}_{\mathrm{ind}}$ are represented by direct/indirect factor nodes $\mathcal{F}_{\mathrm{dir}} \subseteq \mathcal{F}$ and $\mathcal{F}_{\mathrm{ind}} \subseteq \mathcal{F}$ respectively. In Algorithm 3, the outer loop stops when the difference on estimated values is less than a very small number ϵ_O, and the inner loop stops when the difference on BP messages is less than a very small number $\epsilon_I(\nu)$ that varies with the outer iterations.

Feedback vertex set selection

The non-bridge biconnected components generated by Phase I contain loops - it provides path redundancy for resilience but can also add difficulty for BP inference. We consider a particular set of nodes called a feedback vertex set (FVS) denoted by $\mathcal{F}$ whose removal breaks all the cycles and results in a cycle-free graph, which is inspired by [79]. The algorithm proposed in [79] runs in time $\mathcal{O}(m^2 n)$ where m is the number of feedback nodes and n is the total number of nodes. When m is bounded by a small number, this is a significant reduction from $\mathcal{O}(n^3)$ of LBP. [79] further proposed a pseudo-FVS selection process with bounded size and this procedure works well for grid graphs with fast convergence and high accuracy. The problem of finding the minimum FVS for general graphs is still an active research area, and the fastest algorithm for finding the minimum FVS runs in time $\mathcal{O}(1.7548^n)$ where n is number number of nodes [48].

Algorithm 3 The distributed GN-BP based inference

Input factor graph $G_f(\mathcal{X}, \mathcal{F})$, state variables $\mathbf{x}$ with initial values, state threshold $[x_L, x_H]$, measured values $\mathbf{z}$, outer loop stopping threshold ϵ_O, and a vector of inner loop stopping thresholds ϵ_I

Output estimated states $\hat{\mathbf{x}}$

/* Outer state update loop $\nu = 1, \ldots;\ \tau = 0$ */

while $|\mathbf{x}^{(\nu)} - \mathbf{x}^{(\nu-1)}| < \epsilon_o$ **do**

  for $f_i \in \mathcal{F}_{\text{ind}}$ **do**

    $r_i^{(\nu)} = z_i - g_i(\mathbf{x}^{(\nu)})$ using (5.5)

    $C_{i,\Delta x_p}^{(\nu)} = \frac{\partial g_i(\cdot)}{\partial x_p}$, $\Delta x_p \in \mathcal{X}_i$ using (5.9, 5.10, 5.11, 5.12)

  end for

  for $f_i \in \mathcal{F}_{\text{dir}}$ **do**

    $r_i^{(\nu)} = z_i - g_i(\mathbf{x}^{(\nu)})$ using (5.6)

  end for

  for $\Delta x_s \in \mathcal{X}$ **do**

    if $f_{x_s} \in \mathcal{F}_{\text{R}}$ **then**

      $r_{\Delta x_s \to f_i}^{(\nu,\tau=0)} = r_s^{(\nu)}$; $\sigma_{\Delta x_s \to f_i}^{(\nu,\tau=0)} = \epsilon_\sigma$ for $f_i \in \mathcal{F}_s$

    else

      $r_{\Delta x_s \to f_i}^{(\nu,\tau=0)} = \epsilon_r$; $\sigma_{\Delta x_s \to f_i}^{(\nu,\tau=0)} = \infty$ for $f_i \in \mathcal{F}_s$

    end if

  end for

/* Inner message update loop $\tau = 1, \ldots$ */

  while $|r_{f \to \Delta x}^{(\tau)} - r_{f \to \Delta x}^{(\tau-1)}| < \epsilon_I(\nu)$ **do**

    for $f_i \in \mathcal{F}_{\text{ind}}$ **do**

      $r_{f_i \to \Delta x_s}^{(\tau)}$, $\sigma^{2\ (\tau)}_{f_i \to \Delta x_s}$ using (5.17) for $\Delta x_s \in \mathcal{X}_i$

    end for

    for $\Delta x_s \in \mathcal{X}$ **do**

      $r_{\Delta x_s \to f_i}^{(\tau)}$, $\sigma^{2\ (\tau)}_{\Delta x_s \to f_i}$ using (5.16) for $f_i \in \mathcal{F}_s$

    end for

  end while

/* Marginal inference */

  for $\Delta x_s \in \mathcal{X}$ **do**

    $\Delta x_s^{(\nu)}$ using (5.19); $x_s^{(\nu+1)} = x_s^{(\nu)} + \Delta x_s^{(\nu)}$

  end for

/* State validation */

  for $\Delta x_s \in \mathcal{X}$ **do**

    if $x_s^{(\nu+1)} \notin [x_L, x_H]$ **then** $x_s^{(\nu+1)} = \bar{\mathbf{x}}^{(\nu+1)}$

    end if

  end for

end while

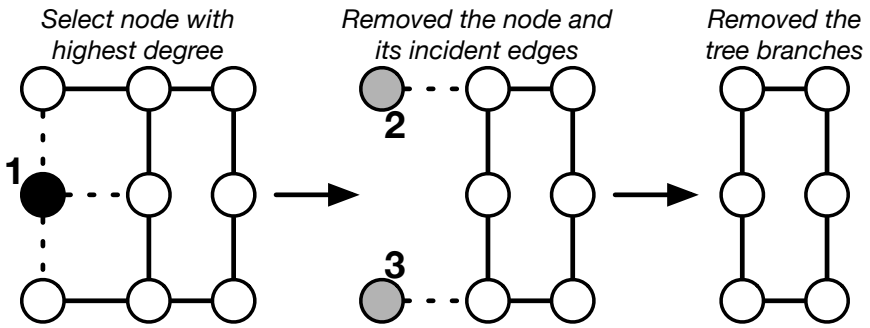


Figure 5.6: Example of 1^{st} iteration of the FVS selection criterion.

Many of water infrastructures in the U.S. have a hybrid network topology (a combination of loops and branches), and thus it is possible to find a FVS with a reasonable size to remove all loops in a water network [57]. Without losing much structural information in terms of removing nodes, the goal is to find a minimum FVS to break all cycles and enable a fast LBP convergence on the remaining graph. After Phase I, all non-cycle-free graphs are biconnected, meaning that all nodes in the graph are part of a cycle. To find an optimal FVS with a small size, we propose a greedy heuristic algorithm: one feedback node is chosen at each iteration, and at each stage we examine the graph excluding the nodes already included in the FVS $\mathcal{F}$ and select the node with the largest degree. We then remove the node along with its incident edges and put it into $\mathcal{F}$. The same procedure will be continued on the remaining graph $\mathcal{T}$ until it is empty. The motivation for this method is given that since the number of cycles is reduced with the removal of nodes, it makes sense to choose nodes with the highest degrees to remove more cycles at each iteration. When multiple nodes have same, highest degree, we randomly select one of them. As an example shown in Fig.5.6, node 1, one of the nodes with the highest degree, is selected and put into $\mathcal{F}$. The removal of node 1 and its incident edges leave $\mathcal{T}$ with 2 tree branches that are further removed from $\mathcal{T}$. This selection procedure continues until $\mathcal{T}$ is empty. The complexity of this algorithm is $\mathcal{O}(km)$, where k is the size of $\mathcal{F}$ and m is the number of edges in given graph G. Compared with $\mathcal{O}(n^3)$ of LBP, constructing a FVS in this manner significantly reduces the computational complexity. The selection algorithm is summarized in Algorithm 4.

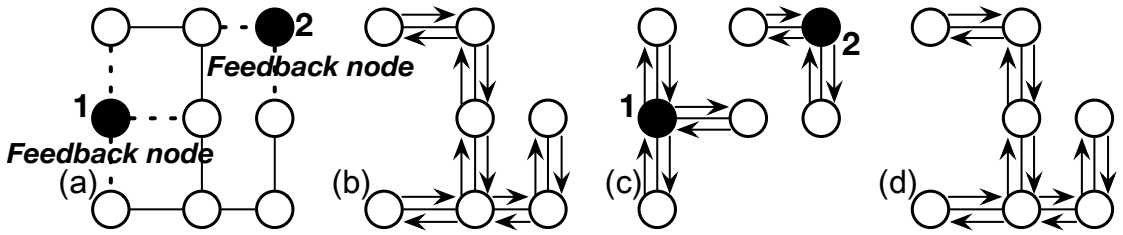


Figure 5.7: The message update scheme with 2 feedback nodes. The nodes here represent variable as well as factor nodes.

To utilize FVS, we use a special update scheme for feedback nodes. Consider the loopy graph in Fig.5.7a, FVS identified by Algorithm 4 contains two feedback nodes. (1) Algorithm 3 is first applied on the cycle-free graph G_T by removing feedback nodes and its incident edges (Fig.5.7b). We obtain inaccurate "partial states" for the nodes in the cycle-free graph. (2) We then compute the inference results for the feedback nodes by applying Algorithm 3 on the subgraph of feedback nodes, their incident edges and neighbors G_T (Fig.5.7c). (3) Last, we make corrections to the "partial states" of the non-feedback nodes by running Algorithm 3 on the cycle-free graph again (Fig.5.7d). Optimal inference results are obtained for all nodes. Noted that in (2) and (3), the initial states of nodes that are neighbors of feedback nodes (e.g., nodes appear both in Fig.5.7c and 5.7d) are determined by the previous stage.

Algorithm 4 The FVS selection criterion

Input biconnected component G generated by Phase I
Output a FVS $\mathcal{F}$
Objective find an optimal $\mathcal{F}$ to break all cycles in G

Let $\mathcal{F} = \emptyset$ and $\mathcal{T} = G$
while $\mathcal{T}$ is not empty **do**
    (a) Get node degrees of $\mathcal{T}$
    (b) Put the node with the highest degree into $\mathcal{F}$ and
        remove it with its incident edges from $\mathcal{T}$
    (c) Clean up $\mathcal{T}$ by eliminating all tree branches.
end while

Hydraulic state estimation

Given a water network, Phase I first splits it into several disjoint connected components, whose hydraulic states are then estimated by Phase II. To execute the GN-BP based inference on a graph $G(V, E)$, the state variables on nodes V need to be initialized and the pseudo indirect measurements on edges E need to be set. The initial values of observed state variables equal to their real measured values, while for those that are not observed, their initial values are set to the average of the observed states. The value of a pseudo indirect measurement is set to the same value as its closest real indirect measured value. The state estimator is summarized in Algorithm 5.

Algorithm 5 The multi-phase hydraulic state estimation

Input water network $\mathcal{G}$, measurements $\mathcal{M}$
Output estimated hydraulic heads of water nodes

/* After Phase I */
for each connected component $G_c(V, E)$ of $\mathcal{G}$ **do**
/* Initialization */
    For $\forall i \in V$: if $\mathcal{M}_i \in \mathcal{M}_{\mathrm{R}}$, $x_i = z_i$; else $x_i =$ average value of $\mathcal{M}_i \in \mathcal{M}_{\mathrm{dir}} \cap \mathcal{M}_{\mathrm{R}}$.
    For $\forall j \in E$: if $\mathcal{M}_j \in \mathcal{M}_{\mathrm{P}}$, $z_j = z_c$, where $c = \arg\min_l \mathrm{distance}(j, l)$ and $\mathcal{M}_l \in \mathcal{M}_{\mathrm{ind}} \cap \mathcal{M}_{\mathrm{R}}$.
/* Inference */
    if Use FVS **then**
        Find FVS of G_c, and get G_T and G_F.
        Alg.3 on factor graphs $G_f(\mathcal{X}, \mathcal{F})$ of G_T and G_F.
    else
        Alg.3 on factor graph $G_f(\mathcal{X}, \mathcal{F})$ of G_c.
    end if
end for

5.4 Experimental Study

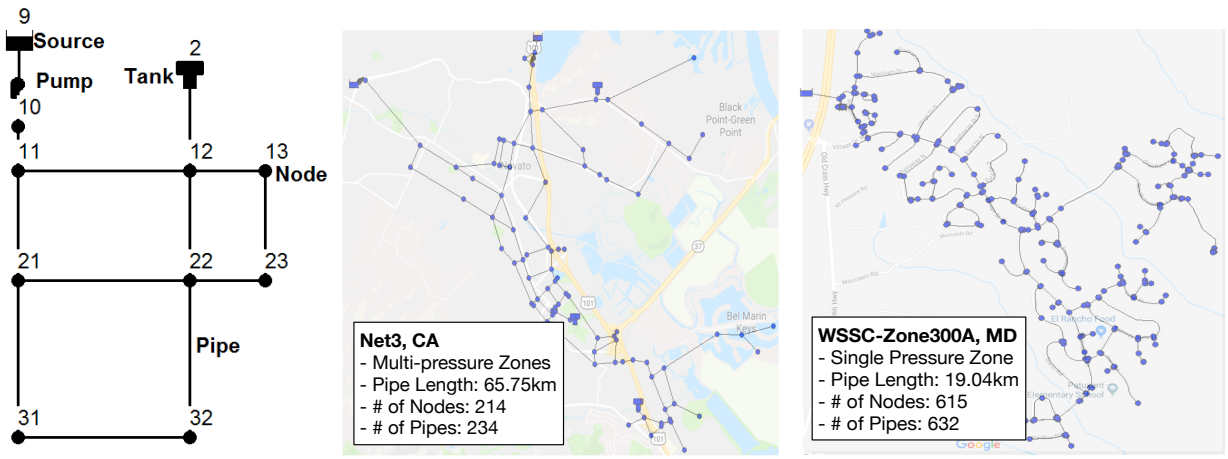
In this section, we examine the effectiveness of the proposed multi-phase state estimation algorithm on a small-scale canonical water network and two real-world water systems where

they have different configurations on topology, network size and pressure range (Fig.5.8). The inference approaches we compared include: GN-BP - directly estimate the states of the entire network by Phase II; FVS+GN-BP - find FVS and estimate the states using FVS by Phase II; Decomp+GN-BP - split the network by Phase I and estimate the states of each component by Phase II; Decomp+FVS+GN-BP - find FVS for each components generated by Phase I and estimate the states using FVS by Phase II. We begin by describing the setup under which the experiments are conducted, and introduce the performance metrics and the results.

5.4.1 Experimental Setup

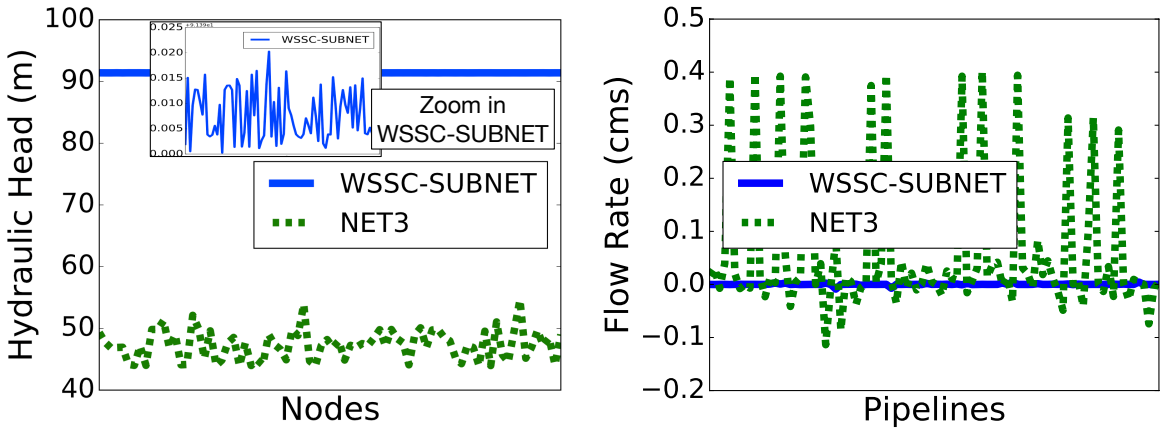
Figures 5.8b/5.8c show the real-world water systems that are used to evaluate the scalability of the proposed approach. NET3 is the service area, containing multiple pressure zones, of North Marin Water District (NMWD) provided by EPA, and WSSC-SUBNET is a single pressure zone of WSSC service area provided by WSSC. Figure 5.9 illustrates that NET3 has relatively large variances on both hydraulic heads and flow rates, where the variance on hydraulic heads is 5.989m compared with $1.386 \cdot 10^{-4}$m of WSSC-SUBNET, and the variance on flow rates is $1.5 \cdot 10^{-2}\mathrm{m}^3/\mathrm{s}$ compared with $2.755 \cdot 10^{-6}\mathrm{m}^3/\mathrm{s}$ of WSSC-SUBNET.

We use WNTR to simulate the earthquake impacts on the water distribution system by generating an earthquake event with magnitude 5.5 (unitless) and shallow depth 5000m at a random location. The pipe failure probabilities are then calculated using the attenuation model of peak ground acceleration (PGA) where $\mathrm{PGA} = 403.8 \times 10^{0.265\mathrm{M}}(\mathrm{R}+30)^{-1.218}$ and the fragility curve that defines the probability of exceeding a damage state as a function of PGA (Fig.5.10). Here M is the earthquake magnitude and R is the distance to epicenter (km). The leak diameter of the broken pipe is generated following the uniform distribution between 0.15 and 0.3 of the pipe diameter. Each pipeline may have different lengths and



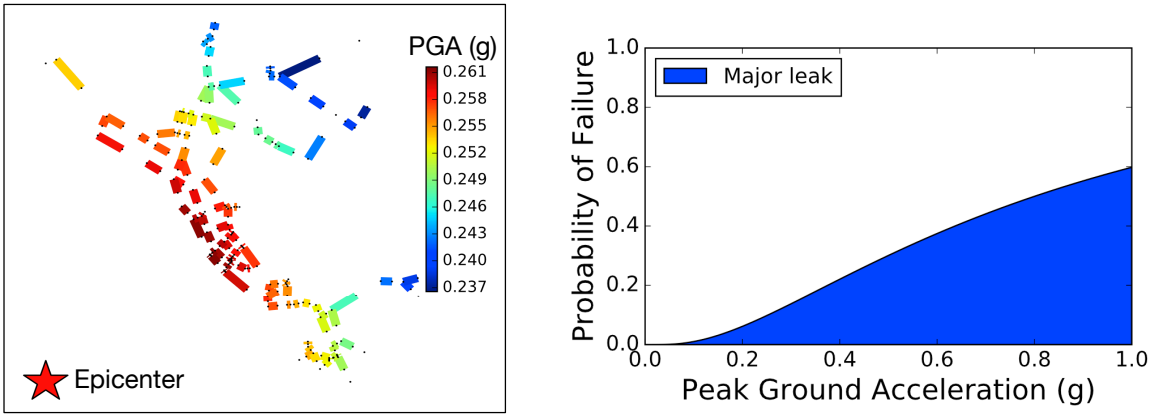
(a) Sample Net (b) NET3 (c) WSSC-SUBNET

Figure 5.8: (5.8a) A sample network provided by EPA and two real-world water systems: (5.8b) the water distribution network operated by NMWD and (5.8c) a single pressure zone operated by WSSC.



(a) Hydraulic Head (b) Flow Rate

Figure 5.9: Comparisons on (5.9a) hydraulic head (m) and (5.9b) flow rate (m^3/s) between WSSC-SUBNET and NET3 water systems at 11am under normal condition.



(a) Peak Ground Acceleration

(b) Fragility Curve

Figure 5.10: (a) PGA of pipelines for a magnitude of 5.5 earthquake. (b) Fragility curve for pipe damage.

diameters. We assume that 50% pipelines are instrumented by flow meters with varied noise variances, and the critical points are instrumented by SCADA monitoring systems with a small noise variance, since, in reality, SCADA systems are less susceptible to physical and cyber failures. The critical points include reservoirs, tanks, pumps, and those articulation points that are used to split the network in Phase I. There are 19 ($\approx$ 8%) and 48 ($\approx$ 8%) critical nodes in NET3 and WSSC-SUBNET respectively.

5.4.2 Performance Metrics

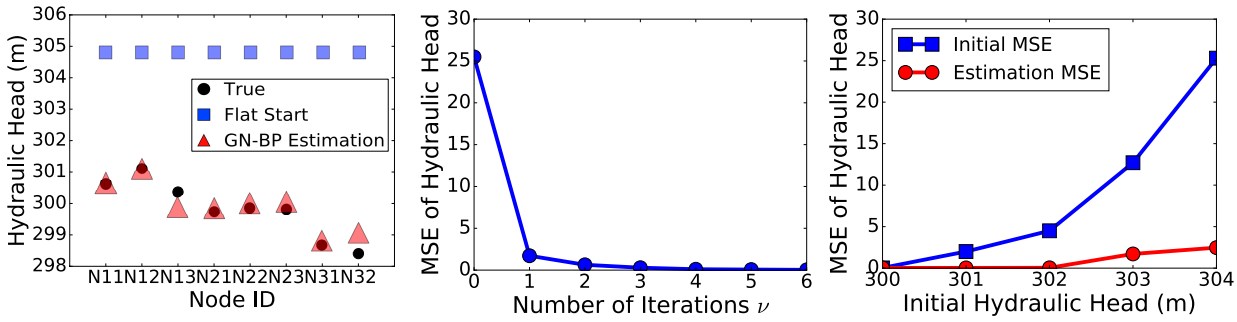
The effectiveness of the hydraulic state estimator is first evaluated in terms of the **time complexity** and the **accuracy**. The sampling rate of industrial flow meters is 15min, meaning that the estimator needs to infer the current system states in less than 15min before new measurements arrive. The faster the estimation converges to the optimal values, the quickly the countermeasures can be adopted. The time complexity is evaluated by the number of iterations to converge, and the accuracy is evaluated by the mean square error (MSE). We also consider two resilience metrics: **pipe damage state** and **water service availability**. There are two damage states: "no damage" and "break", and the goal is to

identify faulty zones where pipes are in the "break" state. To demonstrate this identification performance, we define True Positive (TP) as the number of predicted broken pipes within the distance threshold to the leaky points divided by the number of true broken pipes, and False Positive (FP) as the number of predicted broken pipes not in the distance threshold divided by the number of predicted broken pipes. A higher TP with a lower FP means a better performance. Water service availability at each node is computed as $V_i/\hat{V}_i$ for $i \in \mathcal{V}$, where V_i the actual water volume (m^3) received at node i and $\hat{V}_i$ is the expected water volume (m^3) received at node i. The water service availability can be influenced by failures and operational changes after a disaster, and it is important to estimate the received volume at end-users to localize the areas where they loss the access to the supply and facility. Precision and recall are used to demonstrate this estimation performance.

5.4.3 Complexity and Accuracy of Hydraulic State Estimation

In this section, the proposed state estimation approach is validated through a detailed simulation study on a small-scale canonical water network and two real-world water systems. We first demonstrate the performance of the approach on the small-scale water network (Fig. 5.8a), where the flow rate measurements have a large noise variance ($5m^3/s$). Figures 5.11a/5.11b illustrate that our approach can generate an accurate estimation on hydraulic heads (MSE = 0.02) with a flat start of 305m in a fast manner (converge in 1s). The reason for simulating a flat start is to mimic the water system environment with strong state variation, making typical initial guess method (from the last static state estimation) less informative for the new estimation process. Figure 5.11c shows the fact that the proposed GN-BP based estimation approach does not rely heavily on the initial guess.

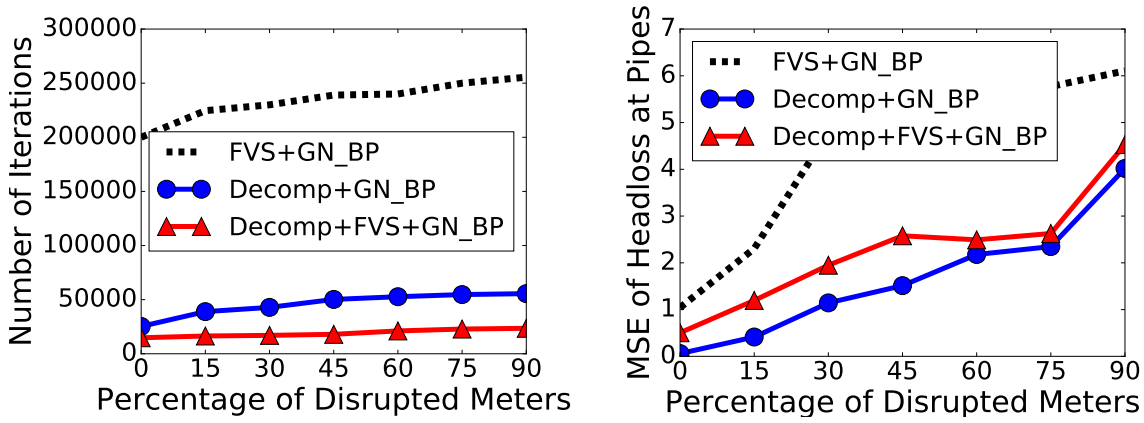
In the following set of simulations, we study the performance under different levels of sensing and infrastructure disruptions, with respect to the number of iterations to converge and the



(a) Hydraulic Head (b) MSE (c) MSE vs Initial

Figure 5.11: Sample Network - (a) estimated hydraulic heads with a flat start 305m and (b) mean square error versus number of outer iterations. (c) Errors versus initial head values.

mean square error of the headloss at pipes. Headloss is the absolute difference on hydraulic heads at the end nodes of a pipeline (5.5), and it can be used to identify pipe failures, since the headloss of a broken pipeline will increase due to the leaking. An earthquake event is generated on NET3, and it causes 10% ($\approx$ 20) pipe failures with varied leak volumes. Different percentages of the disrupted sensing devices are simulated based on the locations of broken pipelines, where those disrupted meters are considered with a large noise variance $2.5 \cdot 10^{-3}\text{m}^3/\text{s}$ compared with the noise variance $10^{-6}\text{m}^3/\text{s}$ of non-disruptive meters. Figures 5.12a/5.12b illustrate that compared with FVS+GN-BP, the proposed two-phase approach (Decomp+) can dramatically reduce the time complexity using less number of iterations and improve the accuracy with lower errors. Because the network decomposition separates the network into several subnets, and each of them has a relatively small size. In addition, each subnet can yield a better initialization by capturing its local information compared with the initialization over the entire network, since the initial states of unobserved state variables are set to the average of the directly observed values in its connected component. The performance of GN-BP (without FVS and Decomp) on NET3 is not shown, because it takes more than 30min to converge and makes the estimation too long to be meaningful. The approach of Decomp+FVS+GN-BP can further decrease the computational complexity but with the cost of less accuracy. Because the removal of feedback nodes breaks the original graph structure, which can result in the loss of topological information. Figures 5.13a/5.13b



(a) Number of Iterations

(b) Mean Square Error

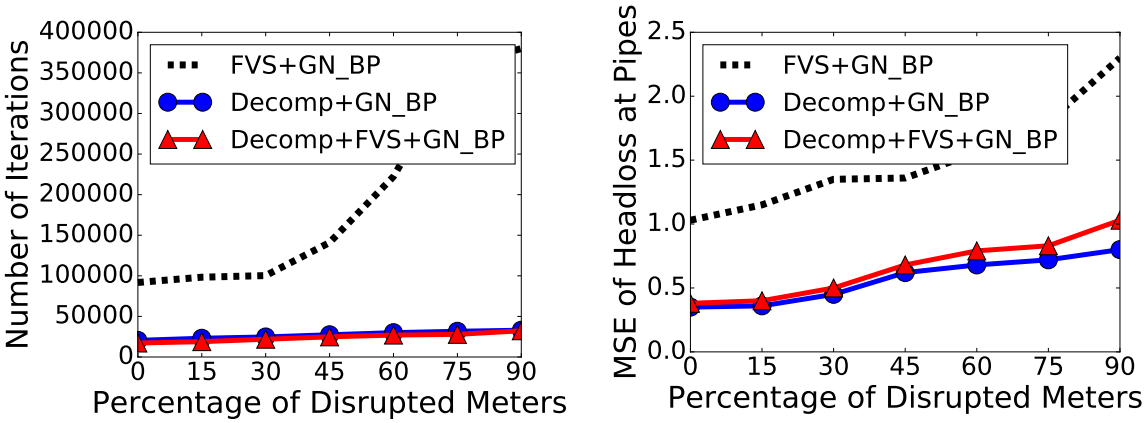
Figure 5.12: NET3 - (5.12a) total number of iterations to converge and (5.12b) mean square error of headloss at pipes versus the percentage of sensor failures.

show the estimation results on WSSC-SUBNET where there are 5% ($\approx$ 30) pipe failures, and the noise variances of disrupted and non-disruptive meters are $10^{-4}\text{m}^3/\text{s}$ and $10^{-8}\text{m}^3/\text{s}$ respectively. Likewise, the two-phase approach yields a better performance with less time complexity and high accuracy.

In Fig.5.14, the comparison is performed on NET3 under different levels of infrastructure (pipe) failures, where the noise variance of meters is $10^{-6}\text{m}^3/\text{s}$. It can be seen from Fig.5.14a, that as the number of pipe failures increasing, our approach is able to converge in approximately the same number of iterations. Fig.5.14b shows that Decomp+GN-BP can achieve a 0.71 MSE of headloss when 9% ($\approx$ 22) pipe breaks.

5.4.4 Faulty Zones Identification

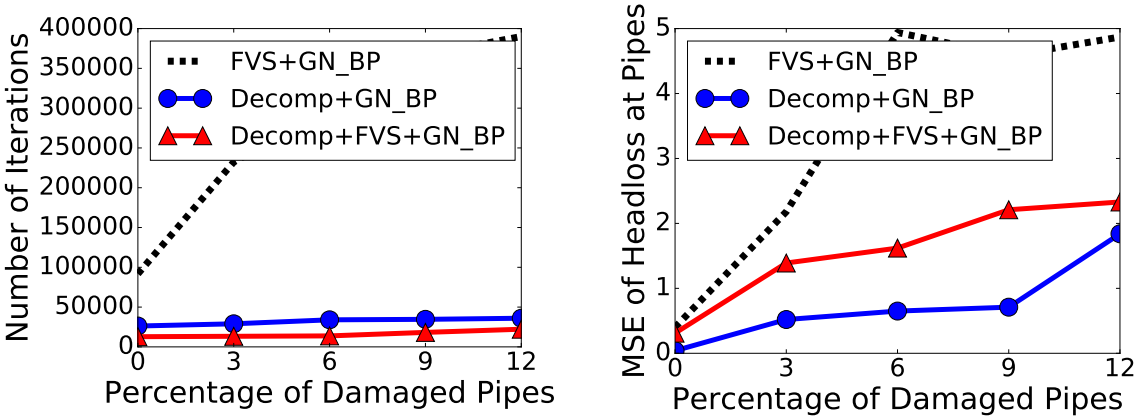
This section explores the hydraulic state estimation for the faulty zones identification. To enable the ability of a system to minimize disruptions and return to the normal function after disruptive incidents, it is important to quickly detect and localize faulty regions such that this information can be used by water agencies and city planners for damage control,



(a) Number of Iterations

(b) Mean Square Error

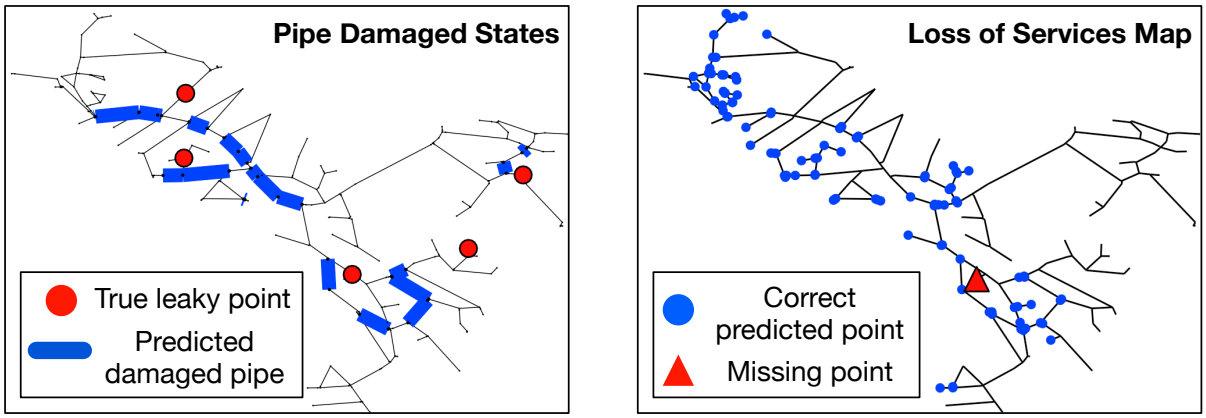
Figure 5.13: WSSC-SUBNET - (5.13a) total number of iterations to converge and (5.13b) mean square error of headloss at pipes versus the percentage of sensor failures.



(a) Number of Iterations

(b) Mean Square Error

Figure 5.14: NET3 - (5.14a) number of iterations to converge and (5.14b) error of headloss at pipes versus percentage of pipe failures.



(a) Pipe Damage States (b) Loss of Services Nodes

Figure 5.15: WSSC-SUBNET - (a) predicted damaged pipes with TP = 0.8, FP = 0.4 of distance threshold 200m; (b) predicted loss-of-service nodes with precise = 1 and recall = 0.99.

community notifications and evacuation plans. We use WSSC-SUBNET in this case study, since the coordinates of its water nodes are their true geo-locations. In this study, 5 major leak events with different large leak volumes are generated at random locations, which are indicated by circles in Fig.5.15a. According to the modeling of leak events (Sec.5.2.5), each pipeline is split into two segments, and the leaky point on a pipe can cause the difference on the headloss between these two segments. The key observation is that the true headloss can be used to identify the broken pipes with TP = 1, FP = 0. In Fig.5.15a, our approach is able to localize damaged pipes with TP = 0.8 of a distance threshold 200m. It can be seen that though the predicted locations are not the exact leaky points, it can help narrow down and target the potential faulty regions such that detailed examinations can be executed effectively. Water service availability is another important resilience metric to identify the places where they loss the access to the facility. The delivered water volume can be calculated using (2.1) where the node pressure is computed using its elevation and estimated hydraulic head. Figure 5.15b shows that 99% of loss-of-service nodes can be localized. With this information, the portable water can be delivered for drinking and other sanitary purposes.

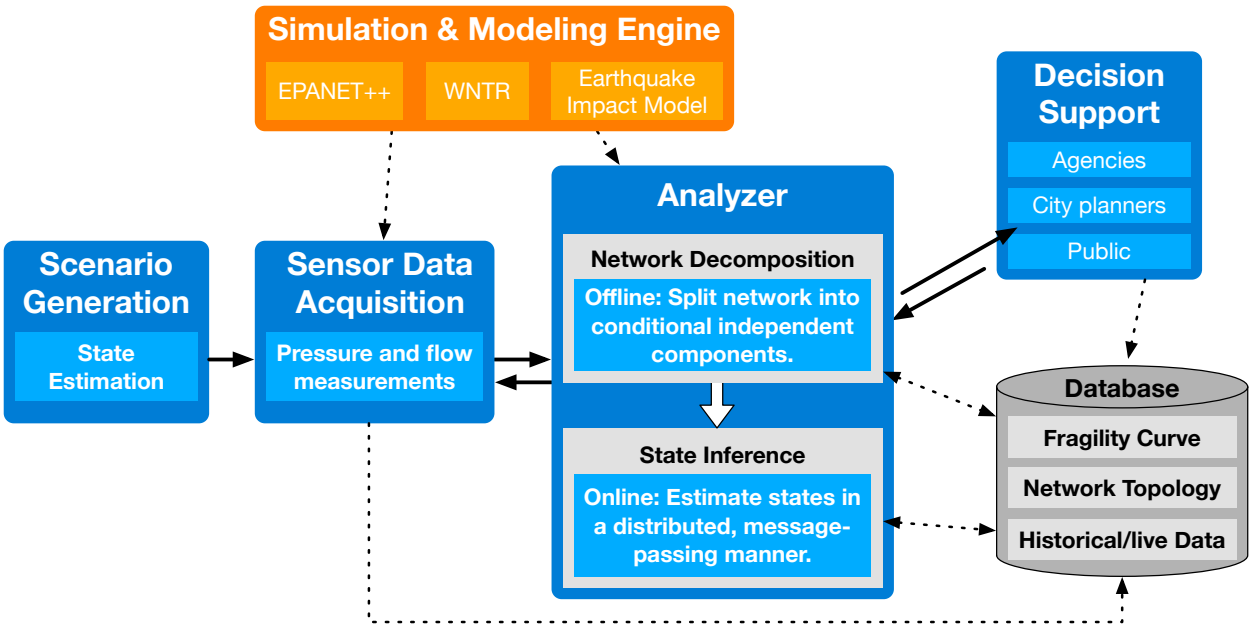


Figure 5.16: This chapter leverages the AquaSCALE system architecture for state estimation under extreme events.

5.5 Chapter Summary and Discussion

In this chapter, we present a novel probabilistic state estimation approach for fault identification, which combines physical constraints with structured nondeterministic information into a single cyber-physical graphical model, for water distribution infrastructures. We consider the real-world water networks under disasters, where different percentages of physical infrastructures and monitoring devices are disrupted, and the proposed two-phase mechanism is scalable for state estimation in large-scale water networks. This paves the way for distributed control for next generation water CPHI systems. One can imagine that the estimated damaged state information is fed into a control decision process, where the pipe network would be reconfigured using remotely controlled valves to save both water and customer demand issues [65].

We add to the previous chapter's approach by improving the resilience of water infrastructures under large massive failures through our proposed AquaSCALE platform introduced in

Chapter 3. The operation of water infrastructures is governed by partial differential equations (5.5) and parameters embedded in these equations are spatially dependent. It is intractable to efficiently abstract useful and correct information from such non-linear, dynamic environment for state estimation. Therefore, as shown in Fig. 5.16, we leverage data from multiple information sources including network topology, failure model and fragility curves as the source for a priori knowledge, hydraulic state estimates as the source for posterior information, and simulation/modeling engines for analysis support. With this comprehensive view of the system, we identify and pre-process some tasks that can be executed offline to save the computational complexity at the run time. Altogether, this enables an efficient state estimation of the water CPHI system when a large number of failures are presented.

Chapter 6

Human-enhanced Analytics for Resilience

Real-time fault source identification is critical in complex distributed infrastructures, e.g., water infrastructures, where failures are difficult to isolate and human operators are often involved (Section 1.3). While previous two chapters explore the water quantity (pipe failures) problem, this chapter focuses on the contaminant source identification in water networks using human-in-the-loop based sensing.

We present AquaEIS, an event-based middleware tailored to the problem of locating sources of failure (e.g., contamination) in community water infrastructures. The inherent complexity of underground hydraulic systems combined with aging infrastructure presents unique challenges. AquaEIS combines online learning techniques, model-driven simulators and data from limited sensing networks to intelligently guide human participants (e.g., domain expert, field staff, consumers) in identifying contaminant sources. The framework integrates the necessary abstractions with event processing methods into a workflow that iteratively selects and refines the set of potential failure points for human-driven *grab sampling*. The integrated

platform utilizes Hidden Markov Model (HMM) based representations along with field reports for event inference; reinforcement learning (RL) methods have also shown promise for further refining event locations and reducing the cost of human engagement. Our approach is evaluated in real-world water networks under a range of distinct events. The results show that AquaEIS can significantly reduce the number of sampling cycles, while ensuring localization accuracy (identified 100% of the failure events in less than 5 sampling cycles as compared to a baseline that can only identify 38% of the events).

6.1 Chapter Overview

In recent years, water utilities have had increasing concerns about the possibility of harm due to accidental or intentional contamination of water systems [44]. [96] shows that, today in the U.S., an estimated 19 million people are exposed to the contaminated water and at least 610 contaminated locations. Public awareness has increased dramatically due to media coverage since 9/11 - incidents have exposed the vulnerability of populations to toxins and contaminants [56, 27]. Accidental contamination events are also becoming more frequent, for instance, metals in pipe materials can leach into the system, ground contaminants can permeate plastic pipes, persistent or transient pressure loss can result in chemicals entering the network through backflow and contaminated soil water entering through leaky joints [90, 108]. In such scenarios, water utilities must quickly identify contaminant sources to ensure the maximum effectiveness of intervention strategies and the minimization of exposure of populations at large to the contaminants.

Challenges: Given that the hydraulic behavior is dynamic and non-deterministic, it is non-trivial to infer the contaminant source with binary measurements (current sensing technologies yield binary indications of possible contaminant presence) from spatially-sparse in-situ sensors (network of hundreds or thousands of pipelines is not actively monitored). Such problem is the canonical "inverse problem", and therefore multiple solutions or near-solutions often exist. Since source injections can originate at any point throughout the network at any time, the solution space can be very large (roughly number of network nodes times number of historical time steps considered), making it intractable to solve the problem uniquely. The choice of a solution is further complicated by the shortage of measurements as compared to source parameters. A pragmatic approach proposed by civil engineers argues that a unique solution is not needed, one can identify a set of locations for which a contaminant injection is possible [111]. Manual samplings (*grab samples*) can help further reduce the resulting set that is likely to be large due to the low density of in-situ sensing - the human engagement

however involves cost and time [142]. This highlights the need for an integrated view of the state of CPHI system to support cost-accuracy-time tradeoffs inherent in fault source identification.

Key Contributions of This Chapter:

• **Integrated platform for event identification in CPHI systems:** We propose and develop AquaEIS, a platform that supports a holistic approach to fault source identification in complex distributed settings. AquaEIS integrates multiple sensing modalities (devices, human-as-a-sensor), computing (ML, simulations) and domain knowledge to enable an observation, analysis and adaptation loop in the system. Though inspired by water infrastructure resilience, the proposed end-to-end solution can be applied to other "flow/traffic/current"-path-dependent systems. (Sec.6.2).

• **Event-driven profile generation:** Our approach abstracts the physical nature of phenomena spread (i.e., contaminant transport) into apriori profiles of anomalies represented using *impact matrices*, which are integrated with live readings from multiple data sources for event processing (Sec.6.2).

• **Online iterative event processing:** AquaEIS is intended to execute in an operational setting on-the-fly, where event processing learns from executions and triggers needed actions to reduce human efforts. At runtime, an iterative event processing strategy is executed in two steps: location inference and location refinement. Here, an approximate set of contaminant sources (often fairly large at early stages) is inferred using current available measurements, which is then refined through human-driven grab samples. The process is repeated until the number of likely sources is sufficiently small. For inference, we propose an HMM model with a pruning technique to quickly eliminate solutions that are inconsistent with physics (i.e., hydraulics) and measurement information (Sec.6.3); the refinement step implements a RL based approach used to determine optimal sampling locations such that it can help locate

the source within a small number of sampling cycles (Sec.6.4).

• **Evaluation in real-world water networks:** We evaluate the AquaEIS approach using a complete ensemble of contamination events on two real-world water networks (Sec.6.5).

6.2 The AquaEIS Approach Overview

AquaEIS (Fig.6.1) is built on the AquaSCALE platform and operated as a real-time event processing service, which can offer utilities the ability to more accurately locate the point of failures. Timely detection can provide valuable information to enable rapid decontamination and disposal response e.g., hydrant flushing. AquaEIS composes ML (RL), modeling (HMM), and data collection methods (in-situ and grab-sampling) at different levels of granularity and latency for a comprehensive system. This is required since limited sensing devices only capture phenomena spread (e.g., contamination), not the fault source, in the networked system. Sensor data has inaccuracies; techniques to process the data have uncertainties. Additional sensing in the form of human operators can help reduce these uncertainties by strategic sampling at selected points in the network, which involves cost and time.

The first step to enable such a system is accurate modeling of features of the physical system and its operations. Water systems are usually described as a set of links and nodes, where links represent pipes, valves and pumps, and nodes represent reservoirs, tanks, joints and end-users (e.g., building). The contaminant spreads through the system based on the network topology and its flow patterns taking the system from one configuration (state) to another. We represent configurations of interest using an event-based architecture and design abstractions that can capture the state of the system.

Event-driven profiles generation - In this step, our goal is to capture signatures/profiles of how contamination events manifest in the underlying network. Given a contaminant

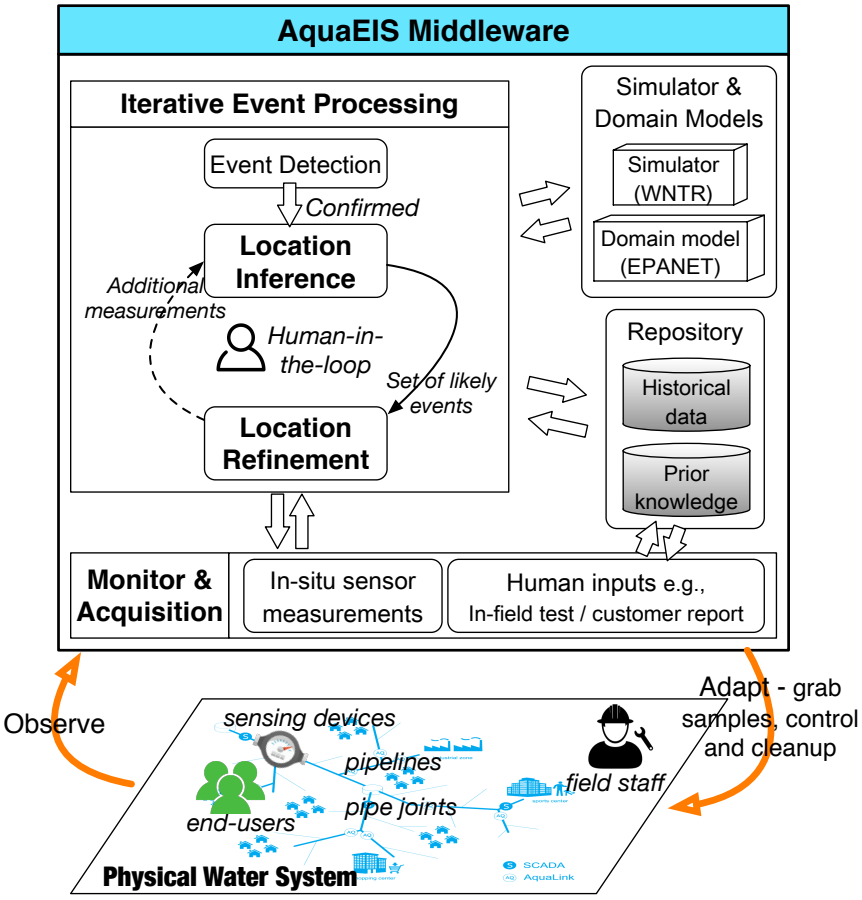


Figure 6.1: AquaEIS integrated event identification system.

injected at a certain node (source), the associated time delays and impact on other nodes depend on the transient flow paths and rates, which are made explicit and thus available for use [111]. To incorporate this knowledge, we generate apriori profiles of anomalies using a commercial-grade water quality model - EPANET [107], where the contaminant is assumed to be injected continuously over time and its transport behavior is modeled as the movement of a non-reactive tracer material through the network [108]. Specifically, we simulate a large ensemble of contamination events over a long enough period such that the length of the simulation is as long as the longest travel time from sources to other nodes. A contamination event is an injection of the contaminant at a particular node beginning at a particular time of day and we consider events that can occur at every node in the network starting at any time of a day. It is worth noting that the contaminant injected at same node but at a different time can propagate differently due to time-varying flows. We assume one event at a time, where the contaminant is injected continuously over time. This ensemble of simulations generates profiles of anomalies which we represent using a collection of impact matrices indexed by time. The impact matrix is an explicit event-to-impact mapping, which contains contamination events as rows and all network nodes as columns. Entries are given a value of 1 to indicate that the event can contaminate the given node, and a value of 0 otherwise. For example in Fig.6.2, 9 events are simulated on NET1 water network (from US EPA) by introducing a contaminant at a random node at an arbitrary time. A matrix at 2PM captures contaminated nodes (columns) caused by these 9 events (rows) at 2PM. As time progresses, the matrix may or may not change depending on the flows and events. This collection of matrices can reflect the flow-path-dependent contaminant transport and will be integrated with live readings from in-situ sensors and human inputs for event identification.

Event monitoring and data acquisition - Often, a contamination event is witnessed and detected by in-situ sensing platforms, and the initial identification with sparse and limited measurements is likely to result in a fairly large set of possible sources. Given the inherent complexity of instrumenting underground systems; utilities in practice obtain

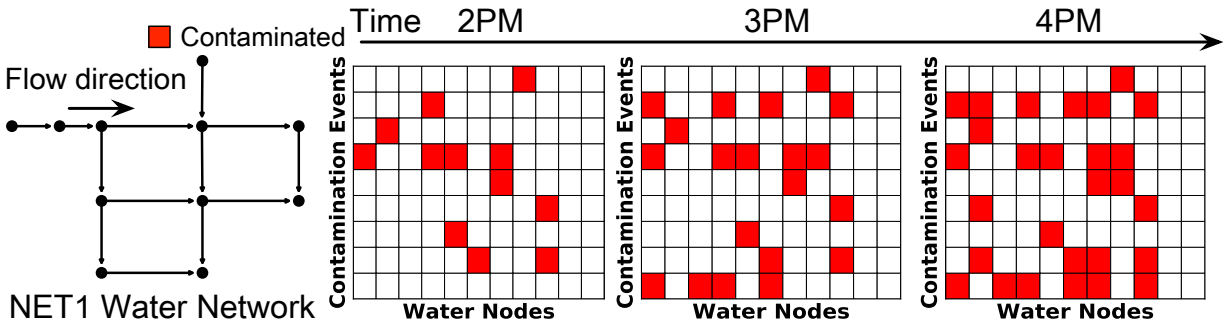


Figure 6.2: Illustration of impact matrices of NET1

additional measurements in the form of grab samples through human participants (e.g., field staff, community manager, customer), which can help in narrowing down the resulting set. Humans-as-sensors are indispensable in settings where sensing needs are uncertain and expensive. The advantage of manual grab sampling is its flexibility to provide spatial diversity at significantly reduced cost when compared with in-situ sensors, however, a grab sample at a certain location would only provide a single measurement in time as opposed to continuous monitoring from a sensor. Due to their characteristics, live sensor readings and human inputs are leveraged differently during the event processing.

Online iterative event processing - To enable a timely and reliable fault identification, event processing occurs in an iterative manner in two key steps: location inference and location refinement. When a contamination is detected, an approximate set of source locations is inferred using current available measurements. If the resulting set is large, optimal sampling locations are determined and human participants are invoked to gather additional information. The cycle of collecting grab samples and inferring source locations is continued until the true source of contamination is identified.

Location Inference - In the presence of an unknown contamination event, water quality varies in time subject to constraints imposed by the fluid mechanics. Live sensor readings capture quality changes over time, however, they are spatially sparse and only provide yes/no indications. To better infer the fault source from limited observations, a methodology should

incorporate the knowledge of spatio-temporal variations of physical system dynamics. As such, this information allows to rule out the solutions that are inconsistent with the physical laws that govern the contaminant propagation. The aggregation of information over time can help distinguish the events resulting in the same instantaneous readings. The sequence of quality measurements reveals the information about how the water quality condition evolved in the past, which can potentially be used to backtrack the entire realization path of the contaminant propagation, identify the source location and inform appropriate actions to remove the source of poor water quality. We therefore propose to formulate the problem of source identification as an HMM inference problem, i.e., inferring the unobserved/hidden water quality status based on time series of sensor readings. In addition, we can also use HMM to estimate the mostly likely propagation path and predict the future extension of contamination - this information is necessary to help devise appropriate control and clean-up strategies to limit cascading failures and reduce exposure to contaminated water. Human inputs as point measurements are used within a proposed pruning technique. It allows an efficient pruning of the search space by imposing the physically based "structural regularization" on the solution via the apriori generated profiles.

Location Refinement - Due to the limited measurements available at the early stages, the identified set of possible sources may be large or small depending on the sensor configuration and the contamination event. Additional efforts (using grab samples) may be required to reduce this uncertainty. Observe that, depending on the number of likely sources and maximum number of samples that can be taken at the same time, multiple sampling cycles may be required. Obviously, it is desirable to reduce the resulting set to a tractable number in as few cycles as possible. Therefore, location refinement is considered as a sequential decision problem, where the current selection should learn from previous decisions and consider its "long-term" influence on subsequent ones. However, it is non-trivial for an informed decision making in such a stochastic environment due to the dynamic flows and unknown events. We therefore formulate it using a RL framework, which offers an online mechanism to learn an

optimal policy that selects sampling locations with more contributions in the "long run" [63, 124].

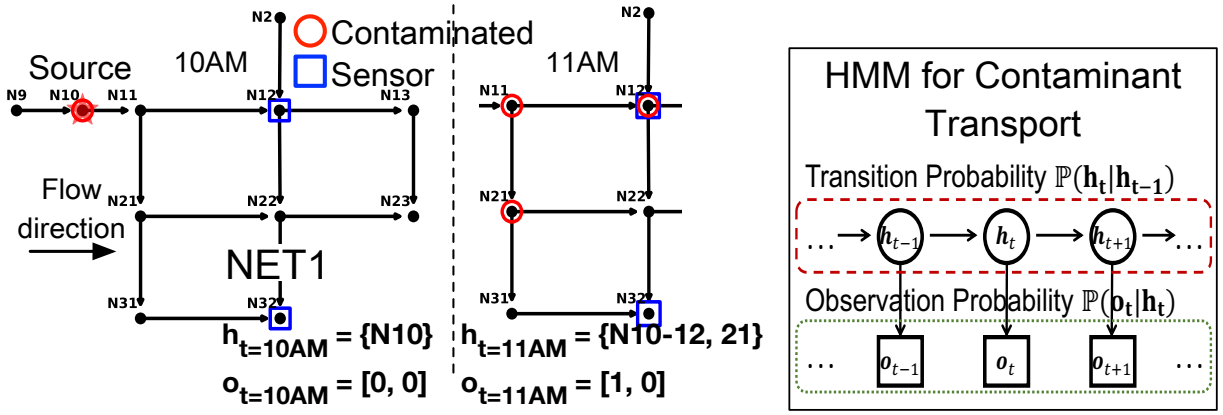
As indicated earlier, the inference and refinement cycle is repeated until the size of the candidate source set is reduced to a desired bound or the solution converges. Recall that the fault source identification is an inverse problem where the existence of a unique solution is not guaranteed. The final resulting set of possible sources may be not singleton depending on the network topology and the contamination event.

6.3 Event Location Inference

We now introduce the first step of event processing, i.e., location inference. Here we use measurements from both in-situ platforms (Sec.6.3.1) as well as human participants (Sec.6.3.2), along with pre-built profiles, to estimate contaminant sources. An in-situ sensor or a human input would yield a positive measurement (unsafe) if the contaminant concentration is above a certain threshold and a negative measurement (safe) otherwise.

6.3.1 Inferring using Live Sensor Readings

Due to the fact that a contaminant spreads through the water network based on the time-varying flows, the propagation behavior is considered as a structured stochastic process. We model it as an HMM and formulate the event identification as an HMM inference problem. We first apply the standard forward-backward algorithm and propose an approximate approach using particle filter that offers a viable alternative for improving the speed of estimation.



(a) Contaminant transport (b) HMM

Figure 6.3: (a) Example of contaminant transport on NET1. (b) Illustration of the HMM for contaminant transport.

Modeling Contaminant Transport using HMM

We consider a water network consisting of V nodes denoted by $\mathcal{V} = \{1, ..., V\}$ and use $v \in \mathcal{V}$ to index node v. The water quality status of a network is represented by a vector $\mathbf{c} = (c_1, ..., c_V) \subseteq \{0, 1\}^V$, where $c_v = 1$ indicates that node v is contaminated. Let $\Gamma(\mathbf{c}) = \{v \in \mathcal{V} : c_v = 1\}$ denote a set of contaminated nodes for quality status $\mathbf{c}$, and $\mathbf{c}_t$ and $\Gamma_t = \Gamma(\mathbf{c}_t)$ denote the status at time t. Since sensors can only provide binary measurements, sensor reading at node v at time t is denoted by $\boldsymbol{L}_{v,t} \in \{0, 1\}$, where $\boldsymbol{L}_{v,t} = 1$ indicates contamination. Let $\mathcal{V}^o \subseteq \mathcal{V}$ denote the subset of nodes where sensors are deployed. The objective is then to infer about quality status $\mathbf{c}_t$ or Γ_t given observation $\boldsymbol{L}_{\mathcal{V}^o,t}$ over time. To locate the source, we want initial status $\mathbf{c}_{t=0}$ or $\Gamma_{t=0}$. It is worth noting that due to the sparse sensor network, many events may be noticed by same sensor at the early stages, adding to the difficulty of inferring the source. At time t, we model the water quality status Γ_t, which is not directly observed, as the hidden state $\mathbf{h}_t \equiv \Gamma_t$, and the sensor readings $\boldsymbol{L}_{\mathcal{V}^o,t}$ as the observed state $\mathbf{o}_t \equiv \boldsymbol{L}_{\mathcal{V}^o,t}$. For example in Fig.6.3a, a contamination event occurs at node N10 starting at 10AM on NET1 water network. As the time progresses, nodes may be contaminated ($\mathbf{h}_t$) and sensors may detect the contamination ($\mathbf{o}_t$). In Fig.6.3b, the evolution of $\mathbf{h}_t$ is formed as a Markov chain with state transition probability $\mathbb{P}(\mathbf{h}_t|\mathbf{h}_{t-1})$ and conditional

on $\mathbf{h}_t$, the observation is an independent process following observation probability $\mathbb{P}(\mathbf{o}_t|\mathbf{h}_t)$.

Given a water network, assuming there are N number of hidden states and M number of observed states, an HMM is characterized by the three elements: (a) state transition probability matrix - $A_{N\times N} = \{\mathbb{P}(\mathbf{h}_j|\mathbf{h}_i)\}$ for $1 \leqslant i, j \leqslant N$; (b) observation probability matrix - $B_{N\times M} = \{\mathbb{P}(\mathbf{o}_k|\mathbf{h}_j)\}$ for $1 \leqslant j \leqslant N$ and $1 \leqslant k \leqslant M$; (c) initial state distribution vector - π_0, which are represented as $\lambda = (A, B, \pi_0)$. Matrices A and B can be learnt from the apriori generated impact matrices by the Expectation Maximization algorithm [103]. Here the detail is omitted due to the page limitation. Given an observation sequence $o_{0:T} = (\mathbf{o}_0, \mathbf{o}_1..., \mathbf{o}_T)$ and a model λ, our goal is to find a probability distribution $\pi_t(\mathbf{h})$ to estimate the hidden state $\mathbf{h}_t$, where $\pi_t(\mathbf{h}) \equiv \mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:T}, \lambda)$ for $0 \leqslant t \leqslant T$ and T is the length of observation sequences. We use $\mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:T})$ for $\mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:T}, \lambda)$, since λ is fixed once a model is learnt. In particular, to locate the source, we want the state distribution at $t = 0$, i.e., $\pi_{t=0}$. Following Bayes' rule and the conditional independence of $o_{t+1:T}$ and $o_{0:t}$ given $\mathbf{h}_t$, $\mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:T})$ can be written as

$$\begin{aligned} \pi_t(\mathbf{h}) &\equiv \mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:T}) = \mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:t}, o_{t+1:T}) \\ &\propto \mathbb{P}(o_{t+1:T}|\mathbf{h}_t = \mathbf{h})\mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:t}) \end{aligned} \tag{6.1}$$

The Forward-Backward Algorithm

To solve (6.1), a short elaboration of standard forward-backward algorithm [102] is first given. This algorithm solves it by recursively computing a set of forward probabilities $\mathbb{P}(\mathbf{h}_t = \mathbf{h}|o_{0:t})$ and a set of backward probabilities $\mathbb{P}(o_{t+1:T}|\mathbf{h}_t = \mathbf{h})$. Given model $\lambda = (A_{N\times N}, B_{N\times M}, \pi_0)$ and a sequence of observations $o_{0:T}$, the forward messages $F_{T\times N}$ can be computed using:

$$F_{t,*} = F_{t-1,*} \cdot A \cdot \text{diag}(\mathbf{o}_t) \qquad 1 \leqslant t \leqslant T \tag{6.2}$$

where $\mathrm{diag}(\mathbf{o}_t)$ is a diagonal matrix of vector $\mathbf{o}_t$, and $F_{t,i} \propto \mathbb{P}(\mathbf{h}_t = \mathbf{h}_i | o_{1:t})$ for $1 \leqslant i \leqslant N$ with $F_{0,*} = 1/N$. The initial forward message can be set uniformly among all states as $F_{0,*} = 1/N$, indicating no prior knowledge on the state before observations. Similarly, the backward messages $R_{T\times N}$ are computed using:

$$R_{t,*} = A \cdot \mathrm{diag}(\mathbf{o}_t) \cdot R_{t+1,*} \qquad 0 \leqslant t \leqslant T-1 \tag{6.3}$$

where $R_{t,i} \propto \mathbb{P}(o_{t+1:T} | \mathbf{h}_t = \mathbf{h}_i)$ for $1 \leqslant i \leqslant N$ with $R_{T,*} = 1$. The algorithm runs in time $O(N^2 T)$. However, we observe that the transition probability matrix A is sparse (Remark 6.1). This implies that substantial computation on zero entries can be reduced. We thus propose a particle filter based algorithm that exploits this specific structure of our HMM model and can reduce the computational complexity.

Remark 6.1 (Sparsity in transition matrix)**.** *A feature of contaminant transport is the sparsity of transition probabilities, in the sense that only a small portion of nodes is contaminated in initial stages and with high probability, the new contaminated nodes* $\Gamma_{t+1} \backslash \Gamma_t$ *only occur on a small subset of nodes.*

A Particle Filter Based Approach

We present a particle filter based approximation, taking the advantage of Remark 6.1, to estimate efficiently the hidden water quality status. Particle filter uses a set of randomly chosen weighted samples (particles) to estimate the parameters of a system (filter procedure) by implementing a recursive Bayesian filter using Monte Carlo (MC) simulations. The basic idea is to represent $\mathbb{P}(\mathbf{h}_t | o_{0:T})$ by a set of random particles with associated weights and to estimate $\pi_t(\mathbf{h})$ in (6.1) based on them [16]. The number of particles P determines the computational complexity, which allows to control the running time by choosing an appropriate P.

Let $\{\mathbf{h}_{0:t}^i, w_t^i\}_{i=1}^P$ denote a set of particles that characterizes $\mathbb{P}(\mathbf{h}_t|o_{0:T})$ where $\{\mathbf{h}_{0:t}^i, i = 1, ..., P\}$ is a set of support points with weights $\{w_t^i, i = 1, ..., P\}$, and $\mathbf{h}_{0:t} = \{\mathbf{h}_\tau, \tau = 0, ..., t\}$ is a set of all states up to time t. The weights are normalized such that $\sum_i w_t^i = 1$. Then, $\pi_t(\mathbf{h})$ can be approximated as

$$\hat{\pi}_{0:t}(\mathbf{h}) = \sum_{i=1}^{P} w_t^i \delta(\mathbf{h}_{0:t} - \mathbf{h}_{0:t}^i) \tag{6.4}$$

where $\delta(x - x_0)$ is the Dirac delta function with the delta mass at x_0 and $\hat{\pi}_{0:t}(\mathbf{h}) = \{\hat{\pi}_\tau(\mathbf{h}) : \tau = 0, ..., t\}$. Algorithm 6 outlines the proposed particle filter based inference, where the two main steps at each time instant are: (a) sampling particles based on transition probabilities such that the probabilities can be reflected by the frequencies of particles, mathematically expressed as

$$\mathbf{h}_t^i \sim A_{\mathbf{h}_{t-1}^i,*} = \mathbb{P}(\mathbf{h}_t|\mathbf{h}_{t-1}^i) \qquad 1 \leqslant i \leqslant P, \tag{6.5}$$

and (b) updating weights based on the instantaneous observations such that particles can be weighted based on the evidence, mathematically expressed as

$$w_t^i = w_{t-1}^i B_{\mathbf{h}_t^i,\mathbf{o}_t} = w_{t-1}^i \mathbb{P}(\mathbf{o}_t|\mathbf{h}_t^i) \qquad 1 \leqslant i \leqslant P. \tag{6.6}$$

In this manner, one can approximate $\hat{\pi}_{0:t}$ by augmenting existing particles that constitute an approximation to $\hat{\pi}_{0:t-1}$ with new observations $\mathbf{o}_t$. As the algorithm runs, some weights may become very small. To reduce the effects of degeneracy and concentrate on those with large weights, we follow the rule of thumb to resample particles based on their weights $\{w_t^i\}_{i=1}^P$ when the ratio $1/\sum_{i=1}^P (w_t^i)^2$ falls below the threshold $P/2$ [54]. The basic idea of resampling is to concentrate on particles with large weights by eliminating those with small weights. In this paper, we use systematic resampling scheme [67], since it takes $O(P)$ linear time and minimizes the MC variation [16], while other resampling techniques can be applied as

alternatives. The resampling will generate a new set of particles $\{\mathbf{h}_{0:t}^{i^*}, w_t^{i^*}\}_{i^*=1}^{P}$ and weights are reset to $w_t^{i^*} = 1/P$ since particles are i.i.d samples.

The proposed algorithm runs in $O(PT)$ time, where the computational complexity scales linearly with the number of particles. In addition, according to the law of large numbers theorem, the accuracy of the discrete approximation increases in the order of $O(1/\sqrt{P})$ [36]. By choosing the number of particles, one could have a tradeoff between the computational cost and the estimation accuracy.

Algorithm 6 The particle filter based inference

Input state transition matrix A, emission matrix B, initial state vector π_0, observations $\{\mathbf{o}_t : t = 0, 1, ...\}$
Output hidden state estimation $\hat{\pi}_\tau(\mathbf{h})$ for $0 \leqslant \tau \leqslant t$

Initialization at $t = 0$: draw particles $\{i, i = 1, ..., P\}$ from π_0: $\mathbf{h}_0^i \sim \pi_0$; set weights $w_0^i = P^{-1}$
for $t = 1, 2, ...$ **do**
    for $i = 1, 2, ..., P$ **do**
        Sample particle $\mathbf{h}_t^i$; update weight w_t^i via (6.5, 6.6)
    end for
    if $1/\sum_{i=1}^{P}(w_t^i)^2 < P/2$ **then**
        Resample $\{i^j = j\}_{i=1}^{P}$, $j \in \{1, ..., P\}$ using $\{w_t^i\}_{i=1}^{P}$
        Set $\mathbf{h}_{0:t}^{i^*} = \mathbf{h}_{0:t}^{i^j}$, $w_t^{i^*} = P^{-1}$ for $i^* = 1, ..., P$
    end if
end for
Compute $\hat{\pi}_{0:t}(\mathbf{h})$ using (6.4)

6.3.2 Inferring using Human Inputs

The fact that in-situ sensors are sparsely deployed and can only provide binary measurements leads to the issue of event detectability (Remark 6.2). As an example shown in Fig.6.4, we use the small water network NET1 configured with two in-situ sensors (as in Fig.6.3a), and generate 9 contamination events with different injection locations. The figure shows that some events ((a, i), (b, d, f, g), and (c, h)) are inherently indistinguishable based on the

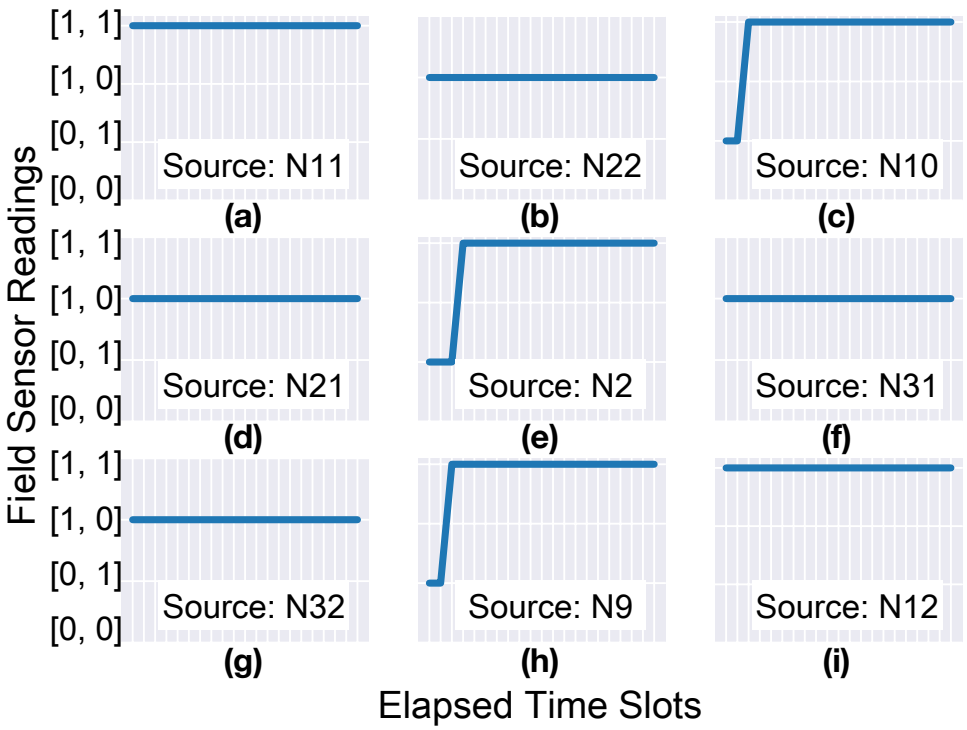


Figure 6.4: Time series of observations from in-situ sensors with a sampling rate of 15min. Contaminant is injected at different nodes, starting at 10AM on NET1 water network.

network topology and sensor network configuration, i.e., same field sensor readings at any time instant. We also examine the incident detectability on a large-scale water network with total pipe length 65.75km and 10% in-situ sensors deployment penetration. The result shows that 30% of contamination events cannot be distinguished with only in-situ sensor measurements. Recall that this is an inverse problem where a unique solution is naturally not guaranteed. Not to mention that the limited measurements will make the problem even harder. Therefore, it is necessary to collect and incorporate additional measurements in order to refine the estimate. Human involvement in data capture is valuable. Human-driven grab samples as single measurements in time are used to prune away inconsistent solutions. As hinted above in Sec. 6.2, the flow-path-dependent hydraulics determines if response from a node is positive (unsafe), contaminant sources will be its upstream nodes, while if negative (safe), its upstream nodes can then be excluded.

Remark 6.2 (Detectability of in-situ sensors)**.** *It is likely that multiple contamination events*

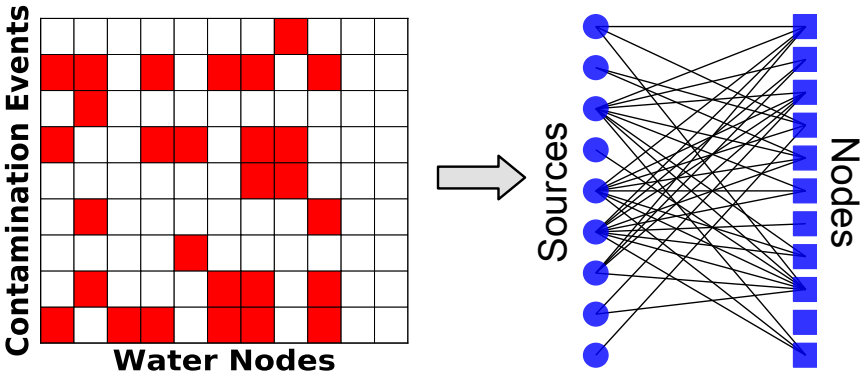


Figure 6.5: Impact matrix to source-to-node bipartite graph.

manifest same sensor readings over time, i.e., $o_{0:t}(\mathbf{h}) = o_{0:t}(\mathbf{h}')$ *for* $\mathbf{h} \neq \mathbf{h}'$*, thus are not distinguishable using only in-situ sensors.*

To better illustrate the pruning process, we first build a bipartite graph for each impact matrix (Fig.6.5). The bipartite graph is a source-to-node map indicating whether or not a network node can be contaminated by an event occurring at a particular source node but starting at any historical time step. A bipartite graph at time t is $\mathcal{G}_t = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$, where $\mathcal{X} \subseteq \mathcal{V}$ is a set of sources, $\mathcal{Y} \subseteq \mathcal{V}$ is a set of nodes and $\mathcal{E} \subseteq \mathcal{X} \times \mathcal{Y}$ is a set of edges connecting source $i \in \mathcal{X}$ and node $j \in \mathcal{Y}$, namely an event occurring at i can have an impact on j at time t. We refer an edge $e \in \mathcal{E}$ as (i, j) where $i \in \mathcal{X}$ and $j \in \mathcal{Y}$, and $\mathcal{G}$ as $\mathcal{G}_t$ in general. Let $U(j)$ define the upstream reachability set of nodes $j \in \mathcal{Y}$, that is $U(j) = \{i : (i, j) \in \mathcal{E}, i \in \mathcal{X}\}$. Then, we can use the response from node j, which is either positive or negative, to determine if its upstream nodes $U(j)$ are possible sources (positive) or must be excluded (negative). Note that without such pruning, the number of unknown incidents is on the order of the number of network nodes multiplied by the number of starting times - easily exceeding 10^6 unknown parameters for practical problems [111]. Algorithm 7 describes this pruning operation, which is straightforward. Given a set of nodes with positive response PR and a set with negative response NR, graph $\mathcal{G}$ can be pruned by taking the intersection of sources that are upstream reachable from PR and removing those that are upstream reachable from NR.

Algorithm 7 The pruning process

1: **Input** a source-to-node map $\mathcal{G} = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$, nodes with positive response PR and negative response NR.
2: **Output** a pruned map $\mathcal{G}' = (\mathcal{X}', \mathcal{Y}', \mathcal{E}')$.

3: **Initialization** $\mathcal{X}' = \emptyset$
4: Update $\mathcal{X}'$ using positive response $\mathcal{X}' = \bigcap_{j \in PR} U(j)$
5: Update $\mathcal{X}'$ using negative response $\mathcal{X}' \setminus = \bigcup_{j \in NR} U(j)$
6: Generate $\mathcal{G}'$ by pruning $\mathcal{G}$ using $\mathcal{X}'$

6.4 Event Location Refinement

After event inference is performed following the detection of a contamination (often by in-situ sensors), it is likely that the identified set of potential sources is fairly large due to the limited measurements. Additional efforts in the form of manual grab samples will be required to help refine the identified set, such that countermeasures can be executed effectively. This motivates us to design an intelligent strategy to provide optimal decisions on sampling locations. We first define the location selection of grab samples as a sequential decision problem and motivate a reinforcement learning (RL) framework for solving it. We discuss two challenges while applying RL and finally propose a RL based approach integrated with an entropy-constraint filter.

6.4.1 The Location Selection Problem

Recall that Sec.6.3.2 introduced the source-to-node graph with an edge indicating that a contaminant injected at a particular node can have an impact on a certain node. Given a set of sources $\mathcal{X}'$ identified by location inference at time t, a source-to-node map is $\mathcal{G}_t = (\mathcal{X}', \mathcal{Y}, \mathcal{E})$. The objective is to select location(s) $\{j, j \in \mathcal{Y}\}$ based on $\mathcal{G}_t$ over time to reduce $\mathcal{X}'$ in as few sampling cycles as possible, subject to the maximum number of samples that can be taken at the same time. Figure 6.6 shows an example of 2 sampling cycles with

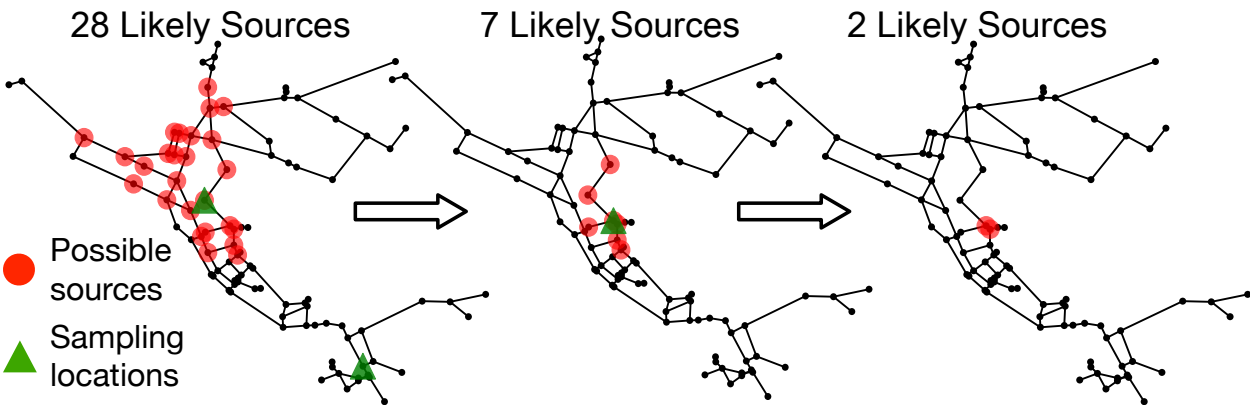


Figure 6.6: Example of sequential decisions on grab-sample locations with maximum 2 samples allowed at a time.

maximum 2 grab samples allowed at a time, where given a set of 28 likely sources, 2 grab samples are taken which reduces the set to 7, followed by another round of sampling with 1 grab sample, and finally 2 possible sources are located.

Intuition behind RL approach - One could consider that the optimal sampling location in each cycle is a set of nodes with the maximum entropy on graph $\mathcal{G}$. As information theory dictates, the data with a larger entropy has more uncertainty and thus conveys more information. It is possible to quickly narrow down the search space by querying nodes with highest uncertainty (maximal information) [35]. However, entropy itself for this problem suffers from two drawbacks: (a) we observe that due to the physical nature of contaminant transport, multiple (collections of) nodes can have same maximum entropy, but result in different source locations. This variance raises the question how to select node(s) from those with maximum entropy which can identify the true source. (b) More importantly, reducing uncertainty greedily is not completely aligned with our objective to seek an optimal *sequence of locations over time*. To address above issues, we propose to apply the RL framework, which is learning to optimize not only the immediate decision but also the next and, through that, all subsequent decisions [124].

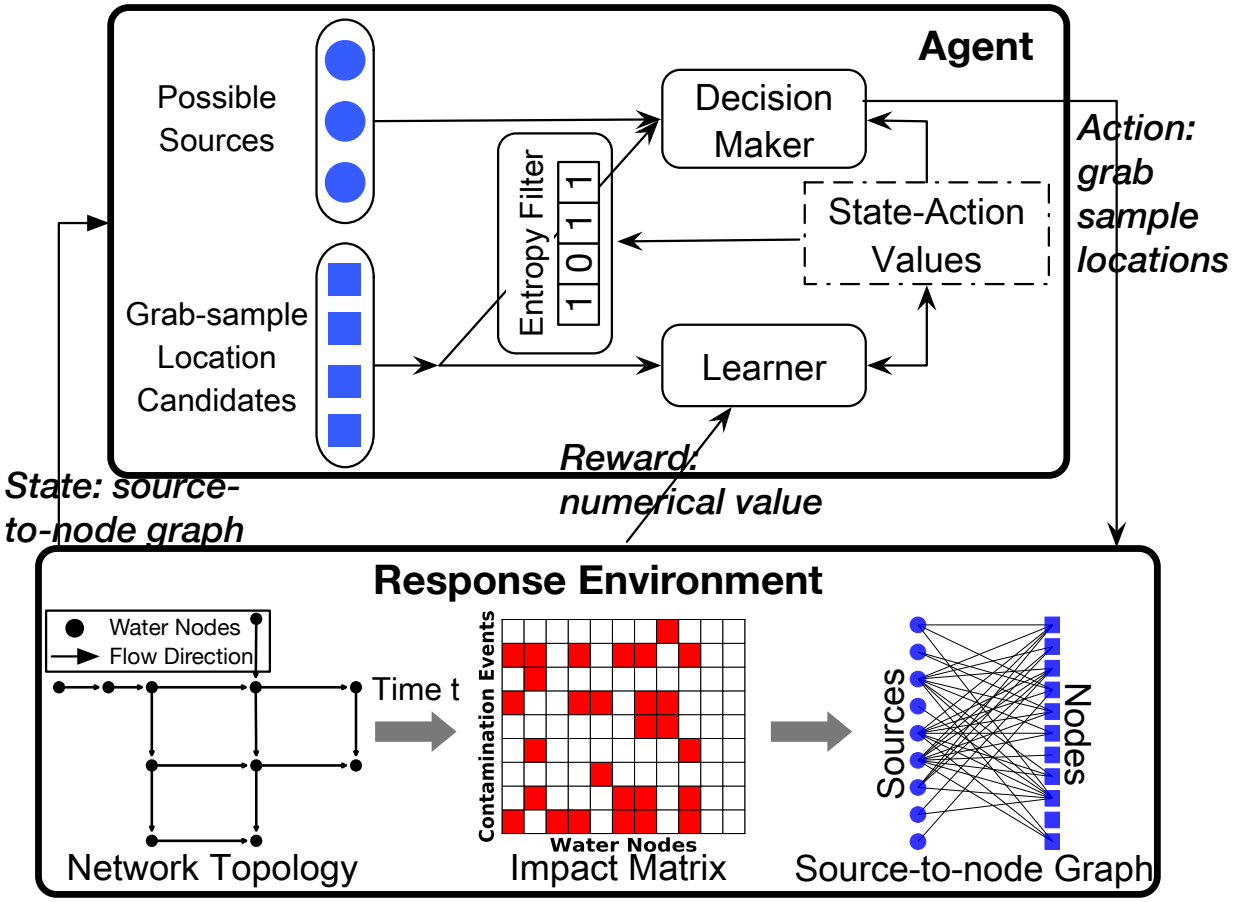


Figure 6.7: The comprehensive view of RL based approach with an entropy filter for sequential location selections.

6.4.2 Modeling Location Selection using RL

Figure 6.7 represents a comprehensive view of adapting RL framework into our problem of selecting sampling locations, where the learner and decision maker together is called *agent*, and the thing it interacts with, comprising everything outside the agent, is called *response environment*. The agent selects grab-sample locations and the response environment responds to it based on the abstracted source-to-node graph. The environment gives rise to numerical rewards that the agent seeks to maximize over time through its choice of sampling locations. In this manner, the agent is able to sense the state of response environment to some extent, takes actions that affect the state and meanwhile has a goal relating to it. The internal components of Fig.6.7 will be explained as follows.

To frame the problem of learning from executions to achieve a goal, we first define four key components:

State: At time step t, the state of response environment $s_t \in \mathcal{S}$ is defined as the source-to-node graph $s_t \equiv \mathcal{G}_t = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$, where $\mathcal{S}$ is the state space. Specifically, a state s contains a set of sources $\mathcal{X}$ and a set of possible impacted nodes $\mathcal{Y}$ that are considered as potential grab-sample locations. The initial state s_0 is determined by the estimate from the location inference step.

Action: On the basis of state $s_t = \mathcal{G}_t$ and maximum number of samples that can be taken at the same time θ, the agent takes an action $a_t \in \mathcal{A}(s_t)$, where a_t is a set of nodes from which grab samples are taken, $\mathcal{A}(s_t)$ is a set of action candidates - combinations of nodes in $\mathcal{Y}$ based on θ, i.e., $\mathcal{A}(s_t) = \{(x_1, ..., x_{\theta'}) : x_i \neq x_j \text{ if } i \neq j,\ x_{i,j} \in \mathcal{Y},\ i, j = 1, ..., \theta', \theta' = 1, ..., \theta\}$, and $\mathcal{A}$ is the action space.

Reward: One step later at $t+1$, in part as a consequence of action a_t, the agent receives a numerical reward $r_{t+1} \in \mathcal{R} \subset \mathbb{R}$ and finds itself in a new state s_{t+1}. $\mathcal{R}$ is the reward space - a subset of real numbers $\mathbb{R}$. Since the aim is to locate the source in as few sampling cycles as possible, we count a penalty (negative reward) of -1 for each sampling cycle and give a delayed reward until the terminal state. A terminal state is either a single source or a set of indistinguishable sources (i.e., have exactly same impact at this time step, in other words, form a complete bipartite graph). A delay reward is given below, assuming true source is TS and $s_t = \mathcal{G}_t = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$:

$$r_t = \begin{cases} 1/|\mathcal{X}| & \text{if } TS \in \mathcal{X},\ |\mathcal{X}| < \text{threshold} \\ -\sum_{i \in \mathcal{X}} \text{dist}(i, TS) & \text{otherwise} \end{cases} \tag{6.7}$$

where s_t is a terminal state, $\text{dist}(i, j)$ computes the shortest distance in terms of number of links (pipes) between nodes i and j and the threshold is to encourage the agent to reduce

the set of likely sources as much as possible, ideally, a singleton with one source. A negative reward is given if the first condition of (6.7) is not satisfied and closer to the true source receives a higher reward. This is to take care of the situations where the contamination event can not be located in a reasonable number of cycles. Instead, nodes that are close to it will be considered. The reward function allows to reflect different criteria that users are interested in. Other metrics such as population exposed to a contaminant and the length of contaminated pipe can also be included.

Policy: A policy defines the agent's way of behaving by mapping states to probabilities of selecting each action, i.e., a probability distribution over set $\mathcal{S} \times \mathcal{A}$. The policy function $p(a|s) \equiv \mathbb{P}(a_t = (x_1, ..., x_{\theta'})|s_t = \mathcal{G}_t)$ takes as input the current state and outputs the probability of actions. The policy is a stochastic rule by which agent can explore the action space and select appropriate actions as a function of states.

According to the definitions above, the agent's goal at t is to select an action so as to maximize the cumulative reward in the long run:

$$J_t \equiv \sum_{k=1}^{T} \gamma^{k-1} r_{t+k} \tag{6.8}$$

where T is the final step (reach a terminal state) and $\gamma \in [0, 1]$ is called the discount rate. The concept of discounting is to control how strongly the agent takes future rewards into account. As γ approaches 1, the agent will strive for a long-term high reward. To train the agent to execute an action in a given state toward the goal, we define a state-action-value function for policy p as q^p, i.e., $q^p : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. If the agent is following policy p, then $q^p(s, a)$ is the expected reward return starting from state s taking the action a:

$$q^p(s, a) \equiv \mathbb{E}_p[J_t|s_t = s, a_t = a]. \tag{6.9}$$

The objective is to find the optimal state-action values $q^p(\cdot)$ following policy p that maximize

the expected return for all state-action pairs, which is defined as:

$$q_*(s,a) \equiv \max_p q^p(s,a) \quad \text{for all } s \in \mathcal{S} \text{ and } a \in \mathcal{A}(s). \tag{6.10}$$

6.4.3 An efficient RL-based Approach

We first discuss two observations (Remarks 6.3 and 6.4) when solving the optimization problem in (6.10) and propose an efficient online temporal difference (TD) based learning approach. Once the optimal state-action values are learnt, grab-sample locations can be selected based on it.

According to the Bellman optimality equation [38], for a specific state-action pair (s, a), (6.10) can be rewritten by substituting (6.8) and (6.9) as:

$$\begin{aligned} q_*(s,a) &= \max_p \mathbb{E}_p[J_t | s_t = s, a_t = a] \\ &= \max_p \mathbb{E}_p[r_{t+1} + \gamma J_{t+1} | s_t = s, a_t = a] \\ &= \mathbb{E}[r_{t+1} + \gamma \max_{a' \in \mathcal{A}(s_{t+1})} q_*(s_{t+1}, a') | s_t = s, a_t = a] \\ &= \sum_{s' \in \mathcal{S}, r \in \mathcal{R}} f(s', r | s, a)[r + \gamma \max_{a' \in \mathcal{A}(s')} q_*(s', a')] \end{aligned} \tag{6.11}$$

where $f(s', r|s, a) \equiv \mathrm{P}(s_{t+1} = s', r_{t+1} = r | s_t = s, a_t = a)$ is the probability of those values (s', r) occurring at step $t+1$, given preceding (s, a). To solve (6.10), we instead need to find a solution of (6.11) for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$ in an efficient manner.

Remark 6.3 (Stochastic response environment)**.** *The response environment is stochastic because of the time-varying flow patterns and the presence of unknown contamination events.*

This uncertainty makes it infeasible to obtain the probability distribution of $f(\cdot|\cdot)$ in an explicit form. However, it is possible to estimate $f(\cdot|\cdot)$ through sequences of states, actions

and rewards from actual or simulated interactions with the response environment. This sequence, called an *episode* in RL, likes this: $s_0, a_0, r_1, s_1, a_1, r_2, ..., s_{T-1}, a_{T-1}, r_T, s_T$ (terminal state). With a certain amount of these episodes, we can average the reward returns for each state-action pair and estimate the distribution of $f(\cdot|\cdot)$.

Remark 6.4 (Large state and action space)**.** *The state space $\mathcal{S}$ and action space $\mathcal{A}$ are combinatorial and can be enormous. Generally speaking, a water network with V nodes can have $|\mathcal{S}| = 2^V$ numbers of possible sources and given a state $s = \mathcal{G} = (\mathcal{X}, \mathcal{Y}, \mathcal{E})$, there are $|\mathcal{A}(s)| = \sum_{i=1}^{\theta} i^{|\mathcal{Y}|}$ numbers of candidates from which grab samples can be taken.*

The problem with large $\mathcal{S}$ and $\mathcal{A}$ is not just the memory needed for a large set of state-action values $q(\mathcal{S}, \mathcal{A})$, but the time and data needed to compute them correctly. Consider a sequence of T sampling cycles, it may need $|\mathcal{A}(s_0) \times \mathcal{A}(s_1) \times \cdots \times \mathcal{A}(s_T)|$ numbers of trial-and-error learning to encounter every state and make a good decision. It quickly makes the search space intractable for a real-world water network (usually with > 100 nodes). We therefore propose a TD based approach - n-step expected Sarsa (NESarsa) integrated with an entropy-constraint filter. Roughly speaking, the TD method uses Monte Carlo (MC) experiments such that it can learn the distribution of $f(\cdot|\cdot)$ directly from repeated random interactions between the agent and its response environment (resolve Remark 6.3). One advantage is that the TD method is naturally implemented in an online, fully incremental fashion such that it can capture the stochastic behavior of an environment without an explicit model of the environment's dynamics [124]. The entropy-constraint filter can help significantly reduce the state/action space by eliminating the nodes with lower entropy (less uncertainty), such that the learning speed can be improved (resolve Remark 6.4).

Mathematically, to find an optimal solution of (6.11), NESarsa learns and updates the state-action value q for policy p recursively along with the interactions using

$$q_{t+n}(s_t, a_t) = q_{t+n-1}(s_t, a_t) + \alpha(s_t, a_t)[J_{t:t+n} - q_{t+n-1}(s_t, a_t)] \tag{6.12}$$

for $0 \leq t < T$, where T is the last step of an episode, $n \geq 1$ is the number of steps and $\alpha(s,a) \in (0,1]$ is the learning rate. We set $\alpha(s,a)$ the reciprocal of the number of occurrences of (s,a). As such, the agent can update $q(\cdot)$ values largely depending on new information at the beginning due to no prior knowledge and adjust it slowly as more information available. The subscripts on $J_{t:t+n}$ indicate a truncated expected return for t using rewards up until $t+n$, instead of the full return shown in (6.8),

$$J_{t:t+n} = \begin{cases} r_{t+1} + \cdots + \gamma^{n-1} r_{t+n} + \gamma^n E_{t+n-1}(s_{t+n}) & t+n < T \\ J_t & t+n \geq T \end{cases} \tag{6.13}$$

and $E_t(s)$ is the expected reward return of state s.

$$E_t(s) = \sum_{s \in \mathcal{A}(s)} p(a|s) q_t(s,a) \tag{6.14}$$

In order to find optimal $q(\cdot)$ values for $\{(s,a) : s \in \mathcal{S}, a \in \mathcal{A}\}$ using limited computational resources within a certain time, we propose an entropy-ϵ-greedy policy $p(\cdot|\cdot)$. Here we filter out nodes carrying less information (small entropy on $\mathcal{G}$) and focus on the exploration and exploitation of relatively unpredictable nodes. The ϵ-greedy ensures that we not only exploit an action that has maximal estimated q value (greedy action), but also explore alternative actions with probability ϵ (non-greedy actions). Therefore, every action a has a chance of being executed in each state s, i.e., $p(a|s) > 0$. As illustrated in Alg.8, given a state s and action candidates $\mathcal{A}(s)$, we create a new set of candidates $\mathcal{A}_{\mathrm{H}}$ by eliminating those whose entropy $\mathrm{H}(\cdot)$ is lower than a given threshold h_o. We then give the minimal probability $\epsilon/|\mathcal{A}_{\mathrm{H}}|$ of selection for all non-greedy actions and the remaining bulk of the probability $1 - \epsilon + \epsilon/|\mathcal{A}_{\mathrm{H}}|$ to the greedy action. Note that $\sum_{a \in \mathcal{A}(s)} p(a|s) = 1$. Algorithm 9 summarizes the learning procedure of an agent through the interactions with its environment. The algorithm terminates when changes on q values are less than a very small number. It is worth noting that this learning converges very fast (often seconds) depending on $\mathcal{X}_0$. The

output state-action values $q(\cdot)$ are then used for location selection: given a state s, an action a with maximum $q(s, a)$ value will be selected.

Algorithm 8 The Entropy-ϵ-greedy policy

Input state-action values $q(\cdot)$, very small number $\epsilon > 0$, maximum number of samples at a time θ and state s.
Output policy $p(\mathcal{A}(s)|s)$

Initialize $p(a|s) = 0$ for $\forall a \in \mathcal{A}(s)$; $\mathcal{A}_{\mathrm{H}} = \emptyset$
for $a \in \mathcal{A}(s)$ **do**
    if $\mathrm{H}(a) > h_o$ **then** $\mathcal{A}_{\mathrm{H}} = \mathcal{A}_{\mathrm{H}} \cup \{a\}$ **end if**
end for
$a^* = \arg\max_{a \in \mathcal{A}_{\mathrm{H}}} q(s, a)$
for $a \in \mathcal{A}_{\mathrm{H}}$ **do**
    $p(a|s) = \epsilon / |\mathcal{A}_{\mathrm{H}}|$
    if $a = a^*$ **then** $p(a|s) = p(a|s) + 1 - \epsilon$ **end if**
end for

6.5 Experimental Study

To evaluate AquaEIS approach in practical settings, we implement a functional service for fault source identification that integrates and coordinates multiple sources and entities. This includes real data (from utilities), commercial-grade models (WNTR simulator and EPANET, TEVA-SPOT and CANARY domain models), and policies set by the US EPA in actual settings. We begin by describing the experimental setup and compare the proposed approaches with several existing baselines. Finally we evaluate the effectiveness of AquaEIS using a wide variety of contamination events.

6.5.1 Experimental Setup

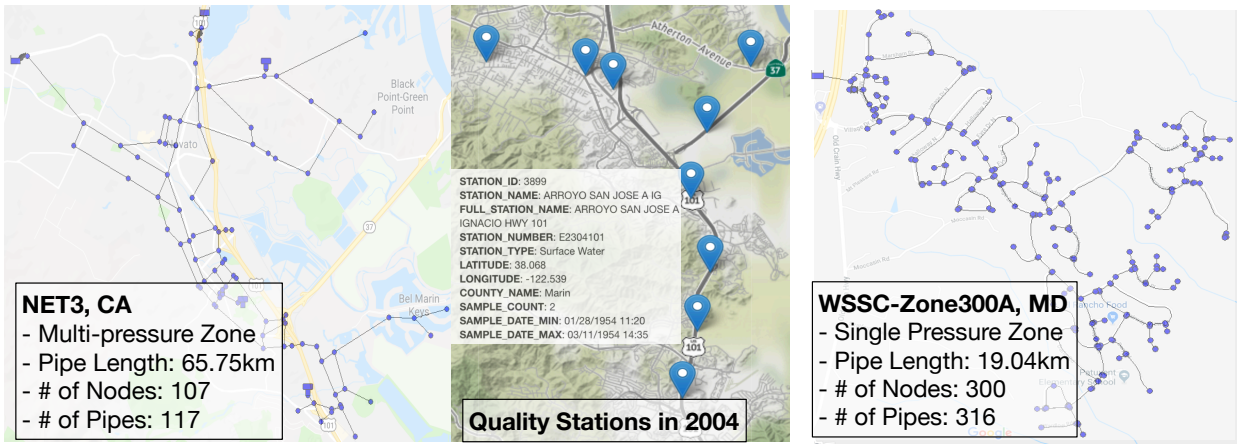
Water Network Model - Figure 6.8 shows two real-world water networks. NET3 (from US EPA) covers the Novato, CA water service area serving 78,823 people operated by

Algorithm 9 The online NESarsa based learning

Input a set of sources $\mathcal{X}_0$, entropy-ϵ-greedy policy $p(\cdot|\cdot)$ (Alg.8) and number of steps n.
Output optimal state-action values $q(\cdot)$

Initialize $q(\cdot) = \text{dict}()$
while stop criterion is not satisfied **do**
    Initialize state $s_0 = \mathcal{G}_0 = (\mathcal{X}_0, \mathcal{Y}, \mathcal{E})$; $T = \infty$
    Select action $a_0 \sim p(\mathcal{A}(s_0)|s_0)$
    for $t = 0, 1, 2, ...$ **do**
        if $t < T$ **then**
            Take action a_t; compute state s_{t+1} via Alg.7
            if s_{t+1} is terminal **then**
                Compute reward r_{t+1} via (6.7); $T = t + 1$
            else
                $r_{t+1} = -1$; select $a_{t+1} \sim p(\mathcal{A}(s_{t+1})|s_{t+1})$
            end if
        end if
        $\tau = t + 1 - n$
        if $\tau \geq 0$ **then** Update $q(s_\tau, a_\tau)$ via (6.12-6.14) **end if**
        if $\tau = T - 1$ **then** Break **end if**
    end for
end while

North Marin Water District (NMWD). WSSC-SUBNET (from WSSC) is a subnet of Prince George's County in Maryland serving $1,137$ people operated by Washington Suburban Sanitary Commission (WSSC). These two networks have different characteristics in terms of number of components, pipe properties (diameter, length, etc.), flow patterns and so forth. The average time of contaminant transport from one node to another is around 24min (NET3) or 15min (WSSC-SUBNET). The mean values can be interpreted in the following way: if a node is randomly selected as the contaminant source, one could expect that, depending on the sensor configuration, it needs at least this amount of time to have a new set of measurements. To discriminate the onset of either an anomalous event or natural variations in water quality, the event detection system CANARY needs to examine multiple outliers within a prescribed window [55]. We therefore set the detection window to 75min for NET3 and 45min for WSSC-SUBNET to allow for multiple readings. Once a contamination is confirmed by CANARY, collected data from past 75min (NET3) or 45min (WSSC-SUBNET)



(a) NET3 and its stations config in 2004 (b) WSSC-SUBNET

Figure 6.8: Two real-world water networks: (a) a multi-pressure zone of NMWD; (b) a single pressure zone of WSSC.

will be used to perform the event identification. The number of measurements depends on the sampling frequency of sensing devices.

In-situ Sensor Configuration - In practice, most water systems are instrumented with a sparse set of sensors with new measurements every 15min. For example, according to California Department of Water Resources, in 2004 NET3 had only 9 quality sensors of 107 water nodes (Fig. 6.8a). The water quality sensor is modeled with contaminant specific detection thresholds and only provides yes/no indications of contamination. In our experiments, the maximum number of sensors is set to 10% of the number of nodes in a network and they provide binary measurements. In this paper, sensors are assumed to perform with 100% accuracy (i.e., no failure) and their locations are generated by TEVA-SPOT. The deployment of different number of sensors for both networks are shown in Fig. 6.9.

Contamination Events - They are specified by the location at which the contaminant is introduced into the network and the time of its introduction. Given that it is difficult to predict a specific contamination incident, a large collection of distinct events are used to enhance the validity of results. Specifically, we consider the scenario in which a contaminant is injected over a 24-hour period starting at anytime of the day and every node in the network

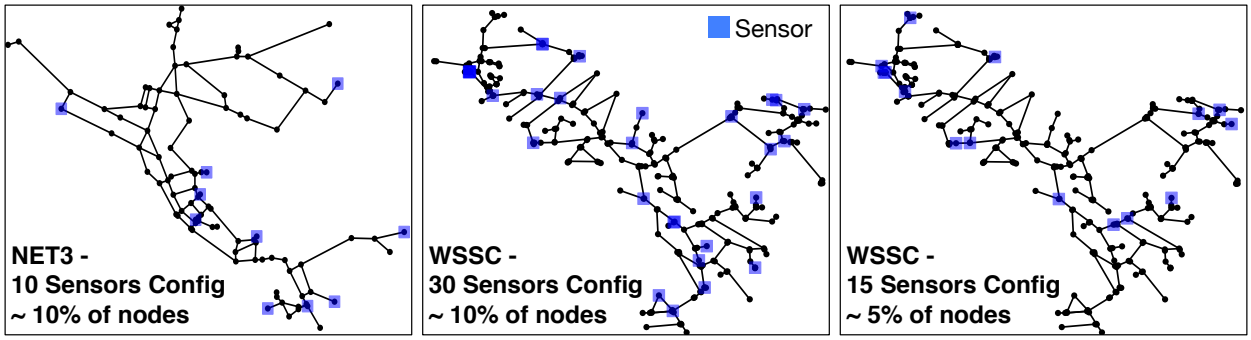


Figure 6.9: In-situ sensor configurations on NET3 and WSSC-SUBNET, generated using TEVA-SPOT.

is considered a potential point of entry. This ensemble of events is generated using WNTR simulator with EPANET water quality model.

6.5.2 Evaluating the Location Inference Step

Methods to compare - We first validate our HMM formulation and show the effectiveness of the proposed particle filter based inference. Two existing baselines are compared: Bayesian probability formulation [83] and contaminant status algorithm (CSA) [111]. The Bayesian formulation calculates the probability of a node being the true source based on the deviation between simulated values and measurements, while CSA assigns a status to each node as either safe, unsafe or unknown by tracking the flows over time. We refer to these methods as Baseline-BP (Bayesian probability), Baseline-CSA, HMM-FB (forward-backward algorithm) and our proposed HMM-PF (particle filter based inference).

Performance metrics - Ideally, we want to identify a set of tractable number of sources, such that field staff can directly inspect the possible locations and confirm the true source. However, due to the detectability issue of in-situ sensors (Remark 6.2), a large set of sources ($PS = \{v : v \in \mathcal{V}\}$) is likely to be identified. This estimate, as the input into location refinement, should include the true source. Otherwise, it may misguide the further processing towards a wrong decision. Therefore, we evaluate the approach by checking if its resulting

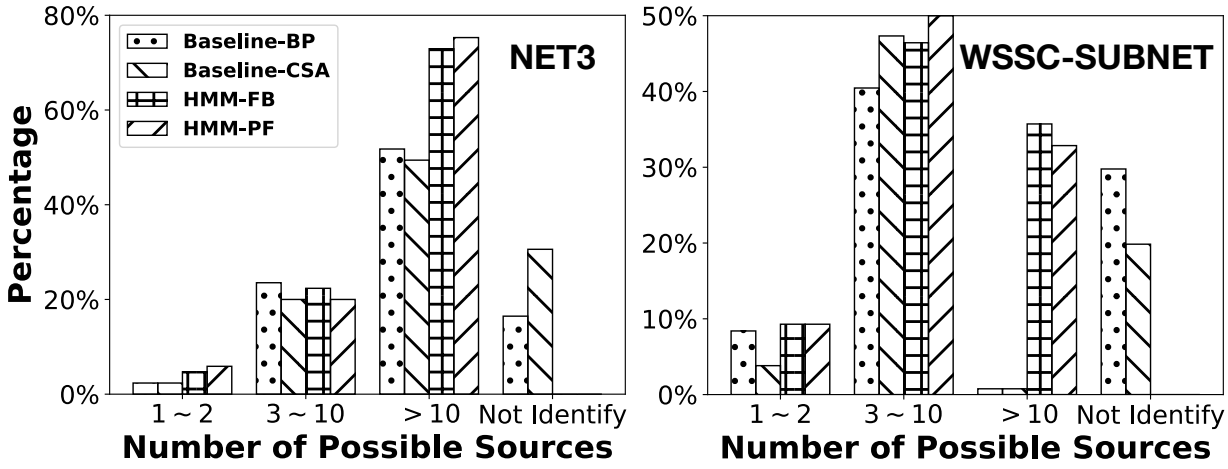
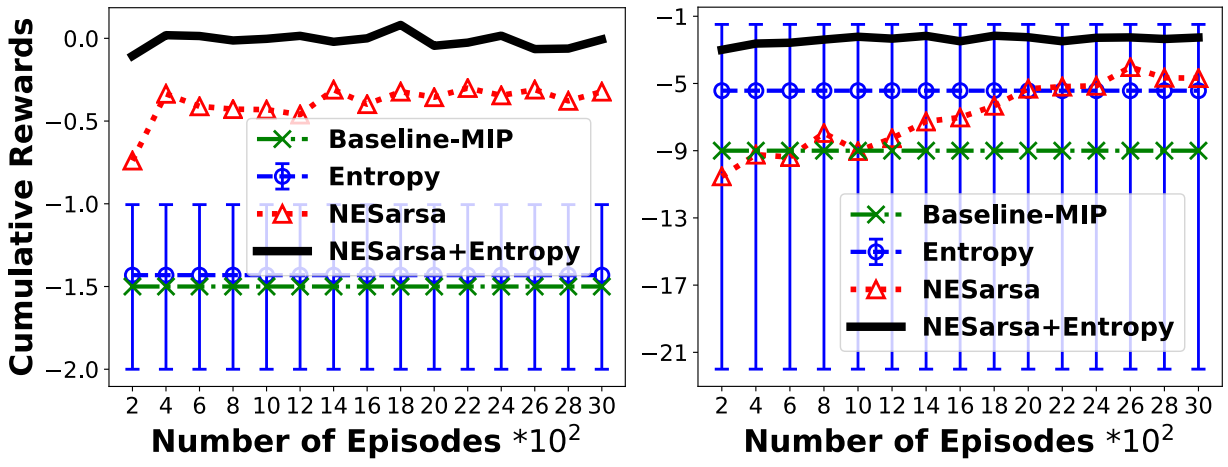


Figure 6.10: Comparison on distributions of number of identified sources by 2 baselines and 2 HMM based methods on NET3 and WSSC-SUBNET water networks.

set PS includes the true source and the smaller the size $|PS|$, the more effective approach.

Figure 6.10 shows the distribution of number of possible sources identified by the four methods for an ensemble of $|\mathcal{V}| = 107$ scenarios on NET3 and $|\mathcal{V}| = 300$ scenarios on WSSC-SUBNET (one for each node in the network with an arbitrary injection time). The results show that 2 baseline methods are not able to identify all true sources, where Baseline-CSA fails to identify 31% scenarios on NET3 and Baseline-BP fails to identify 30% on WSSC-SUBNET. HMM-FB and HMM-PF have better, similar performance in the sense that their resulting sets include the true source for 100% scenarios, however they are not small enough for effective inspection. For example on NET3, around 70% scenarios result in more than 10 possible sources by both approaches. It also shows that the number of identified sources can vary considerably depending on the true event location. This confirms the necessity of integrating additional information to refine the estimate. On average, the running time of HMM-PF is 3 times faster than HMM-FB (Remark 6.1), where we use $10,000$ and $30,000$ particles for NET3 and WSSC-SUBNET separately. Overall, HMM-PF yields a better performance, which will be used in the following experiments.

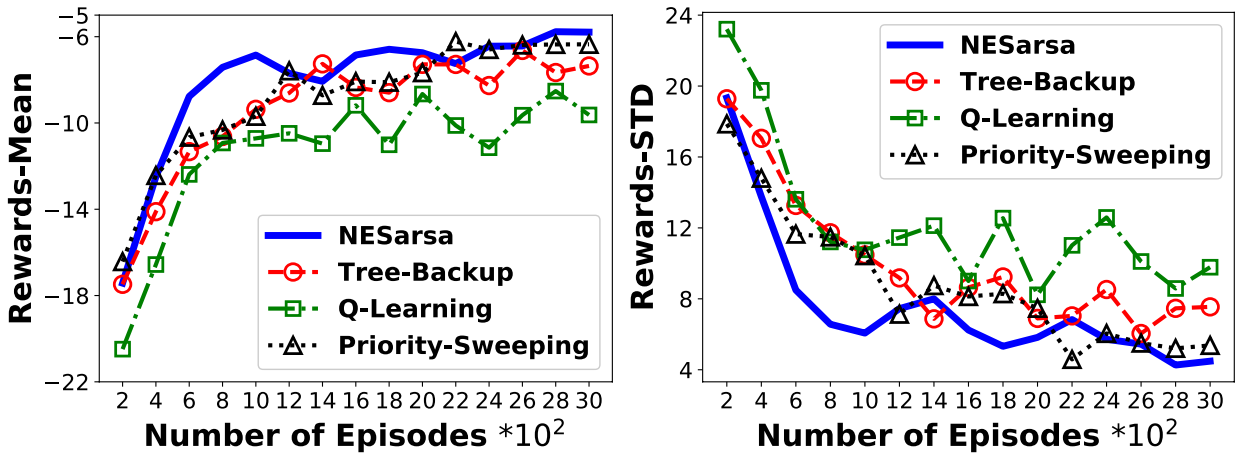


(a) $|PS| = 4, \theta = 1$ (b) $|PS| = 25, \theta = 2$

Figure 6.11: Cumulative rewards versus number of episodes on NET3. Comparisons on average rewards with standard deviation error bars of multiple approaches, given different sizes of sources $|PS|$ and maximum numbers of samples θ.

6.5.3 Evaluating the Location Refinement Step

Methods to compare - To validate the feasibility of RL based design, we compare our proposed NESarsa based approaches with an existing Mixed Integer Programming formulation [142] (Baseline-MIP) and a pure entropy approach (Entropy). Baseline-MIP selects sampling locations that maximize the total pair-wise distinguishability of possible sources, while Entropy selects a set of locations that has maximum entropy on aforementioned source-to-node graph with ties broken arbitrarily. To further show the effectiveness of our approach, we implemented 3 other RL methods including Tree-Backup, Q-Learning and Priority-Sweeping for comparison. Tree-Backup and Q-Learning are off-policy algorithms with different learning strategies, while NESarsa is an on-policy algorithm. The key difference is that off-policy methods instead use two separate policies for learning and making decisions to enhance the agent's exploratory behavior. Though more powerful and general, they are often of greater variance and slower to converge [124]. In Priority-Sweeping, rather than directly learning the state-action values, it learns a model of the environment and uses this model to plan an action.



(a) Mean Value

(b) Standard Deviation

Figure 6.12: Comparisons on (a) mean value and (b) standard deviation of cumulative rewards of multiple RL algorithms.

Performance metrics - Given an intractable number of possible sources, it is desirable to locate the true source in less number of sampling cycles. We use the cumulative reward defined in (6.8) for evaluation. Because it takes into account both the number of sampling cycles (measure of cost) and the distance between the resulting set and the truth (measure of accuracy). The higher the reward, the better the result. Also, the speed of convergence reflects a measure of time.

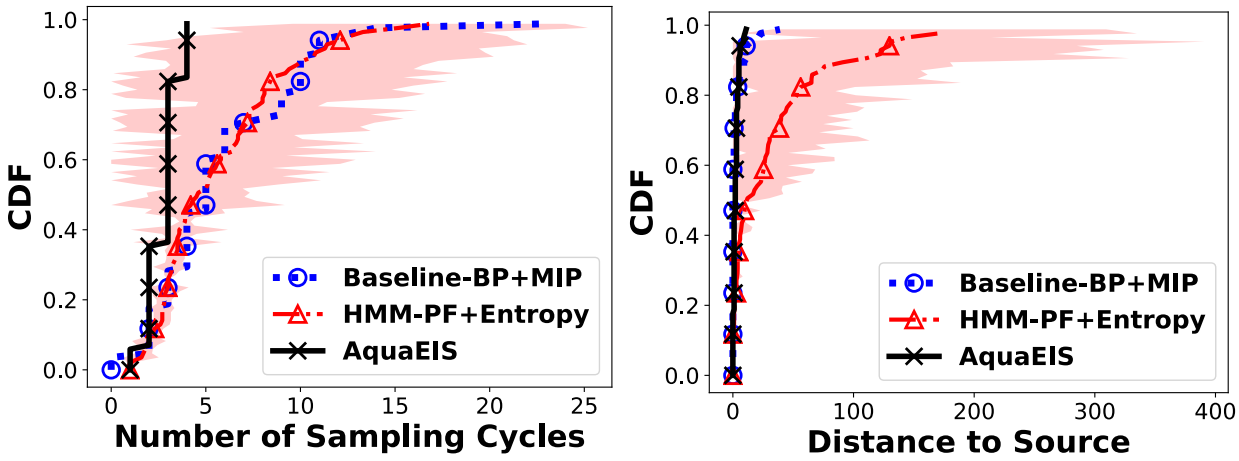
Figures 6.11 show the comparison of multiple methods on average cumulative rewards over 10 simulation runs versus 3000 episodes. For the readability of figures, average values of every 200 episodes are shown. Recall that an *episode* is a sequence of decisions and their rewards (Sec.6.4.3). Note that Baseline-MIP and Entropy methods do not depend on and thus do not change with the number of episodes.

In Fig. 6.11a, 4 possible sources are identified and the maximum number of grab samples at a time is set to 1. Baseline-MIP achieves an average performance of Entropy that shows a large variance. As hinted in Sec.6.4.1, multiple (combinations of) nodes can have same maximum entropy due to the hydraulic behavior by nature, which can introduce the randomness in pure Entropy method. Two NESarsa (+Entropy) based approaches converge to higher rewards in

200 episodes, which may not be shown clearly in figures due to the scale of y-axis. Because they are able to learn the stochastic environment (Remark 6.3) with delayed rewards and have a global perspective on locations from which grab samples can enhance the "long-term" benefit. NESarsa+Entropy is a combination of NESarsa and Entropy-ϵ-greedy policy. In contrast, Baseline-MIP and Entropy suffer from the issue of sequential decisions due to their preferences of local optimal locations. They perform a single optimization analysis that may not contribute to the future decisions. NESarsa+Entropy achieves the best among others with a higher reward and fast convergence (i.e., better performance on cost-accuracy-time tradeoffs). It resolves the problem of entropy ambiguity through trial-and-error learning and overcomes the problem of exponentially large state/action space (Remark 6.4) by targeting and exploring informative nodes using Entropy-ϵ-greedy policy.

In Fig. 6.11b, with 25 possible sources and 2 maximum number of samples, Entropy shows a mean value of -6 with considerable fluctuation, Baseline-MIP yields a stable but lower reward, while NESarsa (+Entropy) converge to the higher rewards (> -6). Compared with Fig. 6.11a, their rewards all decrease, because the increasing number of possible sources results in a larger set of potential contaminated nodes, which requires more sampling cycles to locate a source. Due to the same reason, NESarsa based approaches need more explorations, thus more number of episodes, to converge.

To further provide the insight into our proposed mechanism, Figures 6.12 illustrate that NESarsa converges faster to a higher reward in comparison with the other 3 techniques. This is possibly because the uncertain environment and large state/action space make it hard for Tree-Backup and Q-Learning to explore using two policies and for Priority-Sweeping to learn an explicit model. It is worth noting that the fluctuation tails appearing in all methods are due to the exploration strategy (ϵ-greedy policy) and the performance will be stable once the state-action values are fixed.



(a) Number of sampling cycles

(b) Distance to source

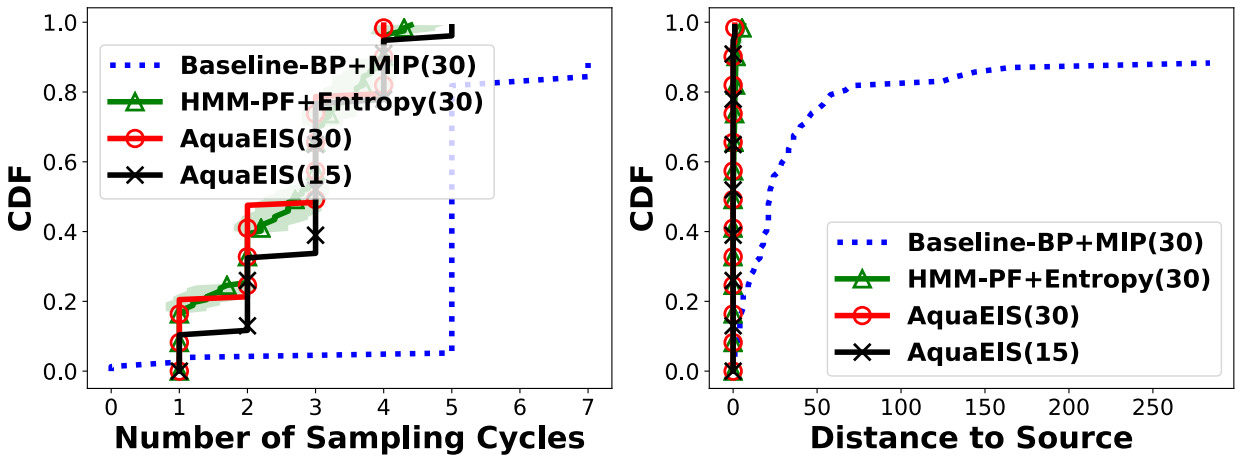
Figure 6.13: NET3 - Cumulative distribution function for (a) number of sampling cycles and (b) distance to the source.

6.5.4 Evaluation of the end-to-end AquaEIS Approach

Methods to compare - Viewed together, we now evaluate the performance of our AquaEIS approach for a complete ensemble of contamination events. We compare it with an existing workflow (a combination of Baseline-BP and Baseline-MIP (Baseline-BP+MIP)) implemented in Water Security Toolkit [44] and a combination of HMM-PF and pure Entropy (HMM-PF+Entropy) to further validate our design.

Performance metrics - An efficient event identification system needs to locate the source in a timely manner. We use the number of sampling cycles as a measure of cost and time, and the distance to the true source (shortest path in terms of number of links) as a measure of accuracy. Less number of cycles with a shorter distance indicates a better performance.

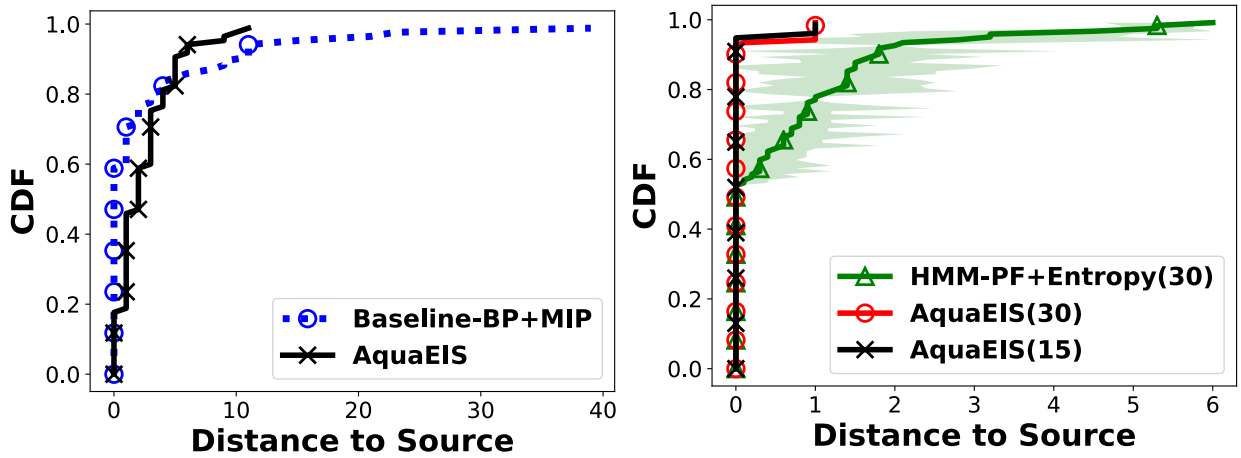
In Fig. 6.13/6.14, each cumulative distribution function (CDF) is computed for a large ensemble of contamination scenarios. The scenario includes a single source at a random node in the network starting at an arbitrary time. Particularly, an ensemble of 300 and 1000 distinct scenarios with different configurations is analyzed on NET3 and WSSC-SUBNET separately. The plots in Fig. 6.13 illustrate the statistical effect of varying the scenario on the



(a) Number of sampling cycles

(b) Distance to source

Figure 6.14: WSSC-SUBNET - Cumulative distribution function for (a) number of sampling cycles and (b) distance to the source with 15 (5% of nodes) and 30 (10% of nodes) sensors.



(a) Zoom in of Fig. 6.13b

(b) Zoom in of Fig. 6.14b

Figure 6.15: For readability, zoomed-in figures of results of (a) NET3 and (b) WSSC.

performance of approaches in identifying the source, where the maximum number of samples in each round is set to 2. In Fig. 6.13a, AquaEIS is able to identify 100% scenarios in less than 5 sampling cycles, while the other 2 methods can only identify 38% on average. HMM-PF+Entropy shows a large variance on those scenarios where more rounds of samplings are needed. With regard to the distance to the true source (Fig. 6.15a), Baseline-BP+MIP locates the true source (distance = 0) for 60% scenarios, however, cannot identify a source (distance > 20) for 5%. AquaEIS can identify a node with distance < 10 for 100% and < 5 for 90%.

Fig. 6.14/6.15b shows similar results on WSSC-SUBNET, where AquaEIS performs better with regard to the number of sampling cycles and distance to the source. Here the maximum number of samples in each round is set to 1 because of the relatively smaller service area. Additionally, it shows the influence of the number of sensors. In particular, the CDFs are compared for the cases of 15 (5% of nodes) and 30 (10% of nodes) sensors. AquaEIS with less sensors (i.e., AquaEIS(15)) needs more rounds of samplings but can generate same performance on distance-to-source. The increasing number of human-driven grab samples causes the rise of the cost involved, while on the other hand preserves the accuracy as a complementary data source. AquaEIS(15) can exactly locate the true source for 95% scenarios with less than 5 sampling cycles. Furthermore, among all the validated scenarios, solutions using AquaEIS are obtained within a few seconds or minutes, making it a viable approach for real-time event identification.

6.6 Chapter Summary and Discussion

In this chapter, we presented the design and evaluation of AquaEIS, a middleware support for an efficient event identification in a complex distributed setting (i.e., water infrastructures), which can optimize cost-accuracy-latency tradeoffs. AquaEIS integrates and coordinates

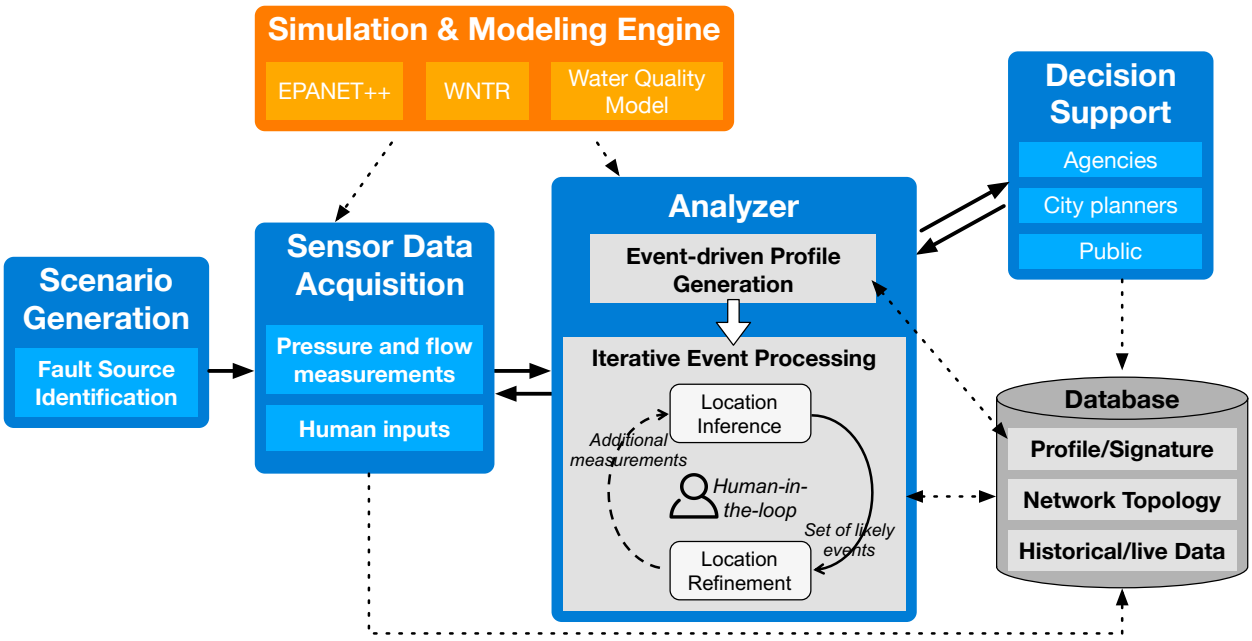


Figure 6.16: This chapter leverages the AquaSCALE system architecture for fault source identification using human-in-the-loop based sensing.

multiple sensing modalities (devices, human-as-a-sensor) and domain knowledge; algorithms for computation and messaging are implemented using custom modules (Remarks 6.1-6.4); recommendations for grab-sampling are generated in real-time. The validation studies presented are performed on an actual implementation with realistic data - our results indicate that we can narrow down sources of failures rapidly enough to allow for utility intervention.

We leverage the AquaSCALE system architecture (Fig.6.16) to improve the resilience of water infrastructures through intelligently guiding human participants in identifying contaminant sources. To capture the physical nature of spatial-temporal variations of water quality, we first use domain model and hydraulic simulator to generate the profile/signature of contamination events that will be used as the apriori knowledge. The proposed event processing, combined with simulation and modeling engines, is executed in an iterative manner on-the-fly, based on the information collected from Data Acquisition module. If additional measurements are needed, optimal grab-sampling locations will be selected and human-driven grab sample will be triggered to collect more information. As time progresses,

more measurements become available and this observation, analysis and adaptation loop will be performed iteratively until the fault source is identified. Together, AquaEIS, built upon AquaSCALE, provides an integrated view of the vulnerability of water CPHI systems to a wide variety of contamination events, and the ability to reduce this vulnerability through efficient event processing.

Chapter 7

Conclusion

7.1 Summary of Thesis

In this thesis, we propose an integrated approach, **AquaSCALE**, to enabling resilience in cyber-physical-human water infrastructures. AquaSCALE can be used by water agency operators with expertise in civil infrastructure to explore problems and solutions in cyberspace before instantiating them into a physical infrastructure. For example, a smaller section of water systems (compared with today usually an entire pressure zone) can be shutdown to prevent cascading failures of pipe burst and to preserve critical water supplies. Such exploration, proactive planning and their effective instantiation during damage/shutdown is relevant in global contexts.

Our approach integrates multiple sensing modalities including IoT devices, human-as-a-sensor and external observations, combined and enhanced with domain-specific models and simulations and machine learning methods, to quickly and accurately identify failures (i.e., pipe breaks and contamination events). The composition of techniques results in cost-accuracy-latency tradeoffs in fault identification, inherent in CPHIs due to the constraints imposed by cyber components, physical mechanics and human operators. We explore three key resilience problems, which are isolation of multiple concurrent faults under a small number of pipe failures in Chapter 4; state estimation of the water systems under extreme events (e.g., earthquakes) in Chapter 5, and contaminant source identification in water networks using human-in-the-loop based sensing in Chapter 6. The effectiveness of our approach is demonstrated through extensive performance evaluations on real-world water networks. This is a key step to enable the next generation water systems, which will not be passive water delivery systems but active highly-distributed event-based control systems [7].

7.2 Future Challenges

As we consider developing a comprehensive middleware solution to explore resilience of community-scale infrastructures, which incorporates the techniques proposed in this thesis, many other challenges remain open. We outline several future directions identified through the study.

7.2.1 Incorporation of Formal Method

In this thesis, the integration of multiple sensing modalities is combined and enhanced with domain specific models and simulations, and machine learning (ML) techniques. However, this has presented two key limitations. Modeling and simulating the operation of community-scale infrastructures is difficult and time consuming, and safety assurance and verification methodologies for machine learning are very challenging. In water infrastructures, the complexity of hydraulic behavior is due to the supply-demand variations, operations to conserve the resources and a wide-range of disruptions. Consequently, domain models require precise and reliable mathematical models of hydraulics and a significant amount of computer memory and computational resources to accommodate the large-scale spatial and temporal data. In addition, we need to provide guarantees that the water CPHI system is behaving correctly, which is necessary in safety-critical contexts. Unfortunately, ML methods in some cases cannot support the formal verification. Therefore, we propose to integrate with formal method strategy to address the aforementioned limitations.

Formal methods are rooted in logic and reasoning using patterns and place-holders rather than specific values, which thus allows representation and exploration of large, possibly infinite state space. We initiate this effort by designing the logic+physics+learning approach (Fig. 7.1) to capture the hybrid architecture and dynamic behavior of CPHI systems at

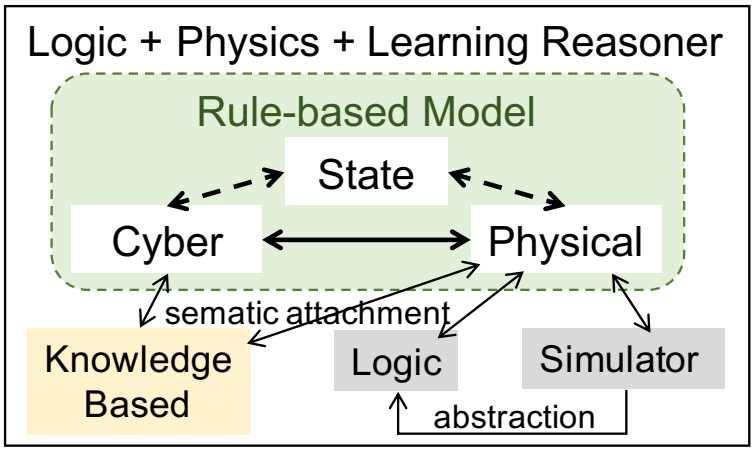


Figure 7.1: Logic and physics interaction.

multiple levels of abstraction, and will include formalization of requirements and objectives. Our models for physical elements will include features in the underlying network such as storages, sensors, communication resources, and environmental effects. Models will be constructed as a modular combination of physical and logical aspects using a knowledge-based representation. This naturally supports integration of logical reasoning tools such as Maude [31] and physics-based simulators such as EPANET and WNTR for hydraulic and quality modeling tools. Being knowledge based, models can link to databases and adapt to real-time information.

In our initial experience, models are formalized in rewriting logic using the Maude rewriting logic system [31]. Using only the structure of a model and the logical constraints, we can already do interesting effect analysis in a timely manner - if something happens in one element where might the effects propagate. For example, the below code snippet is a hydraulic rule written using Maude - if the pressure head of node 2 is in low category, the status of link 9 will switch from close (0) to open (1) to distribute water from the tank.

```
crl [control] :
openlink(9)
<9 : LINK | status : 0> => <9 : LINK | status : 1>
```

```
if pressure(2) is low
```

Although this gives over-approximations compared with domain-specific modeling (e.g., low pressure category instead of the specific value), it generates a practical solution in time, which can guide detailed modeling, processing and testing. In addition, using logical specification of physics, one can get a next level of information from the model and carry out (bounded) model checking to search for conditions where key invariants or constraints fail or policies are violated. For example, some based invariants include pressure must be within a certain range, or water quality meets a given standard (depending on concerns).

$$\text{pressure} \in [a, b]$$

$$\text{quality meets a given standard}$$

A formal model that incorporates constraints can be queried to determine if the system sufficiently satisfies multiple requirements/invariants given thresholds and priority mechanisms.

Overall, we are interested in applying formal methods to decision-making support, which will involve some combination of physics, logic-based reasoning and machine learning components. This enables the ability to directly and more efficiently get states and actions that are weighted highly, and in the meanwhile avoid the property violation.

7.2.2 Community Data Exchange Framework

Community-scale infrastructures can be viewed as networks of systems and processes, which produce and distribute continuous flow of essential services to end-users as interdependent and connected. For example in Fig. 7.2, in order to offer sustainable water service to both utilities and customers, it requires both the power distribution network for pump control and the communication infrastructure for SCADA system. Individual failure can have wide-ranging consequences by propagating from one system to the next due to these in-

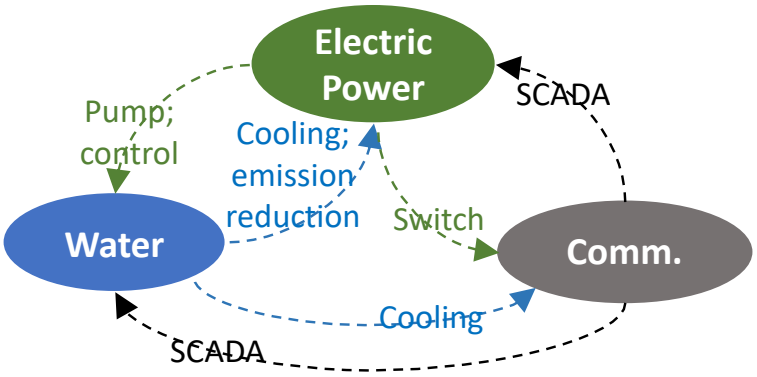


Figure 7.2: Interdependent urban infrastructures.

terconnectivities. Unfortunately, today these infrastructures are suffering from the issue of fragmented management, where they are maintained and operated in a distributed way by different agencies. These agencies have solid expertise, but limited channels for coordination. What is required is a structured approach to view the multiple infrastructures in a unified manner to understand their interactions, quantify the benefits and assess vulnerabilities. Given the emergence of IoT and big data systems, a data-driven approach to allow to exchange historical and live data of relevance to the stakeholders (e.g., agencies, policy makers and citizens) is feasible. Build on our previous experience, we believe that data about the infrastructure (historical and live measurements) and its dependencies, a bulk of which resides within community agencies, if combined and enhanced with other geo-distributed data sources can enable new levels of efficiency and resilience.

7.2.3 Protect Infrastructures from Cyber Attacks

As cyber technologies are increasingly added to existing infrastructures and built into newly constructed infrastructure, potential cyber attack would be a major concern. In fact, great concern has been expressed regarding the security of SCADA systems in the light of the breach that occurred at Maroochy Water Services in Queensland, Australia, in 2000 [116, 106]. The Maroochy Shire Council experienced communication problems in their wastewater

treatment plant, where their pump stations were not working properly. After three months, an engineer finally discovered that someone was hacking into the system and deliberately causing the problems. At that time, one million liters of untreated sewage had been released into a stormwater drain from where it flowed into local waterways.

SCADA systems are widely used for monitoring and controlling power distribution facilities, oil and natural gas pipelines, water distribution systems and wastewater treatment plants. These systems are generally located in the main plant and communication with sensors and control devices to support the operation of lifeline services. However, even with the cybersecurity technology, these types of systems have proven to be vulnerable to cyber attacks [30]. In 2015, the U.S. Department of Homeland Security (DHS) reported 25 cyber attacks in water systems and 46 incidents in power systems. Comparatively, the reported number of attacks in water systems increased by 78.6% from 2014 to 2015. In addition, since the event of September 11, 2001, the U.S. government has been forced to consider the vulnerability of all of the nation's critical infrastructure to terrorist attack. Under these conditions, safeguarding critical infrastructure components is of utmost importance.

Bibliography

[1] Freezing and bursting pipes. *Natural Hazard Mitigation*. A publication of the Institute for Business and Home Safety.

[2] https://www.unwater.org/water-facts/scarcity/. *United Nations - Water*.

[3] Earthquakes a danger to los angeles water supply: https://www.govtech.com/em/disaster/earthquakes-danger-los-angeles-water-supply.html. *Government Technology (gt)*, 2014.

[4] Harsh winter tests u.s. water infrastructure. *American City & County*, 2014.

[5] Community resilience planning guide for buildings and infrastructure systems. *National Institute of Standards and Technology*, 1, 2015.

[6] Fixing wmata: https://www.governing.com/topics/transportation-infrastructure/gov-wmata-transit-problems.html. *GOVERNING*, 2016.

[7] Nec and imperial college london smart water project: https://wp.doc.ic.ac.uk/aese/project/smart-water/. In *Cyber-Physical Systems for Smart Water Networks*, 2016.

[8] Economic impact: https://www.infrastructurereportcard.org/the-impact/economic-impact/. *American Society of Civil Engineers (ASCE)*, 2017.

[9] Infrastructure report card: https://www.infrastructurereportcard.org/. *American Society of Civil Engineers (ASCE)*, 2017.

[10] Annual report: https://www.wsscwater.com/annualreport. *Washington Suburban Sanitary Commission (WSSC)*, 2018.

[11] https://www.un.org/development/desa/en/news/population/2018-revision-of-world-urbanization-prospects.html. *United Nations*, 2018.

[12] Nyc blackout: https://www.cnn.com/2019/07/30/us/nyc-blackout-con-ed-explanation-trnd/index.html. *CNN*, 2019.

[13] W. Abbas, L. S. Perelman, S. Amin, et al. An efficient approach to fault identification in urban water networks using multi-level sensing. In *Embedded Systems for Energy-Efficient Built Environments*, pages 147–156. ACM, 2015.

[14] C. Ai, H. Zhao, R. Ma, and X. Dong. Pipeline damage and leak detection based on sound spectrum lpcc and hmm. In *Sixth International Conference on Intelligent Systems Design and Applications*, volume 1, pages 829–833. IEEE, 2006.

[15] M. Allen, A. Preis, et al. WDS monitoring and decision support using a wireless sensor network. In *Software Eng., Artificial Intelligence, Networking & Parallel/Distributed Computing*, pages 641–646. IEEE, 2013.

[16] M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *IEEE Trans. on signal proc.*, 2002.

[17] L. Begnudelli and B. Sanders. Simulation of the St. Francis dam-break flood. *Journal of Eng. Mechanics*, 133(11):1200–1212, 2007.

[18] L. Begnudelli and B. Sanders. Adaptive Godunov-based model for flood simulation. *Journal of Eng Mechanics*, 134(6):714–725, 2008.

[19] K. Benson, C. Fracchia, G. Wang, Q. Zhu, S. Almomen, J. Cohn, L. Darcy, D. Hoffman, M. Makai, J. Stamatakis, et al. Scale: Safe community awareness and alerting leveraging the internet of things. *IEEE Communications Magazine*, 53(12):27–34, 2015.

[20] K. E. Benson, Q. Han, K. Kim, P. Nguyen, and N. Venkatasubramanian. Resilient overlays for iot-based community infrastructure communications. In *2016 IEEE First International Conference on Internet-of-Things Design and Implementation (IoTDI)*, pages 152–163. IEEE, 2016.

[21] J. Berry, E. Boman, L. Riesen, W. E. Hart, C. A. Phillips, J. Watson, and U. EPA. Teva-spot toolkit and user's manual. *U.S. EPA*, 2012.

[22] J. Berry, R. D. Carr, W. E. Hart, V. J. Leung, C. A. Phillips, and J.-P. Watson. Designing contamination warning systems for municipal water networks using imperfect sensors. *Journal of Water Resources Planning and Management*, 135(4):253–263, 2009.

[23] J. Berry, W. E. Hart, C. A. Phillips, J. G. Uber, and J.-P. Watson. Sensor placement in municipal water networks with temporal integer programming models. *Journal of water resources planning and management*, 132(4):218–224, 2006.

[24] J. Berry, H. Lin, E. Lauer, and C. Phillips. Scheduling manual sampling for contamination detection in municipal water networks. In *ASCE Proc. Water Dist. Systems Analysis Symposium*, 2006.

[25] S. Bradford and B. Sanders. Finite-volume model for shallow-water flooding of arbitrary topography. *Journal of Hydraulic Engineering*, 128(3):289–298, 2002.

[26] U. Brandes and D. Fleischer. Centrality measures based on current flow. In *Theoretical aspects of computer science*, pages 533–544. Springer, 2005.

[27] C. Cameron. Feds arrest al qaeda suspects with plans to poison water supplies. *FoxNews.com*, 2002.

[28] C. G. Cassandras. Smart cities as cyber-physical social systems. *Engineering*, 2(2):156–158, 2016.

[29] H. Chourabi, T. Nam, S. Walker, J. R. Gil-Garcia, S. Mellouli, K. Nahon, T. A. Pardo, and H. J. Scholl. Understanding smart cities: An integrative framework. In *2012 45th Hawaii international conference on system sciences*, pages 2289–2297. IEEE, 2012.

[30] R. M. Clark, S. Panguluri, T. D. Nelson, and R. P. Wyman. Protecting drinking water utilities from cyberthreats. *Journal of the American Water Works Association*, 109(INL/JOU-16-39302), 2017.

[31] M. Clavel, F. Duran, S. Eker, S. Escobar, P. Lincoln, N. Marti-Oliet, J. Meseguer, and C. Talcott. Maude manual. *Stanford Research Institute (SRI)*, 2016.

[32] R. T. Clemen and R. L. Winkler. Combining probability distributions from experts in risk analysis. *Risk analysis*, 19(2):187–203, 1999.

[33] A. F. Colombo, P. Lee, and B. W. Karney. A selective literature review of transient-based leak detection methods. *Journal of Hydro-environment Research*, 2(4):212–227, 2009.

[34] M. Cosovic and D. Vukobratovic. Distributed gauss-newton method for ac state estimation: A belief propagation approach. In *Smart Grid Communications*, pages 643–649. IEEE, 2016.

[35] T. M. Cover and J. A. Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[36] D. Crisan and A. Doucet. A survey of convergence results on particle filtering methods for practitioners. *IEEE Transactions on signal proc.*, 2002.

[37] S. L. Cutter, J. A. Ahearn, et al. Disaster resilience: A national imperative. *Science & Policy for Sustainable Development*, 55(2):25–29, 2013.

[38] A. K. Dixit, J. J. Sherrerd, et al. *Optimization in economic theory*. Oxford University Press on Demand, 1990.

[39] R. Du, C. Fischione, and M. Xiao. Flowing with the water: On optimal monitoring of water distribution networks by mobile sensors. In *IEEE INFOCOM 2016-The 35th Annual IEEE International Conference on Computer Communications*, pages 1–9. IEEE, 2016.

[40] L. Dueñas-Osorio, J. I. Craig, and B. J. Goodno. Seismic response of critical interdependent networks. *Earthquake engineering & structural dynamics*, 36(2):285–306, 2007.

[41] J. Eidinger and C. A. Davis. Recent earthquakes: implications for us water utilities. *Water Research Foundation*, xx, 2012.

[42] H. El Gamal and A. R. Hammons. Analyzing the turbo decoder using the gaussian approximation. *Information Theory*, 47(2):671–686, 2001.

[43] U. EPA. Response protocol toolbox: planning for and responding to drinking water contamination threats and incidents. 2004.

[44] U. EPA. Water security toolkit user manual. 2014.

[45] U. EPA. Enhancements to the epanet-rtx (real-time analytics). *US EPA Office of Research and Development*, 2015.

[46] J. Feng, M. Huang, L. Zhao, Y. Yang, and X. Zhu. Reinforcement learning for relation classification from noisy data. In *32 AAAI Conf. on Artificial Intelligence*, 2018.

[47] R. Fletcher and M. Chandrasekaran. Smartball: a new approach in pipeline leak detection. In *2008 7th International Pipeline Conference*, pages 117–133. American Society of Mechanical Engineers, 2008.

[48] F. V. Fomin, S. Gaspers, A. V. Pyatkin, and I. Razgon. On the minimum feedback vertex set problem: Exact and enumeration algorithms. *Algorithmica*, 52(2):293–307, 2008.

[49] F. Fusco and A. Ba. Fault diagnosis of water distribution networks based on state-estimation and hypothesis testing. In *Communication, Control, and Computing*, pages 886–892. IEEE, 2012.

[50] Y. Gao, M. Brennan, P. Joseph, et al. On the selection of acoustic/vibration sensors for leak detection in plastic water pipes. *Journal of Sound and Vibration*, 283(3):927–941, 2005.

[51] G. Germanopoulos. A technical note on the inclusion of pressure dependent demand and leakage terms in water supply network models. *Civil Engineering Systems*, 2(3):171–179, 1985.

[52] O. Giustolisi, D. Savic, and Z. Kapelan. Pressure-driven demand and leakage simulation for water distribution networks. *Journal of Hydraulic Engineering*, 134(5):626–635, 2008.

[53] A. D. González, L. Dueñas-Osorio, et al. The interdependent network design problem for optimal infrastructure system restoration. *Computer-Aided Civil & Infra Engineering*, 31(5):334–350, 2016.

[54] N. Gordon, B. Ristic, and S. Arulampalam. Beyond the kalman filter: Particle filters for tracking applications. *Artech House, London*, 2004.

[55] J. Hagar, R. Murray, T. Haxton, J. Hall, and S. McKenna. Using the canary event detection software to enhance security and improve water quality. *Proc. of Env & Water Resources Institute*, 2013.

[56] M. Henneberger. A nation challenged: suspects; 4 arrested in plot against u.s. embassy in rome. *The New York Times*, 2002.

[57] E. Hernadez, S. Hoagland, and L. Ormsbee. Water distribution database for research applications. In *World Environmental and Water Resources Congress*, pages 465–474, 2016.

[58] 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*, 34(3):173–183, 1999.

[59] Q. Huang, L. Shao, and N. Li. Dynamic detection of transmission line outages using hidden markov models. *IEEE Trans. on Power Systems*, 2016.

[60] S. H. Identification and V. Analysis. Infrastructure failures. *Seattle Office of Emergency Management*, 2019.

[61] R. Jafar, I. Shahrour, and I. Juran. Application of artificial neural networks (ann) to model the failure of urban water mains. *Mathematical and Computer Modelling*, 51(9):1170–1180, 2010.

[62] M. I. Jordan et al. Graphical models. *Statistical Science*, 19(1):140–155, 2004.

[63] L. P. Kaelbling, M. L. Littman, and A. W. Moore. Reinforcement learning: A survey. *J. artificial intelligence research*, 1996.

[64] S. Kartakis, E. Abraham, and J. A. McCann. Waterbox: A testbed for monitoring and controlling smart water networks. In *Cyber-Physical Systems for Smart Water Networks*, page 8. ACM, 2015.

[65] S. Kartakis, W. Yu, R. Akhavan, and J. A. McCann. Adaptive edge analytics for distributed networked control of water systems. In *IoT Design and Implementation*, pages 72–82. IEEE, 2016.

[66] B. Kingdom, R. Liemberger, and P. Marin. The challenge of reducing non-revenue water (nrw) in developing countries, 2006.

[67] G. Kitagawa. Monte carlo filter and smoother for non-gaussian nonlinear state space models. *J. comput. graph. statist.*, 1996.

[68] K. A. Klise, M. Bynum, et al. A software framework for assessing the resilience of drinking water systems to disasters. *Environmental Modelling & Software*, 95:420–431, 2017.

[69] K. A. Klise, R. Murray, and L. T. N. Walker. Systems measures of water distribution system resilience. Technical report, Sandia National Laboratories (SNL-NM), Albuquerque, NM (United States), 2015.

[70] P. Kohli, P. H. Torr, et al. Robust higher order potentials for enforcing label consistency. *International Journal of Computer Vision*, 82(3):302–324, 2009.

[71] J. Kwac, C.-W. Tan, N. Sintov, J. Flora, and R. Rajagopal. Utility customer segmentation based on smart meter data: Empirical study. In *IEEE conf. on smart grid communications*, 2013.

[72] T. T.-T. Lai, W.-J. Chen, Y.-H. T. Chen, P. Huang, and H.-H. Chu. Mapping hidden water pipelines using a mobile sensor droplet. *ACM Transactions on Sensor Networks (TOSN)*, 9(2):20, 2013.

[73] A. Lambert. What do we know about pressure-leakage relationships in distribution systems. In *IWA Conf. n Systems approach to leakage control and water distribution system management*. Citeseer, 2001.

[74] S. L. Lauritzen. *Graphical models*, volume 17. Clarendon Press, 1996.

[75] P. Lee, M. Lambert, A. Simpson, J. Vítkovsky, and D. Misiunas. Leak location in single pipelines using transient reflections. *Australian Journal of Water Resources*, 11(1):53–65, 2007.

[76] P. J. Lee, H.-F. Duan, M. Ghidaoui, et al. Frequency domain analysis of pipe fluid transient behaviour. *Journal of hydraulic research*, 51(6):609–622, 2013.

[77] M. Li, Y. Jiao, Y. Yang, Z. Gong, J. Wang, C. Wang, G. Wu, J. Ye, et al. Efficient ridesharing order dispatching with mean field multi-agent reinforcement learning. *WWW*, 2019.

[78] W. Li, W. Ling, S. Liu, et al. Development of systems for detection, early warning, and control of pipeline leakage in drinking water distribution: A case study. *Journal of Environmental Sciences*, 23(11):1816–1822, 2011.

[79] Y. Liu, V. Chandrasekaran, A. Anandkumar, and A. S. Willsky. Feedback message passing for inference in gaussian graphical models. *Signal Processing*, 60(8):4135–4150, 2012.

[80] Y. Lu, I. Cohen, X. Zhou, et al. Feature selection using principal feature analysis. In *Multimedia*, pages 301–304. ACM, 2007.

[81] A. Malhi and R. X. Gao. Pca-based feature selection scheme for machine defect classification. *Instrumentation and Measurement*, 53(6):1517–1525, 2004.

[82] D. M. Malioutov, J. K. Johnson, and A. S. Willsky. Walk-sums and belief propagation in gaussian graphical models. *Machine Learning Research*, 7(Oct):2031–2064, 2006.

[83] A. V. Mann, S. A. McKenna, W. E. Hart, and C. D. Laird. Real-time inversion in large-scale water networks using discrete measurements. *Computers & Chemical Eng.*, 2012.

[84] J. Mashford, D. De Silva, D. Marney, et al. An approach to leak detection in pipe networks using analysis of monitored pressure values by support vector machine. In *Network and System Security*, pages 534–539. IEEE, 2009.

[85] E. C. McNichol. Its time for states to invest in infrastructure. *Policy Futures*, 2019.

[86] S. Milan, H. Vaclav, and B. Roger. Image processing, analysis, and machine vision. *Photocopy Edition, Posts & Telecom Press, Peking*, 2002.

[87] D. Misiunas. *Burst Detection and Location in Pipelines and Pipe Networks - With Application in Water Distribution*, volume 1038. Lund Institute of Technology, 2003.

[88] S. R. Mounce, A. J. Day, A. S. Wood, et al. A neural network approach to burst detection. *Water science and technology*, 45(4-5):237–246, 2002.

[89] J. Muranho, A. Ferreira, J. Sousa, et al. Pressure-dependent demand and leakage modeling with an epanet extension–waternetgen. *Procedia Engineering*, 89:632–639, 2014.

[90] R. Murray, T. Haxton, R. Janke, W. Hart, J. Berry, and C. Phillips. Sensor network design for drinking water contamination warning systems. *U.S. EPA*, 2010.

[91] I. Narayanan, A. Vasan, V. Sarangan, et al. One meter to find them all-water network leak localization using a single flow meter. In *Information Processing in Sensor Networks*, pages 47–58. IEEE, 2014.

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

[93] P. Nguyen, A. Thorstensen, S. Sorooshian, et al. A high resolution coupled hydrologic-hydraulic model HiResFlood-UCI for flash flood modeling. *Journal of Hydrology*, 16(3):1171–1183, 2015.

[94] P. Nguyen, A. Thorstensen, S. Sorooshian, K. Hsu, and A. AghaKouchak. Flood forecasting and inundation mapping using hiresflood-uci and near-real-time satellite precipitation data: The 2008 iowa flood. *Journal of Hydrometeorology*, 16(3):1171–1183, 2015.

[95] N. Oliker and A. Ostfeld. Inclusion of mobile sensors in water distribution system monitoring operations. *Journal of Water Resources Planning and Management*, 142(1):04015044, 2015.

[96] B. Pascus. New study claims 43 states expose millions to dangerous chemical in drinking water. *CBS News*, 2019.

[97] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[98] L. Perelman and A. Ostfeld. Operation of remote mobile sensors for security of drinking water distribution systems. *Water research*, 47(13):4217–4226, 2013.

[99] Z. Poulakis, D. Valougeorgis, and C. Papadimitriou. Leakage detection in water pipe networks using a bayesian probabilistic framework. *Probabilistic Engineering Mechanics*, 18(4):315–327, 2003.

[100] R. Puust, Z. Kapelan, D. Savic, et al. Probabilistic leak detection in pipe networks using the scem-ua algorithm. In *Annual Water Dist. Systs. Analysis Symposium*, pages 27–30, 2006.

[101] Y. Qi and S. Ishak. A hidden markov model for short term prediction of traffic conditions on freeways. *Transportation Research Part C: Emerging Technologies*, 2014.

[102] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. *Proc. of IEEE*, 1989.

[103] L. R. Rabiner and B.-H. Juang. An introduction to hidden markov models. *IEEE ASSP magazine*, 1986.

[104] R. Rajkumar, I. Lee, L. Sha, and J. Stankovic. Cyber-physical systems: the next computing revolution. In *Design Automation Conference*, pages 731–736. IEEE, 2010.

[105] A. Rasekh, R. Wu, W. W. A. W. Salim, and M. K. Banks. Operation of mobile sensors for monitoring municipal drinking water distribution systems. In *World Environmental and Water Resources Congress 2014*, pages 362–367, 2014.

[106] C.-P. I. RISK. The future of smart cities: Cyber-physical infrastructure risk. 2015.

[107] L. A. Rossman. Epanet water quality model. *U.S. EPA*, 1993.

[108] L. A. Rossman. Epanet 2 users manual. *National Risk Management Research Lab*, 2000.

[109] J. Rougier. Probabilistic leak detection in pipelines using the mass imbalance approach. *Journal of Hydraulic Research*, 43(5):556–566, 2005.

[110] M. Sadri, S. Mehrotra, and Y. Yu. Online adaptive topic focused tweet acquisition. In *Information and Knowledge Mgmt.*, CIKM '16. ACM, 2016.

[111] D. Sanctis, A. E, F. Shang, and J. G. Uber. Real-time identification of possible contamination sources using network backtracking methods. *J. Water Resources Planning Mgmt.*, 2010.

[112] R. Sarrate, J. Blesa, F. Nejjari, et al. Sensor placement for leak detection and location in water distribution networks. *Water Science and Technology: Water Supply*, 14(5):795–803, 2014.

[113] F. Shang, J. Uber, and L. Rossman. Epanet multi-species extension software and user's manual. *U.S. EPA*, 2008.

[114] Q. Shuang, M. Zhang, and Y. Yuan. Node vulnerability of water distribution networks under cascading failures. *Reliability Engineering & System Safety*, 124:132–141, 2014.

[115] D. Silver, J. Schrittwieser, K. Simonyan, I. Antonoglou, A. Huang, A. Guez, T. Hubert, L. Baker, M. Lai, A. Bolton, et al. Mastering the game of go without human knowledge. *Nature*, 2017.

[116] J. Slay and M. Miller. Lessons learned from the maroochy water breach. In *International Conference on Critical Infrastructure Protection*, pages 73–82. Springer, 2007.

[117] S. K. Sowe, E. Simmon, K. Zettsu, F. de Vaulx, and I. Bojanova. Cyber-physical-human systems: Putting people in the loop. *IT professional*, 18(1):10–13, 2016.

[118] L. Spezia, M. J. Brewer, and C. Birkel. An anisotropic and inhomogeneous hidden markov model for the classification of water quality spatio-temporal series on a national scale: The case of scotland. *Environmetrics*, 2017.

[119] I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline. Pipenet a wireless sensor network for pipeline monitoring. In *Proceedings of the 6th international conference on Information processing in sensor networks*, pages 264–273. ACM, 2007.

[120] G. Strang. Inverse problems and derivatives of determinants. *Archive for Rational Mechanics and Analysis*, 1991.

[121] J. Sun, R. Wang, and H.-F. Duan. Multiple-fault detection in water pipelines using transient-based time-frequency analysis. *Journal of Hydroinformatics*, 18(6):975–989, 2016.

[122] M. A. Suresh, R. Stoleru, E. M. Zechman, and B. Shihada. On event detection and localization in acyclic flow networks. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 43(3):708–723, 2013.

[123] C. Sutton and A. McCallum. An introduction to conditional random fields. *arXiv preprint arXiv:1011.4088*, 2010.

[124] R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. 2018.

[125] R. Takanobu, T. Zhuang, M. Huang, J. Feng, H. Tang, and B. Zheng. Aggregating e-commerce search results from heterogeneous sources via hierarchical reinforcement learning. *WWW*, 2019.

[126] A.-K. Tariq, A.-T. Ziyad, and A.-O. Abdullah. Wireless sensor networks for leakage detection in underground pipelines: a survey paper. *Procedia Computer Science*, 21:491–498, 2013.

[127] J. M. Torres, L. Duenas-Osorio, et al. Exploring topological effects on WDS performance using graph theory and statistical models. *Water Resources Planning & Management*, 143(1):04016068, 2016.

[128] G. Tsoumakas, I. Katakis, and I. Vlahavas. Mining multi-label data. In *Data mining and knowledge discovery handbook*, pages 667–685. Springer, 2009.

[129] M. Y. S. Uddin, A. Nelson, K. Benson, G. Wang, Q. Zhu, Q. Han, N. Alhassoun, P. Chakravarthi, J. Stamatakis, D. Hoffman, et al. The scale2 multi-network architecture for iot-based resilient communities. In *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 1–8. IEEE, 2016.

[130] J. M. van Noortwijk and D. M. Frangopol. Two probabilistic life-cycle maintenance models for deteriorating civil infrastructures. *Probabilistic Engineering Mechanics*, 19(4):345–359, 2004.

[131] P. Venkateswaran, Q. Han, R. T. Eguchi, and N. Venkatasubramanian. Impact driven sensor placement for leak detection in community water networks. In *Proceedings of the 9th ACM/IEEE International Conference on Cyber-Physical Systems*, pages 77–87. IEEE Press, 2018.

[132] P. Venkateswaran, M. A. Suresh, and N. Venkatasubramanian. Augmenting in-situ with mobile sensing for adaptive monitoring of water distribution networks. In *Proceedings of the 10th ACM/IEEE International Conference on Cyber-Physical Systems*, pages 151–162. ACM, 2019.

[133] Y. Wang, J. Cao, W. Li, et al. Mining traffic congestion correlation between road segments on gps trajectories. In *Smart Computing*, pages 1–8. IEEE, 2016.

[134] D. Washburn, U. Sindhu, S. Balaouras, R. A. Dines, N. Hayes, and L. E. Nelson. Helping cios understand "smart city" initiatives. *Growth*, 17(2):1–17, 2009.

[135] K. Watkins. Human development report 2006-beyond scarcity: Power, poverty and the global water crisis. *UNDP Human Development Reports (2006)*, 2006.

[136] J.-P. Watson, H. J. Greenberg, and W. E. Hart. A multiple-objective analysis of sensor placement optimization in water networks. In *Critical Transitions in Water and Environmental Resources Management*, pages 1–10. 2004.

[137] J.-P. Watson, R. Murray, and W. E. Hart. Formulation and optimization of robust sensor placement problems for drinking water contamination warning systems. *Journal of Infrastructure Systems*, 15(4):330–339, 2009.

[138] Y. Weiren and J. McCann. Co-simmate: Quick retrieving all pairwise co-simrank scores. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 327–333, 2015.

[139] Y. Weng, R. Negi, and M. D. Ilic. Graphical model for state estimation in electric power systems. In *Smart Grid Communications*, pages 103–108. IEEE, 2013.

[140] A. V. Werhli, M. Grzegorczyk, and D. Husmeier. Comparative evaluation of reverse engineering gene regulatory networks with relevance networks, graphical gaussian models and bayesian networks. *Bioinformatics*, 22(20):2523–2531, 2006.

[141] A. J. Whittle, L. Girod, A. Preis, M. Allen, H. B. Lim, M. Iqbal, S. Srirangarajan, C. Fu, K. J. Wong, and D. Goldsmith. Waterwise@sg: A testbed for continuous monitoring of the water distribution system in singapore. In *Water Distribution Systems Analysis 2010*, pages 1362–1378. 2010.

[142] A. Wong, J. Young, W. E. Hart, S. A. McKenna, and C. D. Laird. Optimal determination of grad sample locations and source inversion in large-scale water distribution systems. *Water Dist. Sys. Analy.*, 2010.

[143] Z. Yao, J. R. Peddamail, and H. Sun. Coacor: code annotation for code retrieval with reinforcement learning. *WWW*, 2019.

[144] A. Yazdani and P. Jeffrey. Complex network analysis of water distribution systems. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 21(1):016111, 2011.

[145] A. Yazdani, R. A. Otoo, and P. Jeffrey. Resilience enhancing expansion strategies for water distribution systems: A network theory approach. *Environmental Modelling & Software*, 26(12):1574–1582, 2011.

[146] W. Yu and J. A. McCann. Efficient partial-pairs simrank search on large networks. *Proceedings of the VLDB Endowment*, 8(5):569–580, 2015.

[147] W. Yu and J. A. McCann. High quality graph-based similarity search. In *Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 83–92. ACM, 2015.

[148] D. Y. Zhang, C. Zheng, D. Wang, D. Thain, X. Mu, G. Madey, and C. Huang. Towards scalable and dynamic social sensing using a distributed computing framework. 2017.

[149] Q. Zhu, M. Y. S. Uddin, Z. Qin, and N. Venkatasubramanian. Upload planning for mobile data collection in smart community internet-of-things deployments. In *2016 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 1–8. IEEE, 2016.

[150] Q. Zhu, M. Y. S. Uddin, N. Venkatasubramanian, and C.-H. Hsu. Spatiotemporal scheduling for crowd augmented urban sensing. In *IEEE INFOCOM Conf. on Computer Communications*, 2018.