Title
Machine Learning-Based Decision Support to Enhance the Seismic Resilience of Distributed Infrastructure

Author
Huang, Honglan

Publication Date
2021

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA

Los Angeles

Machine Learning-Based Decision Support to Enhance the Seismic Resilience of Distributed Infrastructure

A dissertation submitted in partial satisfaction

of the requirements for the degree

Doctor of Philosophy in Civil Engineering

by

Honglan Huang

2021

ABSTRACT OF THE DISSERTATION

Machine Learning-Based Decision Support to Enhance the Seismic Resilience of Distributed Infrastructure

by

Honglan Huang

Doctor of Philosophy in Civil Engineering

University of California, Los Angeles, 2021

Professor Henry J. Burton, Chair

Seismic resilience of infrastructure has received growing attention in the past decades. This dissertation aims to propose a general decision support framework for the post-earthquake damage and recovery of distributed infrastructures in order to improve their resilience. A damage assessment framework is first proposed that incorporates a neural network-based pre-event assessment model, and a graph neural network-based model for dynamically updating the damage estimates in the recovery process. Next, a spatially explicit model is developed to quantify and estimate the possible recovery trajectory based on the Gaussian Process model. Finally, a deep reinforcement learning based framework is proposed to optimize the repair actions in the recovery process. The framework can provide critical insights to the recovery of distributed infrastructure and improve pre-event planning and post-earthquake recovery tasks.

The dissertation of Honglan Huang is approved.

Ji Yun Lee

Sanjay K. Mohanty

Scott Joseph Brandenberg

Thomas A. Sabol

Yingnian Wu

Henry J. Burton, Committee Chair

University of California, Los Angeles

2021

To my parents

For their love and encouragement

Table of Contents

List of Figures

List of Tables

BIOGRAPHICAL SKETCH

Education:

2013–2017	B.Sc. in Port, Channel and Coastal Engineering
	Tianjin University
	Tianjin, China
2017–2020	M.Sc. in Civil Engineering
	University of California, Los Angeles
	Los Angeles, California, USA
2019–2021	Ph.D. candidate in Civil Engineering
	University of California, Los Angeles
	Los Angeles, California, USA

1. Introduction

1.1 Motivation and Background

Lifeline systems (e.g., water distribution, electric power, gas supply and transportation systems) consist of interdependent structures or components distributed in a large area. Following a major earthquake, multiple components of a distributed infrastructure system can undergo different levels of damage or inoperability, which affect the functionality of the entire system. The loss of functionality can cause significant disruption to the service it provides for living and industrial purposes (Tabucchi et al. 2008), which further impact the economy and society (Lee et al. 2018; Nozhati et al. 2020; Tabucchi et al. 2008, 2010a; Xu et al. 2019).

Infrastructure resilience, i.e., the ability of an infrastructure to withstand, adapt and recover rapidly from disruptive events (Bruneau et al. 2003) has received increasing attention in the two past decades. To achieve the goal of improving resilience, reasonable decisions need to be made regarding the mitigation and response of the system, for example, the design or strengthening of the system for better performance, pre-event planning for potential future hazards, and scheduling of recovery tasks for efficient post-event restoration.

This dissertation focuses on the decision-making for the post-earthquake restoration of distributed infrastructures to improve their resilience. Various types of decision-making tasks are involved in the post-earthquake recovery process that have significant impact on the recovery. Pre-event assessment of the possible damage distribution is critical to risk mitigation and emergence planning, and providing updates to the original damage estimation during the recovery phase can provide suggestions on how plans can be adapted. Quantification and estimation of the possible recovery trajectory can also help the pre-event planning. The optimization of the restorative actions can accelerate the recovery process and reduce the cumulative functionality loss. Modern data-

driven and machine learning methodologies provide an integrated way to solve these problems and improve infrastructure resilience.

1.2 Objectives

The objective of this research is to develop a general decision support framework for reliable damage assessment, recovery trajectory estimation and repair action optimization following earthquakes, based on data-driven and machine learning methodologies. The broader objective of this research is realized through the completion of following tasks:

1. Develop a pre-event damage assessment model using neural networks, for effective estimation of the component-level damage in future earthquakes.
2. Propose a general framework for dynamically updating the damage estimation during the post-earthquake recovery process, to inform and update the inspection and repair plans. The framework is implemented using a graph neural network that incorporates a semi-supervised learning approach.
3. Develop a spatially explicit recovery model to quantify and estimate the recovery of the system following earthquakes. The Gaussian Process model is utilized to model the spatial correlation of the recovery among the system components.
4. Establish a mathematical formulation of the post-earthquake recovery (repair) action as an optimization problem. Propose a deep reinforcement learning based framework for solving the problem and validate the applicability of the methodology.
5. Expand the proposed deep reinforcement learning methodology to address the challenge of a large discrete action space, and evaluate the framework on a realistic system.

1.3 Outlines

Seven chapters are used to present the proposed decision support framework for the post-earthquake damage and recovery assessment of distributed infrastructure.

Chapter 1 provides the background and motivation of the research, as well as an outline for the framework.

A review of the common methods for the damage assessment, recovery modeling and loss and risk assessment of distributed infrastructure, as well as methods for optimizing the post-earthquake recovery, is provided in Chapter 2.

Chapter 3 presents a framework for performing dynamic seismic damage assessment of distributed infrastructure systems using graph neural networks and semi-supervised machine learning. First, pre-event damage assessment is performed using both traditional fragility-based models and a machine learning classification algorithm trained on historical damage data. Then, a graph-neural network is implemented to perform semi-supervised learning and update the pre-event predictions as observations of actual damage become available during the post-earthquake inspection process. The methodology is demonstrated on the pipe network for the City of Napa, California water distribution system. A dataset of pipes damaged during the 2014 **M** 6.0 earthquake is used for validation purposes. A conventional neural network classification model is first trained on a portion of the observed pipe damage and used to perform the pre-event damage assessment i.e., supervised learning. Following the earthquake, a graph neural network model is employed to update the damage estimates given the information incrementally collected during the inspection process.

Chapter 4 proposes a spatially explicit recovery model based on the Gaussian Process machine learning model. The proposed model is first trained to replicate the restoration process of

the water distribution system in the city of Napa following the 2014 $\mathbf{M_w}$ 6.0 earthquake. The transfer learning ability of the model for predicting the recovery for future events is demonstrated through a hypothetical $\mathbf{M_w}$ 6.7 earthquake scenario.

Chapter 5 establishes a formulation of post-earthquake restoration based on the Markov Decision Process and proposes a general deep reinforcement learning based framework for optimizing the repair action sequence. Two commonly-used deep reinforcement learning models: Deep Q Network and Deep Actor Critic are implemented to learn the optimal repair actions through interactions with a simulation-based water network environment. The proposed methodology is tested on a small hypothetical water network system to demonstrate the applicability of the reinforcement learning method.

Chapter 6 expands the methodology by introducing an actor-critic based model, Wolpertinger, to further address the challenge of a large discrete action space in the recovery optimization of realistic systems. The modified model is evaluated on a water distribution network in the city of Novato, California following selected earthquake scenario to demonstrate the effectiveness.

Chapter 7 summarizes the research that is detailed in the previous chapters, discusses the limitations and highlights future directions for the research.

2. Review of Performance Assessment and Recovery Optimization of Distributed Infrastructure

2.1 Performance Assessment of Distributed Infrastructure

2.1.1 Damage Assessment

Damage assessment models are critical to the seismic resilience of distributed infrastructure. The damage assessment model can be used (1) to provide reasonable estimation of the physical degradation caused to the infrastructure in future earthquakes, which is useful for pre-event planning and risk mitigation; (2) to better understand the influential attributes of the ground motion, site and the system components that are related to the damage, which may provide suggestions for design and strengthening of the system; (3) to be integrated as a part of a regional risk assessment model.

2.1.1.1 Fragility Function-Based Model

Conventional damage assessment models developed in the past decades are often based on fragility functions. The fragility function is developed based on one or more historical earthquakes, and a function is fitted to represent the probability that a component exceeds certain damage states, conditioned on earthquake intensity parameters.

For water distribution networks, the most-commonly used approach is the repair rate (RR) model. Specifically, a statistical model is developed to capture the empirical relationships between the repair rate (e.g., number of damaged points per unit length of the pipeline) and the earthquake intensity parameter (e.g., Peak Ground Velocity (PGV), Peak Ground Acceleration (PGA) and Peak Ground Displacement (PGD)) (Isoyama et al. 2000), and a probabilistic model of the component damage states is developed using the repair rate as parameter (Pineda-Porras and Najafi 2010). The RR model adopted in American Lifelines Alliance (ALA) takes the form of a linear

model (American Lifelines Alliance 2001)

$$RR = a \cdot PGV \tag{2.1}$$

or a power model

$$RR = b \cdot PGV^{c} \tag{2.2}$$

where a, b, c are coefficients fitted from the historical earthquake data. Next, for a pipe segment of length L with a given RR, the probability that n repairs are needed is estimated using a Poisson distribution:

$$P(N = n) = \frac{(RR.L)^{n}}{n!}.e^{-RR.L} \tag{2.3}$$

Similarly, for the seismic damage assessment of power distribution systems, a repair rate-based fragility model is often adopted. Eidinger et al. (2016) proposed the repair rate model for overhead distribution components as

$$RR = k_1 \cdot k_2 \cdot k_3 \cdot (1.388 \cdot PSA_{T=3.0} - 0.0415),\ PSA_{T=3.0} \geq 0.03g;$$
$$0.0, \text{ otherwise} \tag{2.4}$$

where $PSA_{T=3.0}$ is the peak spectral acceleration at period $T = 3.0$ seconds, k_1 reflects the primary construction type, k_2 is the secondary construction type and k_3 is a factor for age effect. The repair rate of underground distribution components is estimated in a similar way as those of the buried pipelines as in Eqs. 2.1-2.2.

2.1.1.2 Machine Learning-Based Model

Researchers in civil engineering have been applying machine learning to estimate or model the damage to infrastructure systems in the past decade. In general, these damage models take in characteristics of the earthquake, site property, and attributes of the components as explanatory variables, and the damage states of the individual components are treated as the response/target variable. A machine learning model is then trained to capture the complex nonlinear relationship

between the target and explanatory variables. The general form can be expressed as

$$\hat{Y} = f(X) \tag{2.5}$$

where $\hat{Y}$ is the estimated damage state of the component, and X is the feature vector that includes the earthquake parameters (e.g., intensity on the location of the component), site properties (e.g., V_s30), and attributes of the component (e.g., age, material and length of pipelines).

Several studies have explored the capability of tree-based machine learning models (e.g., Random Forests, Boosted Regression Trees) in modeling and estimating the damage and performance of water distribution systems (Bagriacik et al. 2018; Winkler et al. 2018; Wu and Baker 2020). In Bagriacik et al. (2018), an in-depth comparative study of statistical and machine learning based modeling for pipeline damage following the 2011 earthquakes in Christchurch, New Zealand is carried out. The study uses a set of earthquake related parameters and pipe attributes as explanatory variables, evaluates and compares the city, individual pipe, and suburb level predictive performance of the repair rate (RR), logistic regression (logit), boosted regression trees (BRT) and random forests (RF) models under three data balancing schemes and on multiple metrics. Machine learning models have also been applied to power system faults (Goswami and Roy 2019) and power grid outage (Eskandarpour and Khodaei 2017). For bridge systems, Mangalathu et al. (Mangalathu et al. 2020; Mangalathu and Jeon 2019) applied and compared a series of machine learning models for regional damage assessment.

2.1.2 Recovery Modeling

Post-earthquake recovery models are used to estimate the possible trajectories for future earthquakes and are therefore useful for pre-event risk management and post-event planning. A comprehensive state-of-the-art review of the recovery modeling for socio-technical systems can be found in to Kang (2018). A review of several common modeling methods within the scope of

post-earthquake recovery of distributed infrastructure is provided in this subsection. Several other modeling methods, for example, agent-based modeling (Nejat and Damnjanovic 2012a) and network-based method (Didier et al. 2015), are used more often in the scope of recovery of housing or general community, hence, they are not included.

2.1.2.1 Data-Driven Statistical Model

Statistical recovery models aim to provide recovery time estimates or a restoration probability (Martell et al. 2020). These models are learned on recovery datasets from past earthquakes and intend to predict for future earthquakes.

Various regression-based methods have been studied to model post-earthquake recovery trajectories. MacKenzie and Barker (2012) proposed a dynamic inoperability input-output model (DIIM) to quantify the resilience of distributed infrastructure. A mixed-effect model is developed to include the influence of different scenarios. The proposed framework is used to model the power outage and recovery of the electric power system in Oklahoma. Monsalve and de la Llera (2019) introduced a time-series based statistical model for evaluating and simulating the restoration of complex interdependent systems, while modeling their restoration as interdependent processes. The model is fitted on restoration data from six iconic earthquakes and evaluated against a suite of service restoration curves associated with different infrastructure. In Cao et al. (2020), a framework for estimating the recovery curve of infrastructure is proposed based on the Gaussian Process Regression and utilizing the expert-elicit data.

Several researchers also explored the use of survival analysis for modeling infrastructure recovery. Mojtahedi et al. (2017) used Cox's proportional hazards model to estimate the recovery of transport infrastructure across regional areas in New South Wales, Australia. Barker and Baroud (2014) proposed a proportional hazards model for the recovery of infrastructure systems. A case

study on electric power outage in the US from January 2002 to June 2009 is presented to illustrate the proposed methodology. In both studies, the type of event was included as a covariate, and the proportional hazards model was used to provide estimates of the instantaneous rate of recovery of an infrastructure system and the likelihood that recovery occurs prior to a given point in time.

2.1.2.2 Discrete Event Simulation Model

In the Discrete Event Simulation (DES) models, the response of a system is modeled as a sequence of events that occur at discrete time (Fishman 1978; Tomar et al. 2020; Varga 2001). The DES models have several core elements: entities, attributes, events, resources and time. Entities represent specific objects within the system. Events are occurrences that can affect the state of an entity. Resources are objects that provide services to entities (Fishman 2001; Kang 2018; Tomar et al. 2020). In terms of post-earthquake restoration of distributed infrastructure, the physical components of the systems are modeled as entities (e.g., pipelines, tanks and pumps of water networks, transmission substations and power plants of power networks), the inspection/repair crews and materials are represented as resources, and the activities involved in the restoration process (e.g., inspection, repair and travel) are represented as events.

Çağnan and Davidson (2007) used DES to model the restoration of the Los Angeles Department of Water and Power (LADWP) electric power system, following the 1994 Northridge earthquake. Tabucchi et al (Tabucchi and Davidson 2008; Tabucchi et al. 2010b) also used DES to model the restoration of the LADWP water network following the 1994 Northridge earthquake. In their study, The Graphical Iterative Response Analysis of Flow Following Earthquakes(GIRAFFE) was used to model the hydraulic performance and serviceability of the water system. Kang and Lansey (2013) proposed a DES based restoration framework for water networks, which includes six main steps: (1) Identify initial damage state; (2) Calculate resources required;

(3) Prioritize and allocate resources; (4) System recovery according to prioritization rules; (5) Calculate system serviceability; and (6) Plot restoration curve. Tomar et al. (2020) used DES model for the recovery of water distribution systems, using the water network of the city of Napa as the case study. The Water Network Tool for Resilience (WNTR) was used to perform hydraulic analysis of the water system. The DES model was first calibrated to replicate the recovery trajectory from the 2014 south Napa $\mathbf{M_w}$ 6.0 earthquake. The authors then developed a pipe fragility function based on data from the 2014 event. The tuned DES model was used to access the recovery of the Napa network in a hypothetical $\mathbf{M_w}$ 6.7 earthquake scenario occurring on the West Napa Fault. 1000 realizations of damage maps were sampled and the corresponding restoration process and hydraulic performance was assessed. The uncertainty is the restoration was also quantified.

2.1.2.3 Stochastic Simulation Model

Stochastic simulation models represent the response of a system through random variables that change with individual probabilities. The recovery of distributed infrastructure can be modeled in two ways: discrete-time, discrete-state-based models that capture the probability of the system transitioning to a different state at a given discrete time; time-based models capture the probability density of the time it takes for the system to leave the current state once entered (Kang 2018; Mishalani and Madanat 2002)

A discrete-time, discrete-state Markov Process based model is proposed by Kozin and Zhou (1990) to model the post-earthquake restoration of lifeline systems. Decò et al. (2013) proposed a probabilistic framework for the pre-event assessment of bridges. A model was proposed to describe the initial loss and restoration of functionality. Fragility analysis was performed to evaluate the damage induced to bridges by a seismic event. A six-parameter

sinusoidal function was then used to describe the restoration process. Dhulipala et al. (2021) proposed a generalized framework for modeling post-event recovery of individual system and system of systems (SoS) under multiple hazards based on the Markov Processes. Four of Markov type processes were considered in the framework: Markov chains, Markov processes, semi-Markov processes (SMP), and semi-Markov processes with non-renewal (SMP-NR) features. The framework was applied to a spectrum of problems including hindcasting recovery, updating recovery in light of new data, accounting for hazard intensity and system damage states in recovery forecasting, and capturing temporal interactions between multiple hazards while forecasting recovery.

2.1.3 Loss and Risk Assessment

Different from the single-location facilities, the performance assessment of the distributed infrastructure should incorporate the spatial correlation among the system components, therefore a scenario-based simulation approach is often adopted, in which the joint performance of all system components/facilities are accessed for one earthquake at a time (Soleimani et al. 2021).

Extensive studies have been carried out for the loss assessment following a single event or a small set of events consisting of significant historical earthquakes in order to estimate the impact from these destructive scenarios and inform emergence planning (Crowley and Bommer 2006). Some studies aim at developing frameworks to estimating the economic or serviceability loss associated with a single event. Yuan (2008) conducted a study to estimate the direct economic loss caused by the 2008 $\mathbf{M_w}$ 8.0 Wenchuan earthquake in China. The damage to buildings and infrastructure account for about 36% and 22% of the total direct loss. Miano et al. (2015) proposed a loss assessment framework for a potfolio of bridges using the 1980 Irpinia earthquake in Italy as a case study. The fragility function was developed using a Bayesian updating scheme where the

bridge-specific fragility data available in the literature are used as a prior. The probability density function of the repair cost given the damage state is also developed using Bayesian updating using the cost for collapsed and damaged bridges in past earthquakes as a prior. 3000 Monte Carlo simulations of the PGA field are performed to estimate the probability distribution of the repair cost. On the other hand, many of the studies reviewed in Section 2.1.2 focused on quantifying the immediate functionality loss and the restoration of a distributed infrastructure in a historical earthquake (Masoomi et al. 2020; Tabucchi et al. 2010a; Tabucchi and Davidson 2008; Tomar et al. 2020).

For the purpose of decision making practice such as insurance applications, however, it is often required to assess the infrastructure performance for a large number of or all possible future events (Crowley and Bommer 2006; Tomar and Burton 2021a). In this regard, Mehralian and Azarbakht (2020) proposed a framework for the loss assessment of power distribution system in Arak city, Iran. The main steps involve: (1) performing Probabilistic Seismic Hazard Analysis (PSHA) for the study region using a seismic catalogue of earthquakes with a radius of 150 km; (2) developing component fragility curves using historical data; and (3) using the HAZUS framework (Federal Emergency Management Agency (FEMA) 2003) to estimate the direct loss. Farahani et al. (2020) used a similar approach to assess the loss to the gas pipeline network in Asaluyeh, Iran. In the analysis, physical damage, direct economic and human impacts are considered. Laucelli and Giustolisi (2015) proposed a framework for risk assessment of water distribution systems. In the study, the pipe fragility functions are adopted from the American Lifelines Alliance (2001). A multi-objective optimization program was formulated to search for the most destructive scenarios. The risk and vulnerability of the water network was assessed based on the unsupplied water demand obtained from pressure-driven hydraulic simulation. It is noteworthy that the restoration

process and resilience metrics were not considered in these studies.

The risk-based assessment framework proposed by Tomar and Burton (2021a) integrates the previous approaches, in which the immediate loss and the restoration process are assessed based on a selected stochastic event catalogue. The key steps in the framework are as follows: (1) generate a set of stochastic earthquake events; (2) for each event, a set of ground motion maps is produced using one or more appropriate ground motion and spatial correlation models; (3) for each map, several realizations of the system damage are sampled from a fragility-based damage assessment models as reviewed in Section 2.1.1; (4) Use a DES-based recovery model to simulate the recovery trajectory following each damage case; (5) Compute the resilience metrics associated with the restoration; (6) Integrate over the events to obtain the risk-based metrics (e.g., annual exceedance rate).

2.2 Recovery Optimization of Distributed Infrastructure

Various tasks, e.g., inspection, damage assessment, resource allocation and repair are involved in the post-earthquake restoration of distributed infrastructure. Optimizing the scheduling of these tasks can have significant influence on the rapidity and effectiveness of the restoration. Two common types of optimization formulations are studied in the literature. The first is based on integer programming (IP) or mixed integer linear programming (MILP). In this formulation, a set of decision variables are introduced and the optimization objective is established (e.g., minimize the total loss of serviceability), and a series of constraints are then imposed which incorporate the resource availability, task precedence and the description of how the system operates or updates. Because the problem scale of the recovery optimization is large, heuristic algorithms are usually developed to solve such programs. This type of formulation often involves a complex set of equations, and are often system-specific. The second formulation is based on the Markov Decision

Process (MDP), where a decision maker/agent interacts with an environment/system in finite time steps and learns to improve the policy for making decisions. The system is characterized by a finite set of states, and the agent can take an action from a finite set of actions in each time step and receive some rewards as feedback. In this formulation, a physical simulator model of the system is often used, which gives the response once an action is taken. Such a formulation is often simple, as long as the state, action and reward function are appropriately defined.

2.2.1 Mixed Integer Linear Programming

Mixed Integer Linear Programming (MILP) is a mathematical optimization program in which some of the variables are constraints to be integers while the other variables can be non-integers. MILP can be used in many general task scheduling problems.

Xu et al. (2007) proposed a stochastic integer program to optimize the inspection, damage assessment and repair tasks for the post-earthquake recovery of power distribution systems. The optimization objective is to minimize the average time each customer is without power, and a genetic algorithm is used to solve the program. The optimization model outputs the recommended set of schedules, and the priority of actions are input to a DES model for the more complex restoration process to evaluate the performance. The proposed method is tested on the LADWP power distribution system subjected to 47 earthquake scenarios, and three metrics: the average time each customer is without power, time required to restore 90% of customers, and the time required to restore 98% of customers.

Nayak and Turnquist (2016) proposed an integer programming model for scheduling the repair tasks to minimize the cost of the recovery for water distribution systems. The objective consists of two parts: the disruption cost of unmet water demand, and the cost of repair actions. The constraints include resource availability and task precedence. In addition, the objective

function requires evaluation of the water demand of the system in each time step, hence a flow problem is embedded, i.e., the hydraulic equations are calculated given the current state of the system. A simulated annealing (SA) algorithm is proposed to solve the program. The proposed model is tested on the hypothetical Anytown network system for several hypothetical scenarios.

Xu et al. (2019) further proposed a MILP-based general formulation for the repair sequence scheduling for the post-disaster recovery of critical infrastructure. The formulation is based on three assumptions: (1) the repair teams have the same work efficiency; (2) each damaged component requires only one repair team; (3) the repair time for each component is known and given beforehand. A flow operation model is adopted and included in the constraints. Two novel heuristic methods (NHM) are proposed by the authors which combines the time index-based heuristic methods (TIBHMs) and the component index-based exact solution method (CIBESM). The proposed NHMs were tested on the electric power trans-mission system in Shelby County, Tennessee. The damage maps were generated by assuming earthquake magnitudes from $\mathbf{M_w}$ 6.0 to $\mathbf{M_w}$ 9.0 with a step of 0.5, and 500 scenarios are generated for each magnitude. The proposed solutions were compared with three component importance-based methods (CIBMs), a genetic algorithm-based method (GABM), the TIBHM and CIBESM.

Bellagamba et al. (2019) proposed a decision support algorithm for the restoration of water supply networks. The priority list of pipe inspection is first generated using a fragility-based measure. A MILP model is then formulated to optimize the pipe repairs, which is solved using a GA algorithm. A case study based on the Christchurch water supply network recovery following the 22 February 2011 $\mathbf{M_w}$ 6.2 earthquake is carried out for demonstration purposes. The evaluation results show that the proposed method successfully reduced the proportion of disruption after two days by approximately 30% and reduced the overall system resilience loss by 20%.

2.2.2 Markov Decision Process

The Markov Decision Process (MDP) is a general mathematical framework for modeling sequential decision-making problems. In MDP, a decision maker (agent) interacts with an environment in finite time steps and receives some reward/feedback as a result of each action. The output of the MDP is a good decision-making policy. Small scale MDPs can be solved by dynamic programming (DP), while large scale MDPs are often intractable and solved by approximate solutions such as the rollout algorithm, reinforcement learning (RL), and deep reinforcement learning (DRL) methods. The optimization for scheduling of restorative tasks in the post-earthquake recovery can be viewed as a sequential decision-making problem, in which different sequences of restorative actions result in different system resilience outcomes (e.g., restoration curves). Therefore, such problems can be modeled as an MDP, where the state represents the damage to the system, and the action represents the scheduled restorative action, and the reward can be formulated based on certain metrics that represent the current functionality of the system.

Sarkale et al. (2018)proposed a MDP-based modeling approach for the restoration of water distribution systems. The formulation includes a simulator for the response of the water network after a repair action is executed, and the repair time is stochastic. The objective is to have the maximum people get water in the minimum amount of time. In the MDP formulation, the state is defined as the vector of component damage levels, the action is defined as a vector of the repair actions to be carried out at the current time step, and the reward is defined as the number of people who have water after executing the current action divided by the total repair time up to current time step. The rollout algorithm was adopted for solving the MDP. The portable water network in Gilroy, California was considered for case study, subjected to an $\mathbf{M_w}$ 6.9 earthquake at the San Andreas Fault. Evaluation results for 100 scenarios show that the rollout algorithm outperforms

the base policy. The authors later applied a similar formulation for the recovery optimization of the electric power distribution system in Gilroy, California (Nozhati et al. 2019). Two objectives were considered: minimizing the number of days needed to restore electricity to a certain fraction of the total population and maximizing the number of people who have electricity per unit of time. A more general formulation was further proposed in Nozhati et al. (2020) with the scope of community recovery, which integrates the recovery of water, power networks and food retailers.

Gol et al. (2019) proposed a MDP-based framework for the electric system operators to restore electricity following earthquakes. In the formulation, the state represents the damage status of the branches in the network, the action represents the set of branches to be energized at the current time step, and the transition probabilities are computed based on Peak Ground Acceleration (PGA) data. The outcome of the model is an optimal strategy to energize the power lines after earthquake damage.

In Memarzadeh and Pozzi (2019b), a MDP based framework was proposed for managing pre-event operations and the post-event restoration of infrastructure. A Safe Q-Learning algorithm was proposed which is based on the Q-Learning method (Watkins 1989) and incorporates constraints on the action exploration. Examples of management under the risk of extreme events on a hypothetical infrastructure system, a water reservoir, and a reinforced concrete bridge were used to illustrate the methodology.

3. Dynamic Seismic Damage Assessment of Distributed Infrastructure Systems using Graph Neural Networks and Semi-Supervised Machine Learning

3.1 Introduction

Comprehensive and reliable estimation of the damage to distributed infrastructure systems following earthquakes is critical to risk mitigation and emergency planning of the affected community (Bagriacik et al. 2018; Tomar et al. 2020). Existing damage estimation models commonly establish empirical relationships between the damage/repair rate and seismic intensity parameters (Pineda-Porras and Najafi 2010). The development of an empirical model relies on learning to extract the complex relationships from the representative datasets collected from past earthquakes, and advanced machine learning methods have been shown to be effective for this purpose.

The development of the traditional repair rate (RR) based fragility model involves selecting damage scenarios from historical earthquakes, fitting a statistical model for the empirical relationships between the repair rate and the earthquake intensity parameter, and proposing a probabilistic model of damage with the repair rate as parameter (Pineda-Porras and Najafi 2010). The most commonly used intensity parameter for buried pipelines is Peak Ground Velocity (PGV), while several other parameters (e.g., Peak Ground Acceleration, Peak Ground Displacement (Isoyama et al. 2000)) have also been used in previous studies. The American Lifelines Alliance (ALA) model (American Lifelines Alliance 2001) uses PGV as the intensity measure, and the effects of pipe material, joint type, and soil corrosiveness are considered when computing the fragility for the pipelines.

In the civil engineering community, researchers have been exploring the application of

statistical and machine learning methods. Sun et al. (2020) summarizes the application of machine learning in structural design and performance assessment into four categories, one of which is predicting damage following the occurrence of a hazard event. To date, a large amount of effort have been placed on developing models for individual infrastructure (e.g., buildings). Limited studies have explored the application of machine learning to distributed infrastructure systems. In this regard, Winkler et al. (2018) applied tree-based methods for predicting pipe failures due to deterioration in a medium sized city in Austria. Wu and Baker (2020) utilized the Random Forest machine learning model to estimate the performance and reliability of water distribution networks. Bagriacik et al. (2018) performed an in-depth comparative study of statistical and machine learning based modeling for pipeline damage following the 2011 earthquakes in Christchurch, New Zealand. The study uses a set of earthquake related parameters and pipe attributes as explanatory variables, evaluates and compares the city, individual pipe, and suburb level predictive performance of the repair rate (RR), logistic regression (logit), boosted regression trees (BRT) and random forests (RF) models under three data balancing schemes and on multiple metrics. It is worth noting that these studies utilize classical machine learning methods and focus on tree-based models. Mangalathu and Jeon (Mangalathu and Jeon 2019; Mangalathu et al. 2020) applied a series of machine learning models for the failure mode prediction of bridge columns and the regional damage assessment of bridge systems.

Most of the current applications are aimed at developing models to predict damage for the future earthquakes i.e., pre-event predictions. On the other hand, less effort has been put on studying models that can help improve or update the inspection and damage assessment processes during the recovery. Specifically, a question of interest is whether the information collected at certain time points and a few locations within the network can inform the damage estimation of

the entire system, i.e., inferring the unknown damage from the known damage. If so, the estimation could be better than only using a model previously developed based on prior earthquakes. A semi-supervised learning-based model can potentially address the problem. Since a system network can be modeled as a graph, the damage estimation model can utilize the information provided by the connections among the system components. The graph neural network (Wu et al. 2019c), which has been proposed in recent years, is useful for this type of modeling.

This study presents a methodology that uses a graph-based neural network model to dynamically update pre-event seismic damage assessments in distributed infrastructure systems. The pre-event assessment can be performed using the traditional fragility-based approaches or a supervised machine learning classification model (e.g., random forests, decision trees). Incremental information collected during the inspection is then used to update damage estimates by implementing semi-supervised learning via the graph neural network. The pipe network for the City of Napa, California water distribution system is used as an application testbed. However, it is worth noting that the developed methodology is generalized and can be applied to other types of distributed infrastructure (e.g., bridge networks, building portfolios, power networks). The remainder of the paper is organized as follows: An overview of the framework is presented in Section 3.2. The theoretical foundations of neural networks including graph neural networks are discussed in Section 3.3. A brief description of the Napa water network and the pipe damage dataset from the 2014 $\mathbf{M_w}$ 6.0 South Napa earthquake is provided in Section 3.4 and Section 3.5 presents the results of the pre-event damage assessment using the neural network-based model trained on this dataset, which are compared to results from an existing repair rate-based fragility model. Section 3.5 applies the graph neural network-based model for informing the damage of the entire network given the partial information provided during the inspection process. Finally,

Section 3.6 summarizes the contributions of this study and discussions directions for future work.

3.2 Overview of Methodology

An overview of the proposed methodology is presented in Fig. 3.1 where the overall goal is to provide real-time updates to damage predictions to better inform resource-allocation and planning decisions in the post-earthquake environment.

The framework is comprised of two main steps. It starts with a pre-event assessment immediately following the earthquake (day 0), which provides an initial estimate of the possible damage $\hat{y}(0)$ to the distributed infrastructure system. The pre-event assessment model is developed using "labeled data", where the features are the earthquake attributes (e.g., epicentral distance, intensity) and system component attributes (e.g., material, age), and the damage labels for all system components are given. The prior damage information can be obtained from reconnaissance datasets of historical earthquakes that are reported on websites, reports and journal papers. The data is then used to train a supervised learning-based machine learning models (e.g., random forests, neural networks), or to develop/calibrate a traditional fragility-based model. The learned model then produces an initial damage estimate given the attributes of the system components and the information of the current earthquake.

Next, a post-event assessment model provides dynamic updates of the damage estimation given the information incrementally collected during the inspection process following the earthquake. The model is developed using "partially labeled data", where the features are the earthquake and system component attributes, and a subset of the damage labels for the system components is given. Specifically, the case where the subset of damage labels collected up to day t is available. A graph neural network model is trained using the collected data via the semi-supervised learning approach, which utilizes the network connectivity information in the training

process to better infer the unknown damage from the known ones. The learned model then produces an updated estimate $\hat{y}(t)$ for the damage map of the entire system, which can help adjust and update the current inspection and repair plans, and therefore improve the overall recovery process. This dynamic learning and updating process continues with the restoration process (e.g., repeats weekly), as shown in Fig. 3.1.

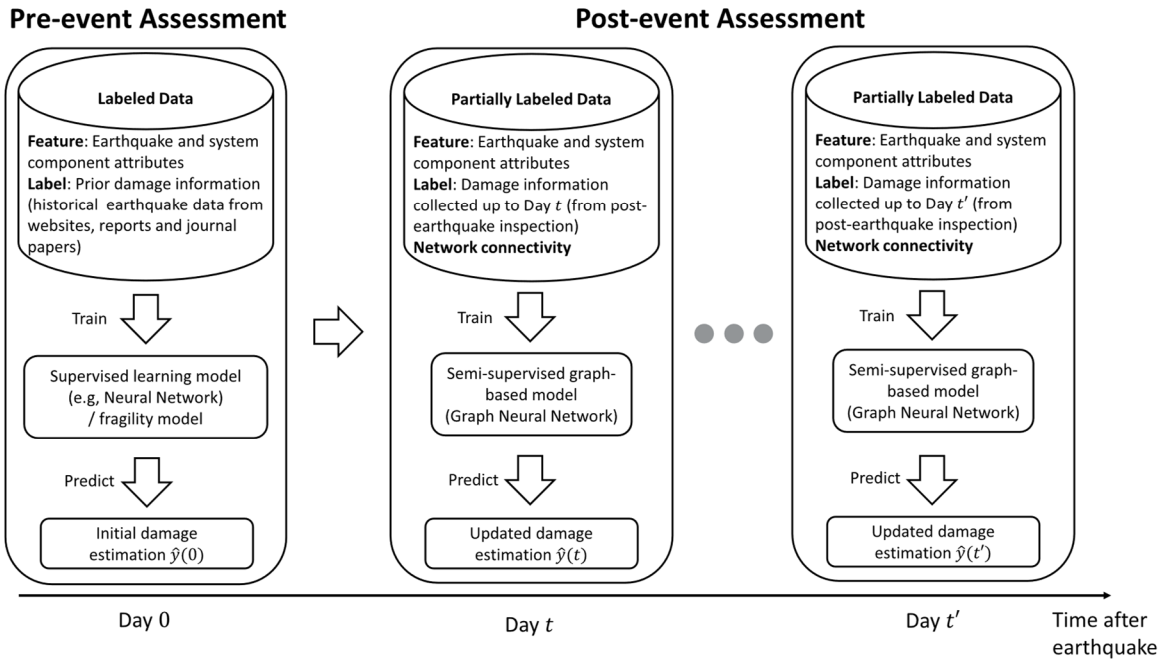


Figure 3.1 Schematic overview of the proposed methodology

3.3 Neural Networks and Graph Neural Networks

This section presents the theoretical formulation of the neural network-based machine learning method and its graph variants.

3.3.1 Neural Networks

Machine learning (ML) is a category of algorithms that are able to learn from data the underlying patterns in order to make predictions and support decision making (Sun et al. 2020). Artificial Neural Networks (ANN) is a type of machine learning/pattern recognition model

inspired by information processing in biological systems. The goal of the ANN is to approximate the function that defines the complex relationship between the input and output. As the first and simplest type of neural network, a feed-forward neural network (or multi-layer perceptron) takes the input features x and extracts information through a series of layers (Wu et al. 2018):

$$h^l = a^l(W^l h^{l-1} + b^l) \tag{3.1}$$

where h^l is the lth hidden layer, W^l is the weight matrix, b^l is the bias vector, and $a^l(\cdot)$ is the activation function. The final layer outputs the predicted response $\hat{y}$, as shown in Fig. 3.2.

There are various advantages of neural networks. First, by stacking multiple layers, the neural network is able to extract new comprehensive representations of the input features. Second, the model is able to flexibly capture complex nonlinear relationships between the input and output and is sometimes described as a "universal function approximator" (Csáji 2001).

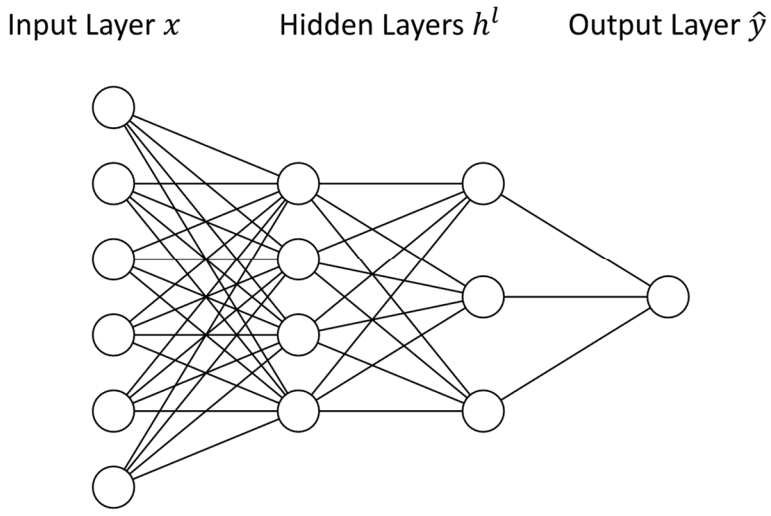


Figure 3.2 Schematic representation of neural networks

3.3.2 Graph Neural Network

Within the last decade, neural networks have achieved unprecedented successes in tasks

involving information retrieval from Euclidean data (image, video, and text). Recently, neural networks have been used to perform predictions on graphs, which is a more complex and general data type (Zhou et al. 2018). For example, there is increasing interest in learning from a knowledge graph, where entity nodes (e.g., author, paper, conference) are connected by links which represent their relationships (e.g., whether a paper is cited by the other) (Wu et al. 2019c). Graph Neural Networks (GNNs) were developed to learn from such complex graph data. Fig 3.3 shows a schematic representation of the GNNs.

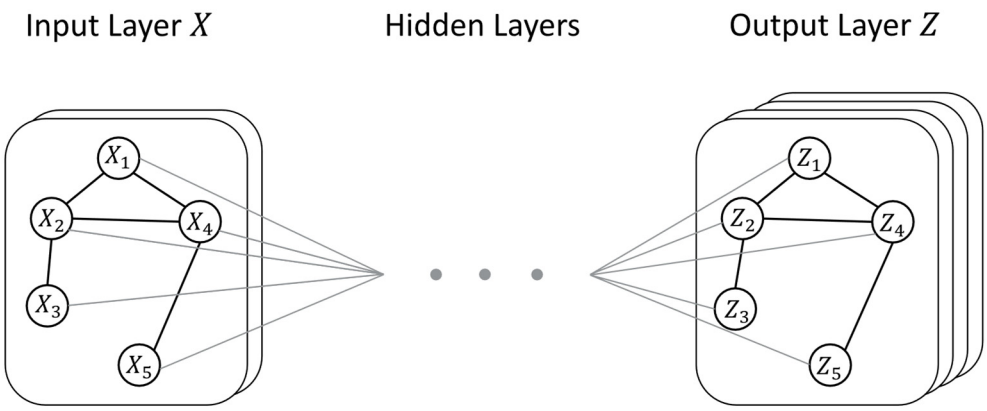


Figure 3.3 Schematic representation of graph neural networks

3.3.3 Graph Convolutional Network

Convolutional neural networks (CNNs) have led to big breakthroughs in computer vision tasks because of their ability to learn feature representations and extract information from raw images. The basic idea of the convolution operation is to slide a filter though the pixels to create feature maps of lower-dimension and summarize particular aspects of information from the input. Similar to the convolution applied to images (grid data), a convolution operation can be applied to graph data to generate node representations by aggregating information from a neighborhood (Wu et al. 2019b). Kipf and Welling (2016) proposed the Graph Convolutional Network (GCN) which

takes the form

$$Z = f(X, A) = \text{softmax}(\hat{A}\text{ReLU}\big(\hat{A}XW^{(0)}\big)W^{(1)}) \tag{3.2}$$

where $X \in R^{N\times C}$ is the input graph with N nodes and C features (channels), A is the adjacency matrix which encodes the connections among the nodes, $Z \in R^{N\times F}$ is the output representation with F features, which in this form is a probabilistic distribution of the classes over the nodes. $\hat{A} = \widetilde{D}^{-\frac{1}{2}}\tilde{A}\widetilde{D}^{-\frac{1}{2}}$, where $\tilde{A} = A + I_N$ is the adjacency matrix with a self-loop, I_N is the identity matrix, and $\widetilde{D}_{ii} = \sum_j \tilde{A}_{ij}$. $W^{(0)} \in R^{C\times H}$ is the input-to-hidden weight matrix, where H denotes the number of feature maps in the hidden layer, $W^{(1)} \in R^{H\times F}$ is the hidden-to-output matrix. The softmax function is defined by $\text{softmax}(x_i) = \frac{e^{x_i}}{\sum_{j=1}^{K} e^{x_j}}$. In summary, the GCN takes the original graph and produces a new representation over the nodes.

3.3.4 Graph Attention Network

When aggregating the information from neighborhoods, the GCN does not evaluate the importance of the information from different neighbors. In other words, for a given node, the messages from all neighbors are considered to be equally important. To improve the model, there is an interest in learning to determine which neighbors are more important, which are then weighted to produce a new representation. The Graph Attention Network (GAT) (Veličković et al. 2018) provides this improvement by utilizing the idea of an attention mechanism, which was introduced in natural language processing tasks (e.g., machine reading, machine translation), and has achieved great success in combination with Recurrent Neural Networks (RNN), a type of neural network for processing variable length sequences like texts (Sherstinsky 2018). In machine translation tasks, for example, a sentence in English is input to the model to be translated to Spanish, and the attention mechanism within the model learns to put different weights on the words of the

input sequence when producing each word in the output sequence. In GAT, attention weights are put on a node's neighbor when producing its new representation. Specifically, for node $j \in N_i$, where N_i is the set of neighbor nodes of node i, the attention score can be computed by

$$e_{ij} = a(Wh_i, Wh_j) \tag{3.3}$$

where $a(\cdot)$ is the attention score, h_i is the input feature representation of node i, and W is the weight matrix. The coefficients are further normalized to have $\alpha_{ij} = \text{softmax}_j(e_{ij})$. After the attention coefficients are computed, we can generate the new representation for node i as

$$h_i' = \sigma(\sum_{j \in N_i} \alpha_{ij} W h_j) \tag{3.4}$$

where $\sigma(\cdot)$ is the sigmoid function.

3.3.5 Supervised and Semi-supervised learning

In supervised learning, we have the labeled data $\{(x_i, y_i)|i = 1, \dots, n\}$, where $x_i \in R^p$ is the feature vector of instance i, y_i is the label, which can be continuous in regression problems and discrete in classification problems. The objective of supervised learning is to minimize the empirical loss function defined by (Murphy 2012)

$$\underset{\theta}{\text{argmin}} \frac{1}{n} \sum_{i=1}^{n} \phi\big(y_i, f(x_i; \theta)\big) + \lambda\Omega(\theta) \tag{3.5}$$

For semi-supervised learning, we have the labeled data $\{(x_i, y_i)|i = 1, \dots, L\}$, and unlabeled data $\{x_i|i = L + 1, \dots, n\}$, and the objective is to learn a predictive model which not only utilizes the labeled data, but also takes useful information provided by the unlabeled data. Semi-supervised learning on graphs further considers the connections among all nodes, and the objective is to learn an embedding over the nodes, given the data and a subset of labels. For example, in GCN, the loss function is constructed by (Kipf and Welling 2016)

$$L = -\sum_{l \in y_L} \sum_{f=1}^{F} Y_{lf} \ln Z_{lf} \tag{3.6}$$

where Z is the output representation computed based on the information from all nodes and the edges, y_L is the set of available labels. The loss function evaluates the cross-entropy over the labeled instances.

3.4 Overview of Napa Water Network and Pipe Damage from 2014 South Napa Earthquake

The methodology is applied to the pipe network in the water distribution system that serves the City of Napa, California. The relevant information is provided by Napa Water Division (NWD). The Napa water network serves a population of approximately 88,000 people. The system includes 3 water treatment plants, approximately 612 km (380 mi) of pipeline (7365 pipe segments), 15 storage tanks with a total of 113,562 m^3 storage, and 10 pump stations. Additional details of the topology and components that encompass the system are provided in (Napa Water Division 2017; Tomar et al. 2020).

A $\mathbf{M_w}$ 6.0 earthquake occurred in South Napa on August 24, 2014, with an epicenter location of 38°13′12′′N;122°18′36′′W. The earthquake caused a total of 250 pipeline leaks (203 pipe segments, 2.76% of the total pipelines) to the water system. Fig. 3.4 shows the spatial distribution of the pipe damage, most of which is distributed evenly on the west side of the system. Approximately 75% of the damaged pipes were made from Cast Iron (CI) and the remaining 25% were made from Asbestos Cement (AC), Polyvinyl Chloride (PVC), Ductile Iron (DI), Steel (STL), and other unknown materials (mostly reinforced concrete). Pipe leaks were repaired relatively quickly with 120 leaks fixed in 5 days, and during that time, only 5% of the customers were without water. Within a day of the earthquake, NWD was able to assemble repair materials (e.g., pipe,

repair clamps, fittings) valued at approximately $250,000. Within a month after the earthquake, the roof system of the damaged tank was replaced (Napa Water Division 2017).

Figure 3.4 Spatial distribution of the pipe damage in the Napa water network following the 2014 earthquake

3.5 Supervised Learning Model for Pre-Event Damage Assessment

3.5.1 Feature Selection

The features used for this study can be divided to earthquake related parameters and pipe attributes (Bagriacik et al. 2018). The earthquake parameters include PGV and epicentral distance. The PGV data is interpolated using Kriging from the recorded sites to the pipe locations (Tomar et al. 2020). Fig. 3.5 shows the spatial distribution of PGV over the Napa water pipe network. Fig. 3.6 shows the distribution of the earthquake related features, grouped by the damage state following the 2014 event. It can be observed that the damage tends to occur at locations with higher PGV. However, it is also noteworthy that there are a great portion of pipes with high PGV values did not suffer from damage. This might suggest a limitation of the commonly used repair rate-based fragility model, in which PGV is used as the primary indicator of damage.

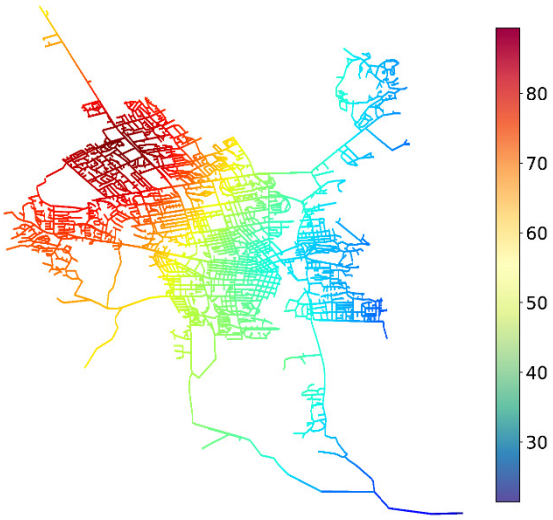
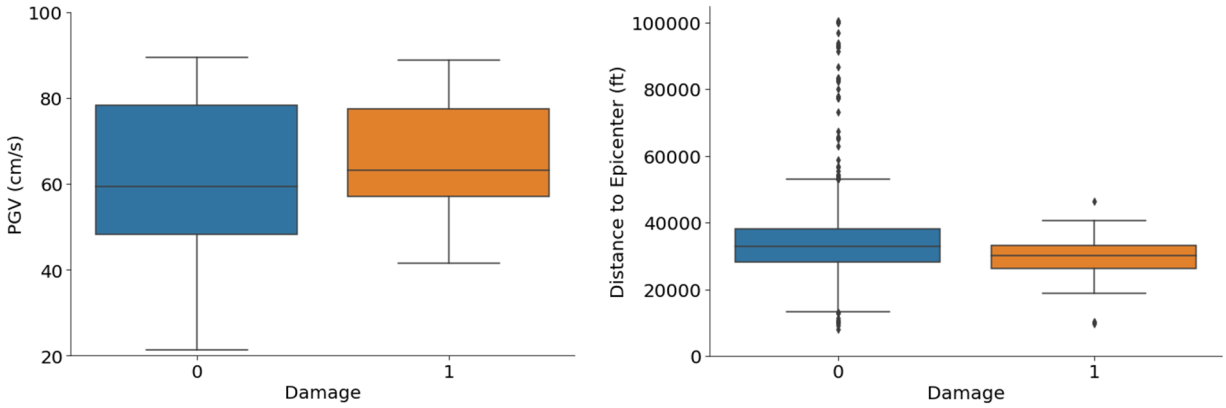


Figure 3.5 Distributions of the PGV (cm/s)



(a) (b)

Figure 3.6 Distribution of (a) PGV and (b) epicentral distance for the pipe damage states

The pipe attributes are selected based on their correlation with earthquake-induced damage as evidenced by historical events (Bagriacik et al. 2018). The selected pipe attributes include the material, roughness, diameter, length, installation year and pipe type. The distribution of pipe materials is shown in Fig. 3.7. Most of the pipes are made of Ductile Iron (DI) and Cast Iron (CI). In the analysis, the pipe material types with less than 100 pipes are lumped together to the "Other"

category. Therefore, the final types used are Ductile Iron (DI), Cast Iron (CI), Steel (STL), Asbestos Cement (AC), C900 and Other. Fig. 3.8 shows the distribution of pipe attributes for the two damage classes. According to Fig. 3.8 (a), damage is more likely to occur for pipes with smaller roughness coefficient. Fig. 3.8 (b) and 3.8 (c) shows the distribution of the pipe diameters. The box plot in Fig. 3.8 (c) includes large outlier points, and Fig. 3.8 (d) zooms closer to the statistically meaningful part. It can be observed that the distributions of pipe diameter are almost the same within damaged and undamaged pipes, even considering the outlier points. However, based on previous studies (Bagriacik et al. 2018), pipe diameter is an important factor. Specifically, it was found that smaller diameter pipes have a higher likelihood of damage. Fig. 3.8 (e) and 3.8 (f) show the distribution of pipe lengths. Fig. 3.8 (e) contains large outlier points. When the distribution of pipe length for undamaged pipes is examined, it is determined that many long pipes did not fail. However, if the focus is placed on the main statistical component of the distributions, it can be concluded that longer pipes are more prone to damage, which is consistent with the HAZUS model (FEMA 2012). Fig. 3.8 (g) shows the distribution of the year of installation for the pipes where it is noted that older pipes are more likely to be damaged. From Fig. 3.8 (h), which shows the counts for two different pipe types for the two damage classes, it can be concluded that most damage occurs on distribution lines.

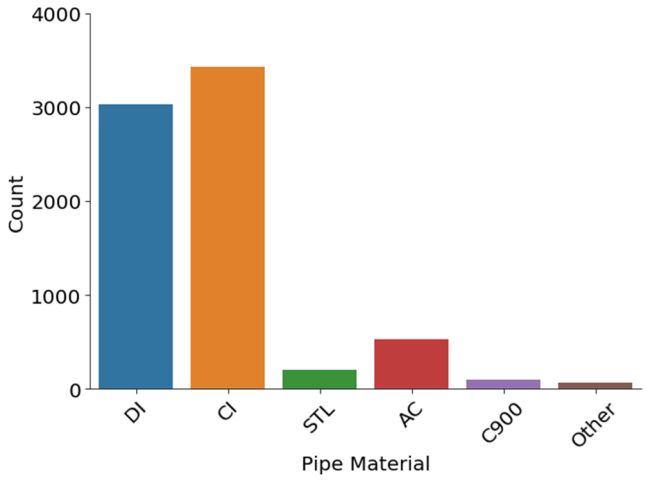
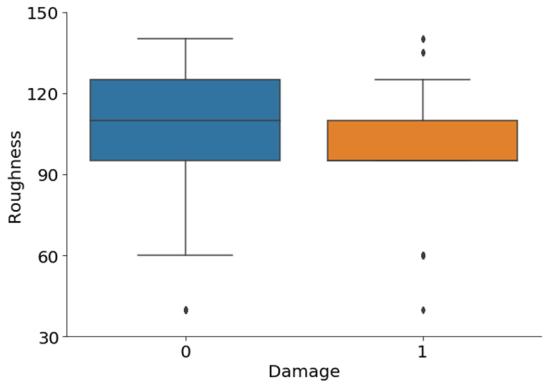
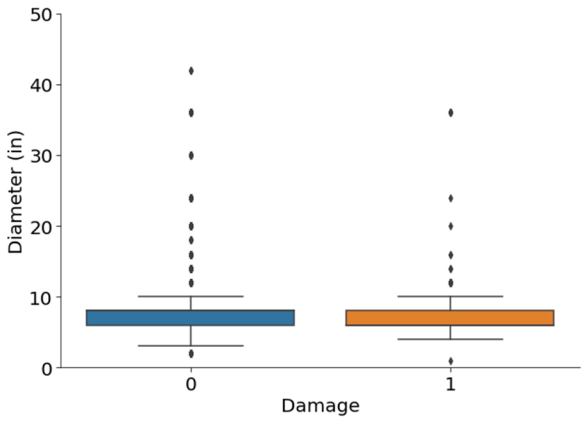


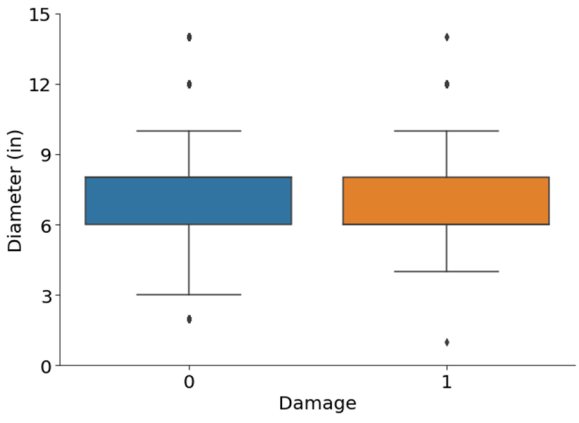
Figure 3.7 Distribution of pipe material type in the Napa water network



(a)



(b)



(c)

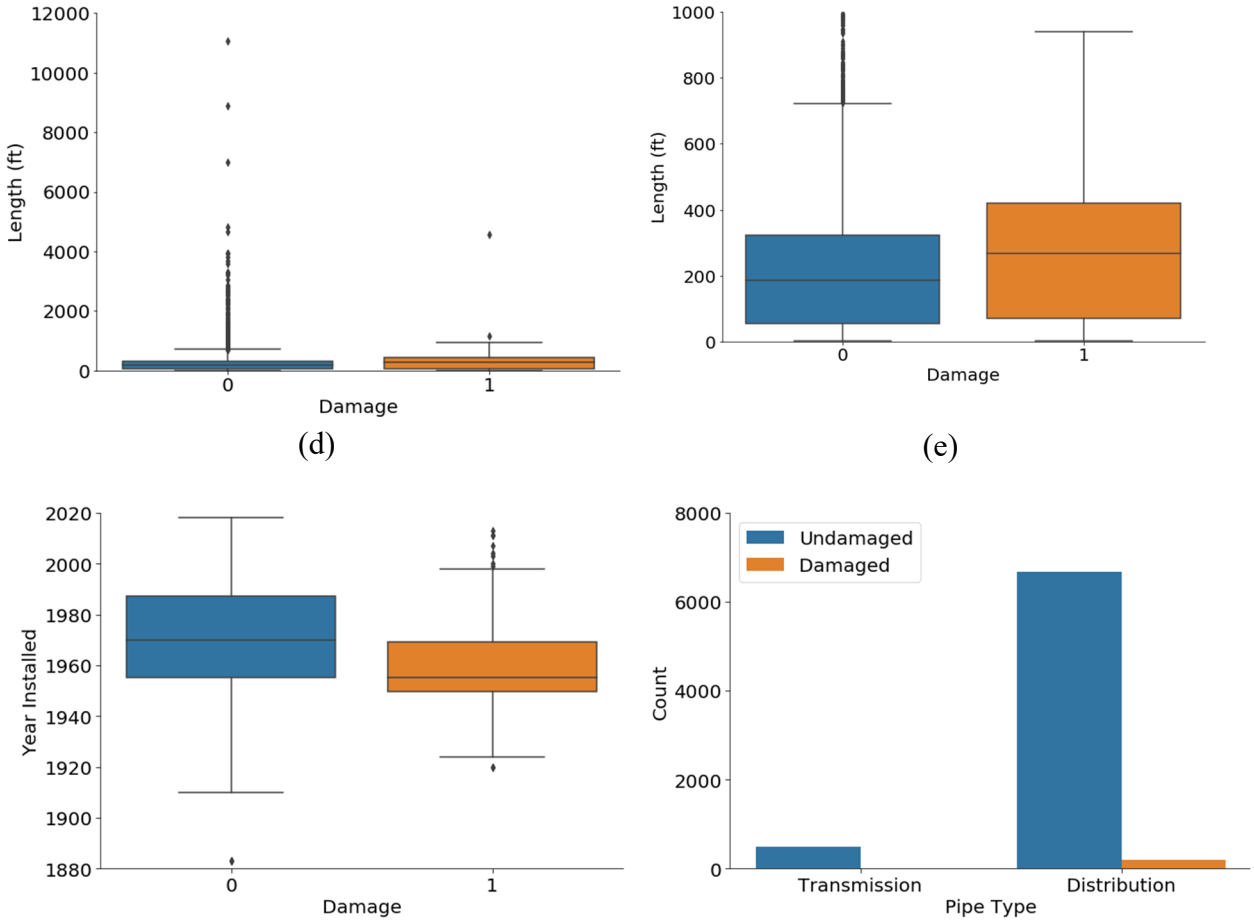


Figure 3.8 Distribution of pipe attributes for the two damage classes: (a) roughness, (b) diameter, (c) diameter, (d) length, (e) length, (f) year of installation and (g) transmission versus distribution

For training, the data is preprocessed as follows: the categorical variables (pipe material and pipe type) are one-hot encoded (a standard approach for preprocessing categorical data) whereby, for each possible value from a categorical variable, a new binary column is created where 1 indicates taking the value and 0 otherwise. The numerical variables are standardized to have zero-mean and unit variance for each variable.

3.5.2 Data Resampling and Model Implementation

The dataset consists of 7162 (97.24%) undamaged and 203 (2.76%) damaged pipe segments, which is highly imbalanced. Without further operation, a statistical model trained on this data is likely to predict all the pipes as undamaged. There are several ways to improve the model development on highly imbalanced classes, such as data resampling techniques (e.g., over/down sampling), algorithmic methods (e.g., boosting based ensemble methods), and loss-sensitive methods (e.g., introducing class weights in computing the loss function). In this study, the Synthetic Minority Oversampling Technique (SMOTE) (Chawla et al. 2002) is adopted to deal with the imbalance issue. The basic idea of SMOTE is to generate synthetic minority instances similar to the real minority instances using the k nearest neighbor method (Chawla et al. 2002).

The dataset is split to 70% training and 30% testing data with stratified sampling such that both training and testing sets preserve the same imbalance ratio (203/7262), which makes the testing set representative for the practical scenario. The SMOTE method is applied for resampling on the training data, and the trained model is tested on the imbalanced testing data.

The neural network model is implemented using PyTorch (Paszke et al. 2019). The model consists of two hidden layers each with 64 units, and leaky rectified linear units for activation. The output layer consists of 2 units (corresponding to the number of classes) and log-softmax activation which outputs the log probabilities of the class being 0 or 1. The training is performed using the Adam optimizer with a learning rate of 0.001, and l_2 regularization is applied with weight 0.01.

3.5.3 Model Performance Evaluation

The evaluation of the imbalanced dataset also requires special treatment. For example, the typically adopted accuracy could be uninformative since the model would achieve excellent accuracy even if it predicts all pipelines as undamaged. Typically, the Precision and Recall metrics

are examined. Precision is the proportion of true positives among those positive predictions

$$Precision = \frac{TP}{TP + FP} \tag{3.7}$$

where TP is the number of true positive instances, and FP is the number of false positive instances. Recall is the proportion of true positives among those actual positives

$$Recall = \frac{TP}{TP + FN} \tag{3.8}$$

where FN is the number of false negatives. In the context of damage prediction, we may put more emphasis on the Recall, since identifying damage is a major concern, and for the most part, over-prediction (false positives) is more tolerable that the opposite (false negatives).

Since the accuracy is no longer effective in evaluating the imbalanced data, the Balanced Accuracy is computed instead, which is the average recall obtained from all classes, and can serve as a measure for accuracy of the imbalanced data:

$$Balanced\ Accuracy = \frac{1}{K}\sum_{k=1}^{K} Recall_k \tag{3.9}$$

where k denotes the kth class.

Finally, we look at the F-beta score which is a harmonic mean of precision and recall

$$F_\beta = (1 + \beta^2)\frac{Precision \cdot Recall}{(\beta^2 \cdot Precison) + Recall} \tag{3.10}$$

When a larger β is used, the score weights more importance on Recall than Precision. Typically, 0.5, 1, and 2 are used as the β value depending on the focus of the evaluation (Baeza-Yates and Ribeiro-Neto 2011). In this study, $\beta = 2$ is used because more focus is put on Recall.

The evaluation result on the testing set is presented in Table 3.1. The model performs well in the Recall score, meaning that the predictions cover most of the true damage. The low precision suggests that the model is likely to over-predict damage for future earthquakes. This is primarily

due to the issue of imbalance. Although the SMOTE is able to balance the distribution in the training data, and thus producing a more reasonable model than directly training on the highly imbalanced data, when applying the model to the imbalanced test data, it is likely to overpredict the damage. Later in this section, we apply the same procedure to a simulated damage dataset that is less imbalanced to demonstrate the capability of the model. In addition, from the model's perspective, the predicted damaged pipes share very similar features, and they are indeed equally likely to suffer from damage if the neural network model is able to capture the underlying true probabilistic model of the damage, but in the 2014 event these pipes did not actually undergo the damage. The model has reasonable performance on balanced accuracy, which indicates it is able to predict most of the instances accurately in each class, i.e., most of the damaged pipes are accurately predicted and most of the undamaged pipes are correctly predicted as well. Although there are some actual undamaged pipes predicted as damaged, the proportion is small among the large number of undamaged pipes.

Table 3.1 Evaluation results on the imbalanced testing dataset

Metric	Recall	Precision	F2 Score	Balanced Accuracy
Value	0.66	0.07	0.241	0.701

3.5.4 Comparison with the Repair Rate Model

The gain a better understanding of the performance of the neural network model compared with currently available models for pipe damage estimation, the repair rate-based model is also accessed. In Tomar et al. (2020), a repair rate model was fitted to the actual damage data from the 2014 South Napa Event, which is a linear model of the form

$$RR\ (\text{per km}) = 0.01053 \times \text{PGV}\ (\text{cm/s}) - 0.33649 \tag{3.11}$$

According to HAZUS-MH model proposed by the American Lifelines Alliance (ALA) (FEMA 2012), for a pipe segment of length L with a given RR, the probability that n repairs are needed is estimated using a Poisson distribution:

$$P(N = n) = \frac{(RR.L)^n}{n!}.e^{-RR.L} \tag{3.12}$$

Previous studies have found that the repair rate model can reasonably predict the total number of damaged components in the network (i.e., the repair/damage rate prediction is accurate), but fails to identify the actual damage state of the individual pipes (Bagriacik et al. 2018). It was also noted in Tomar et al. (2020) that there is inconsistency between the spatial distribution of PGV and the actual damage.

The repair rate model outputs the fragility (probability of failure) for each pipe. To use the model to make a damage prediction, we can place a threshold on the estimated probability and classify a pipe as damaged if its associated probability of failure is larger than the threshold. On the other hand, we can also sample from the probability and obtain a damage realization. Therefore, the repair rate model is evaluated in two ways. First, the prediction is performed by placing a threshold on the estimated probability of failure from the repair rate model. The thresholds are set such that the predicted total number of damaged pipes is (1) the same as the observed damaged (203); (2) the same as total number of damaged pipes predicted by the NN model (1284) (to compare the two models when they predict the same total number of damages). Second, 1000 damage realizations are generated from the fragility curve, and the prediction performance for each realization is evaluated.

Table 3.2 shows the evaluation of the two models on the entire dataset, and the average performance metrics from the 1000 simulations are presented in Table 3.3. It can be concluded

that the NN model significantly outperforms the RR model in all metrics for the 2014 South Napa damage dataset. The precisions from both models are far from ideal, but the neural network model achieves consistent good performance in Recall, F2 Score and Balanced Accuracy whether by thresholding or by simulation.

Table 3.2 Evaluation results for entire dataset by thresholding

Metric	Recall	Precision	F2 Score	Balanced Accuracy
NN Model	0.61	0.10	0.315	0.745
RR Model 1	0.054	0.048	0.053	0.511
RR Model 2	0.270	0.0427	0.130	0.549

Table 3.3 Performance on entire dataset from the 1000 simulations

Metric	Recall	Precision	F2 Score	Balanced Accuracy
NN Model	0.656 ± 0.03	0.062 ± 0.0028	0.225 ± 0.01	0.687 ± 0.015
RR Model	0.03 ± 0.012	0.037 ± 0.014	0.031 ± 0.012	0.504 ± 0.0059

To summarize, the neural network performs well on the Recall, F2 Score and Balanced Accuracy metrics, which are of most concern for this specific application, i.e., to develop a model to have a good estimation of the pipe damage given the pipe attributes, location, site properties and earthquake information. In this regard, the preference would be for the model to provide sufficient coverage of the actual damage. Given the imbalanced data and features, the trained model overestimates the damage to the network. This finding reinforces the well-known fact that the data determines the upper limit of the predictive performance of the statistical and machine

learning model, and the specific algorithm only helps in approaching the upper limit.

3.5.5 Additional Evaluation on a Simulated Damage Dataset

To further evaluate the potential of the neural network model when less imbalanced datasets are included, an experiment is performed on a dataset generated from scenario-based simulation. A $\mathbf{M_w}$ 6.7 event occurred at West Napa fault with an epicentral location of 38°13′12′′ N; 122°18′36′′W is used as the scenario event. The median shaking intensities are computed using the Campbell and Bozorgnia ground motion attenuation model (Campbell and Bozorgnia 2014). The inter- and intra-event residuals are taken to be 0.322 and 0.576 for PGV. 1000 spatially correlated PGV maps are generated using the model by Jayaram and Baker (2009), and 1 damage realization is performed on each PGV map. Next, the single realization which results in the maximum number of damaged cases is selected, which results in 2820 damaged pipes. The train/test split and resampling procedures are performed the same as earlier. Table 3.4 shows the evaluation results of the trained model on the test set. It can be seen that for the more balanced data, the precision achieved by the model is significantly improved and is comparable to the recall. The F2 score and Balanced Accuracy also correspondingly improved.

Table 3.4 Evaluation results on the simulated dataset

Metric	Recall	Precision	F2 Score	Balanced Accuracy
NN Model	0.74	0.68	0.724	0.759

3.6 GNN Based Semi-supervised Pipe Damage Prediction

In this section, we consider the case of utilizing partial information collected about the component damage following the earthquake to propagate and predict damage to all components

in the system. This can be useful for updating plans for inspection and repairs given the known information in the current phase, and further reduce the time and resource needed for the recovery. To this end, a graph neural network model is trained in a semi-supervised learning setting. More specifically, given the features of the components and the damage observations (labels) for a subset of the network, we seek to predict the damage to the entire network while considering the connections among the system components.

3.6.1 Graph Representation and Model Implementation

For illustration, the Napa water network model is developed using the WNTR package (Klise et al. 2017), and the network map is transformed into an undirected graph $G = (N, E)$ where N is the set of nodes which present the junctions of the water network, E is the set of edges which present pipelines of the water network. Because the objective of the study is not to perform inference on the junction nodes, but to learn and update the information of the components (pipelines), the graph representation uses the center of the pipelines as nodes. For this purpose, a line graph transformation is applied to the original graph: for each edge (pipeline) in G, we make the center point as a node in the new graph $L(G)$; for every two edges in G that have a common node, we make an edge between their corresponding nodes in $L(G)$. Fig. 3.9 provides an illustration of the line graph transformation process. Therefore, the new graph $L(G)$ is a representation of the connections among the pipelines.

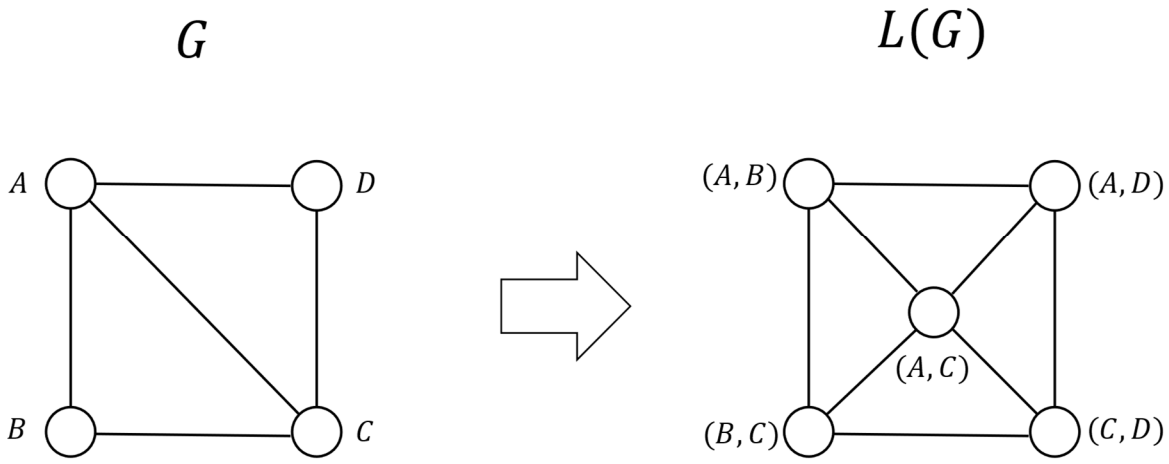


Figure 3.9 Line graph transformation

The GCN is implemented with a single hidden layer with 128 units, the output layer is a log-softmax layer with 2 units (corresponding to the number of classes). The GAT is implemented with two hidden attention layers: the first hidden layer has 128 units and 8 attention heads computing a total of 1024 features; the second hidden layer has 64 units and 4 attention heads computing a total of 256 features; the output layer consists of a single attention head that computes 2 features (corresponding to the number of classes) as the final representation of the graph, followed by a log-softmax activation. Both graph networks are trained using stochastic gradient descent with the Adam Optimizer. The learning rate is set to 0.001 and L_2 regularization of 5e-4 is used. For the semi-supervised learning, full-batch training is performed (i.e., all data are fed to the network in a single pass), and 5000 epochs are used for training.

3.6.2 Evaluation of GNN Performance

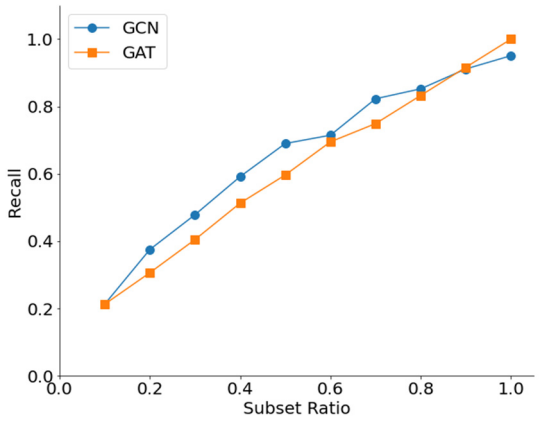
A set of experiments are performed where subsets containing damage labels from 10% to 100% are generated for training, and the imbalance ratio of damage/undamaged are preserved for all subsets. The damage prediction for the entire network is performed in each case. For evaluation, the metrics used in the previous section (Recall, Precision, Balanced Accuracy and F_2 score) are

adopted. Additionally, the "Additional Damage Predicted" metric is introduced and computed as the difference between the correctly predicted and known damages normalized by the known damages, which provides a measure of how the model can produce additional information of damage.

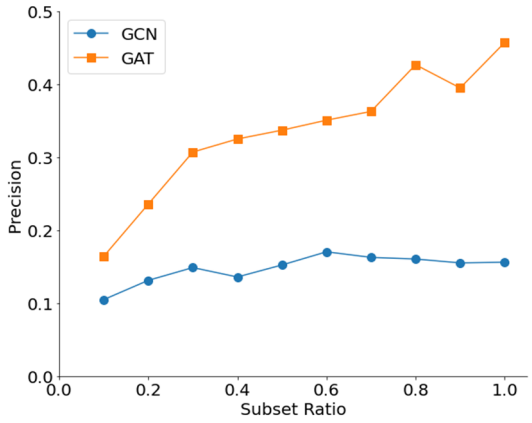
The evaluation results are shown in Fig. 3.10. The performance of the two models are comparable in Recall, and the score increases almost linearly with the subset ratio. The models are also equally good in Balanced Accuracy. It is worth noting that, from Fig. 3.10 (b), the GAT performs significantly better in Precision than the GCN. In addition, the Precision of the GAT continuously improves when the size of the subset of known damage increases, while the Precision of the GCN only improves slightly and remains below 0.2. This is attributed to the advantage of the attention mechanism which considers different weights from the neighbors. As a result, the F2 scores from the GAT are also much higher than those from GCN. Starting with a small subset of 10%, the models can already learn quite helpful information, achieving approximately 0.6 in Balanced Accuracy. Using a 10% subset of the actual damage for training, both models are able to capture an additional 10% of actual damage. Using training subsets with ratios ranging from 20% to 50%, the GCN is able to capture approximately 20% additional damage in each case, while the percent of additional damage the GAT model able to capture remains around 10%. Using more than 60% of the damage, the additional damage the model able to capture starts to decrease. This means that the learning is more effective in the early stages. To sum up, the models are able to provide informative predictions for at least 5% more damage at each stage before 80% damage information is collected.

If the prediction on the entire dataset is used as the standard, using a 50% subset, the GNN models are able to achieve Recall scores comparable to those in NN models (Table 3.2), and have

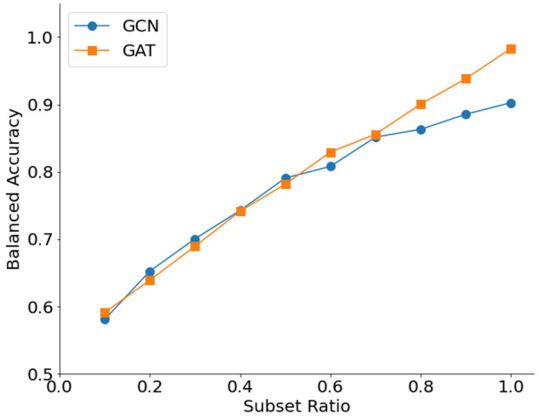
much better Precision. Using a subset of 70%, which is what is used in supervised learning, the models are able to achieve approximately 0.75 in Recall, and the GAT reaches 0.36 in precision, which is much better than the supervised learning model based on Table 3.3. This suggests that if we use the semi-supervised learning approach when partial information following a specific earthquake is given, the GNN based approach is better than using a NN model trained on previous earthquakes, since the NN model trained on the same data could not achieve the same performance as the GNN.



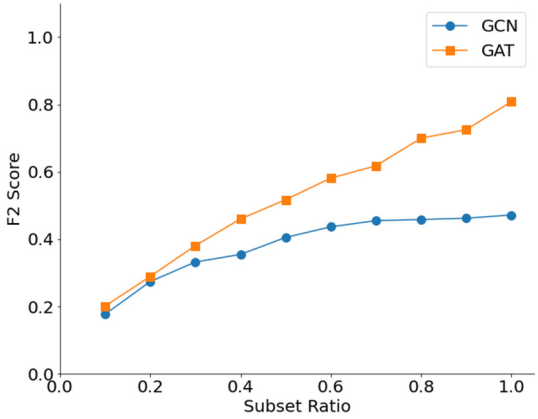
(a)



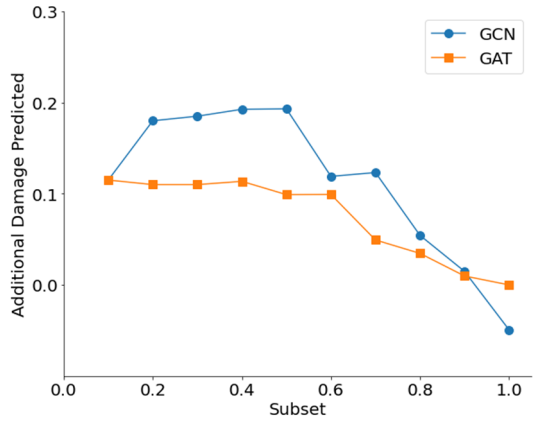
(b)



(c)



(d)



(e)

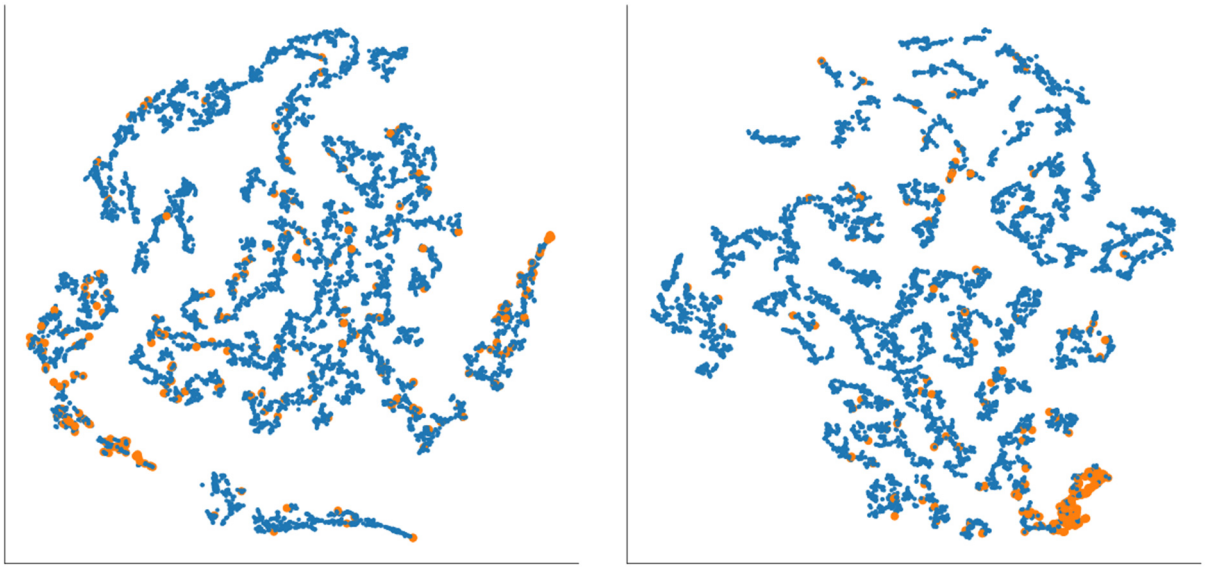
Figure 3.10 Evaluation results for the GNN models trained using different subsets of observations (labels)

3.6.3 Learned Graph Representation from the GNN Models

As noted in Section 3.3, the GNNs generate a new representation over the nodes of the graph, which are then used for classification. In many cases, the node representations (or embeddings), which are one of the final products of a GNN model, can be used to further understand the data. Therefore, we plot the node representations from the GAT models to demonstrate how it works to produce the output.

To illustrate how the GNN model learns to separate the damaged from undamaged instances, Fig. 3.11 shows the representations of the output graph using t-distributed Stochastic Neighbor Embedding (t-SNE) (Van Der Maaten and Hinton 2008), a method for visualization. It needs to be noted that the 2D representation is unrelated to the topology of the original network graph. The blue color represents the undamaged data, and the orange color represents the damaged data. It can be observed that when trained with a 10% subset, the damaged nodes are scattered, and there is no obvious pattern to distinguish them from the undamaged nodes; When trained with a 50% subset, we can see a cluster of damaged nodes forms clearly on the bottom-right corner of

the graph, which indicates that the model is able to find a pattern that can be used to separate the two classes of nodes. However, we also observe that the cluster is overlapped with some nodes from the undamaged class. Also, there are some damaged nodes scattered in different parts of the network. When all the data is used, the damaged nodes form a cluster on the bottom of the output graph. Again, the cluster is overlapped with undamaged nodes. Therefore, we can further conclude that from the model's perspective, these nodes (representing pipelines) clustered together are very similar, and they would be classified as damaged when we place a classification rule on the output graph.



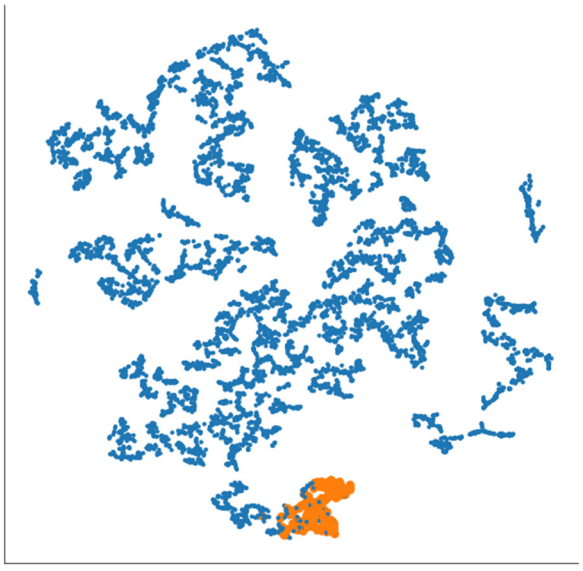
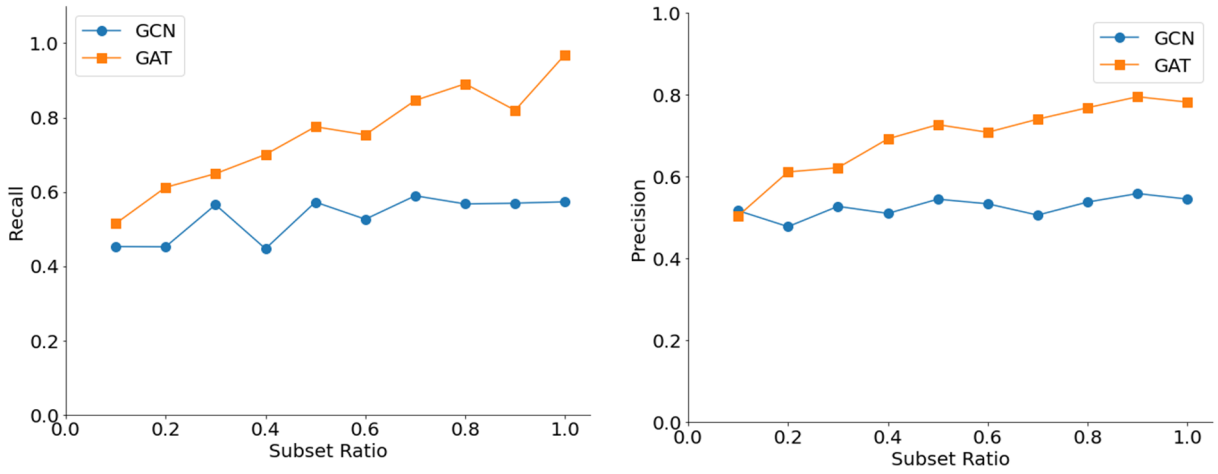


Figure 3.11 Graph representations from the models trained using (a) 10%; (b) 50%; and (c) 100% subsets

3.6.4 Additional Evaluation on a Simulated Damage Dataset

The simulated dataset described in Section 3.5.5 is used again for additional evaluation of the GNN models, using different subsets of observations (labels). Fig. 3.12 shows the evaluation results. It can be seen that all scores are improved for the balanced data, and the Precision and Recall scores are comparable and mostly higher than 0.5.



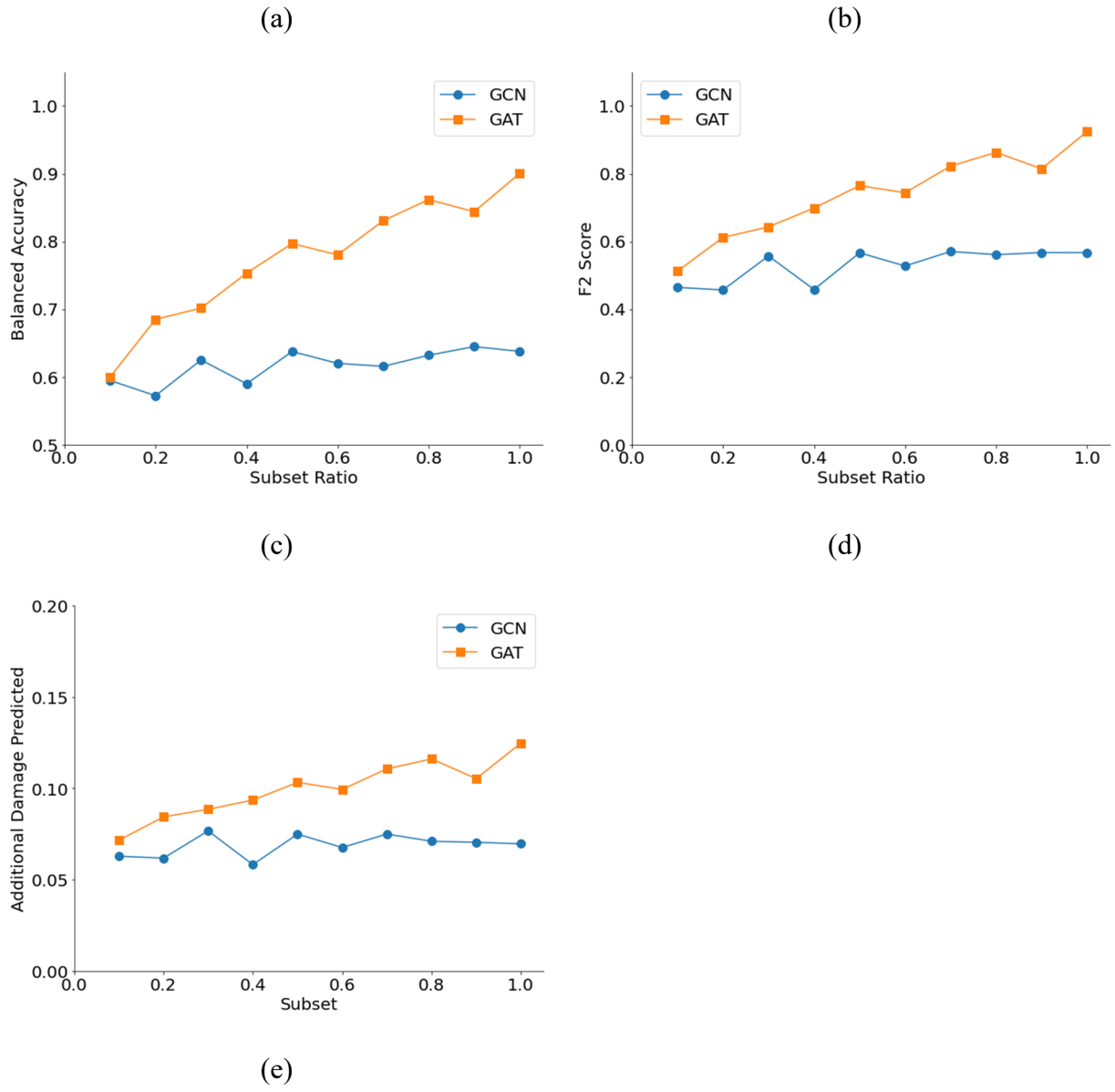


Figure 3.12 Evaluation results for the GNN models trained using different subsets of observations (labels) from the simulated dataset

3.7 Conclusion

This study presented a general methodology for improving pre-event component damage assessment of distributed infrastructure systems using information collected during inspections. A neural network model was trained from historical earthquake data to estimate the distribution of

damage immediately following the earthquake (i.e., the pre-event assessment), although, it was noted that this step can also be performed using a fragility-based model. Semi-supervised learning was then performed using a graph neural network model to update the damage estimate given the information incrementally collected during the inspection process.

A case study was performed on the pipe network for the City of Napa water distribution system, using a component-damage dataset from the 2014 earthquake. For the pre-event assessment, the data was randomly split into 70% training set and 30% test set, the neural network was trained on the training set and evaluated on the test set, considering four metrics: recall, precision, F2 score and balanced accuracy. The evaluation results suggested that the neural network performs better than the repair rate fragility-based model fitted on the same dataset in classifying individual pipeline damage. Among the evaluated metrics, the recall, F2 score and balanced accuracy all have reasonable values, but the precision was relatively low, which was due to the high imbalance of the classes in the dataset. The lower precision suggested that the model is likely to overpredict the number of damaged components in future earthquakes. For post-event assessment, the Graph Convolutional Network (GCN) and Graph Attention Network (GAT) were utilized to learn to predict the damage map for the entire water network given the damage information of 10% to 100% subsets i.e., semi-supervised learning. The results showed that GAT had much better precision than the GCN, which can be attributed to the attention mechanism. From the prediction increment metric, it suggested that when the GNNs are used during the post-earthquake recovery, an additional 10% of damage is captured given the known information collected during inspection.

The study demonstrated the applicability of graph neural network models for real-time updating of component-level seismic damage in distributed infrastructure systems. The proposed

methodology can be extended for assessment on other types of natural hazards and infrastructures. Furthermore, the study showed their potential in informing and updating resource-allocation and recovery plans in the post-event environment. Several limitations are acknowledged which require special attention for future studies. The performances of the NN model, although better than the commonly used repair rate model, still has significant room for improvement. Future studies should focus on how to improve the learning with imbalance, which is valuable especially for the infrastructure damage estimation problem, because most of the real data in this aspect is highly imbalanced. Because of data availability, we did not study the performance of the models on other events and system types. Another related future research thread couple explore the use transfer learning to address the general lack of data within each system type. Lastly, the current study focused on predicting binary classes (damaged and undamaged). Therefore, future studies should explore the viability of the GNN when more than two damage classes are present (e.g., multiple levels of damage).

4. A Spatially Explicit Post-earthquake Recovery Model for Distributed Infrastructure

4.1 Introduction

Post-earthquake recovery models can provide useful decision support for pre-event planning (Kang et al. 2018). The increasing attention on infrastructure resilience has spurred calls for the development of models to quantify the post-earthquake recovery of infrastructure systems.

In the past two decades, researchers have been studying post-event recovery models for built infrastructure (e.g., buildings and lifelines). Various methods have been proposed. For example, Nejat and Damnjanovic (2012) proposed a multi-agent framework to model the recovery accounting for neighbors' dynamic interactions. Kang et al. (2018) used stochastic process-based model for the housing recovery in 2014 south Napa earthquake, with time parameters predicted by the random forest machine learning model. Tomar et al. (2020) used the discrete event simulation method to model and hindcast the recovery of water distribution system following the 2014 Napa earthquake.

Most of the previously proposed stochastic process-based recovery models establish the relationship between recovery time parameters and various explanatory variables, while the spatial correlations of the recovery time for different components over the region is often not explicitly modeled. A spatially explicit recovery model can better capture recovery at the system or regional level. Gaussian Process (GP) is a stochastic process-based machine learning model which can capture the spatial dependence over the responses. Previous studies have demonstrated the capability of the GP model in capturing the spatial correlation. For example, Sun et al. (2018) studied the interpolation of structural response demands across buildings using the Kriging model (an application of the GP model in spatial statistics); Sheibani and Ou (2020) used the GP model

to infer the damage to buildings over a region following earthquakes.

This study presents a methodology that uses a GP model to estimate the spatial distribution of the recovery time of the components over the infrastructure system. The remainder of this chapter is presented as follows. Section 4.2 provides a review of the Gaussian Process; An overall description of the 2014 Napa earthquake dataset used in this study, including the key explanatory variables and observed recovery trajectory is presented in Section 4.3; Section 4.4 presents the development of a GP based spatially explicit recovery model, and discusses the alternative ways of using the model and the effect of different kernels in the performance of the GP model; Section 4.5 studies the capability of the learned GP in estimating the recovery trajectory for other (future) events. Finally, the contributions of this study are summarized, and the limitations are discussed in Section 4.6.

4.2 Review of Gaussian Process and Machine Learning

Gaussian Process is a stochastic process of multivariate normally distributed random variables (Seeger 2004). It is a non-parametric model based on Bayesian inference. The Gaussian Process Regression model defines a distribution over unknown functions, where the values of the functions at any finite number of points follows joint Gaussian distribution (Seeger 2004; Sheibani and Ou 2020; Tomar and Burton 2021b)

$$f(x) \sim GP\big(m(x), k(x, x')\big) \tag{4.1}$$

where x is the input vector, $m(x) = E[f(x)]$ is the mean function, and $k(x, x') = E[(f(x) - m(x)(f(x') - m(x'))$ is the covariance or kernel function. Given a set of training data $\{(x_i, y_i) \mid i = 1 \ldots, n\}$, we represent the features x_i as the matrix X, given the new (test) input matrix X^*, the training outputs y and the test functions f^* follows joint distribution

$$\begin{pmatrix} y \\ f^* \end{pmatrix} \sim N\left(0, \begin{bmatrix} K(X,X) + \sigma_n^2 I & K(X,X^*) \\ K(X^*,X) & K(X^*,X^*) \end{bmatrix}\right) \tag{4.2}$$

where $K(X, X^*)$ denotes the kernel (covariance) between training and testing data. Therefore, the posterior predictive distribution of f^* can be derived as

$$f^*|X, y, X^* \sim N(E[f^*], Cov(f^*)) \tag{4.3}$$

where $E[f^*] = K(X,X^*)[K(X,X) + \sigma_n^2 I]^{-1} y$, and $Cov(f^*) = K(X^*,X^*) - K(X,X^*)[K(X,X) + \sigma_n^2 I]^{-1} K(X,X^*)$.

Selecting an appropriate form of the kernel function is important for the specific application of interest. In this study, several commonly used kernel functions are assessed and compared. The radial-basis function (RBF) if defined as (Seeger 2004)

$$k(x, x') = \sigma_f^2 \exp\left(-\frac{|x - x'|^2}{2l^2}\right) \tag{4.4}$$

where l is the length scale parameter, σ_f^2 is the signal variance which controls the scale of the covariance.

The Matern kernel is defined as (Seeger 2004)

$$k(x, x') = \sigma_f^2 \frac{2^{1-\nu}}{\Gamma(\nu)} \left(\frac{\sqrt{2\nu}|x-x'|}{l}\right)^{\nu} K_\nu \left(\frac{\sqrt{2\nu}|x-x'|}{l}\right) \tag{4.5}$$

where $K_\nu(\cdot)$ is a modified Bessel function $\Gamma(\cdot)$ is the Gamma function.

The Rational Quadratic (RQ) kernel is defined as (Seeger 2004)

$$k(x, x') = \sigma_f^2 \left(1 + \frac{|x - x'|^2}{2\alpha l}\right)^{-\alpha} \tag{4.6}$$

where α is a scale mixture parameter.

4.3 Description of the data

This study uses the water distribution system that serves the City of Napa, California as the

test bed. The relevant information is provided by the Napa Water Division (NWD). The Napa water network serves a population of approximately 88,000 people. The system includes 3 water treatment plants, approximately 612 km (380 mi) of pipeline (7365 pipe segments), 15 storage tanks with a total of 113,562 m^3 storage, and 10 pump stations. Additional details of the topology and components that encompass the system are provided in (Napa Water Division 2017; Tomar et al. 2020).

The selected data used is the damage and recovery data from the 2014 south Napa earthquake, which contains a total of 250 pipeline leaks on 203 pipe segments of the water system. Fig. 4.1 shows the spatial distribution of the pipe damage.

Figure 4.1 Spatial distribution of the pipe damage in the Napa water network in the 2014 event

4.3.1 Observed Recovery Trajectory

Fig. 4.2 shows the recovery trajectory over time for the 2014 event. As the key component is the pipes, this study defines the recovery in terms of the number of pipe repairs completed. It can be observed that there is a steep slope in the initial phase of the recovery. Pipe leaks were repaired

relatively quickly with 120 leaks fixed in 5 days (Napa Water Division 2017; Tomar et al. 2020). At 20 days after the earthquake, approximately 70% of the pipelines has already been repaired. The rate of repair then decreases significantly, and continue to decrease over time, due to the reduction of inspection and repair crews. All pipes are repaired after 156 days.

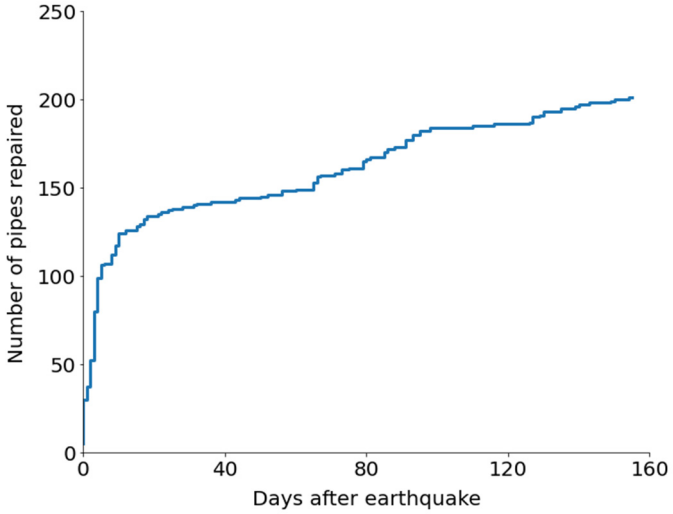


Figure 4.2 Observed recovery trajectory from the 2014 event

4.3.2 Target and Explanatory Variables

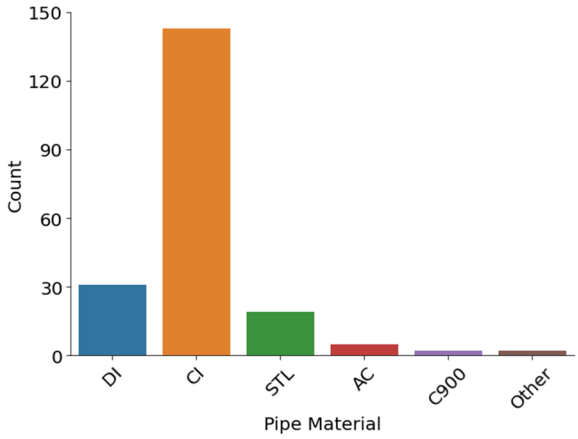
The post-event recovery model starts with a physical damage assessment (Kang et al. 2018). The damage information is treated as an input to the recovery model. Therefore, given the damage information, the earthquake information does not need to be taken as input for the model. The features included for the model are the location information of the components: latitude and longitude, which are used to inform the spatial correlation of the recovery; and several key characteristics of the individual component (pipe). For the pipe system used in this study, the selected pipe attributes include the material, roughness, diameter and length. The detailed description of the pipe attributes can be referred to Chapter 3.

Fig. 4.3 shows the distribution of the features for the damaged pipe subsets. The distributions on the entire set is described in Chapter 3. To be consistent with the distribution over

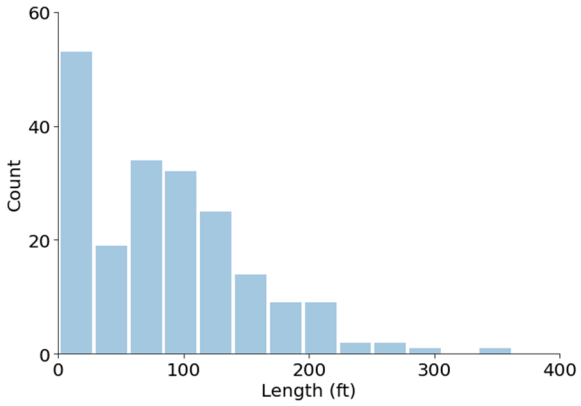
the entire system, the pipe material types with less than 100 pipes are lumped together to the "Other" category. The final types used are Ductile Iron (DI), Cast Iron (CI), Steel (STL), Asbestos Cement (AC), C900 and Other. It can be observed that most damaged pipes are made of Cast Iron (CI), Ductile Iron (DI) and Steel (STL). The pipe lengths range from 0 to 1200 feet. Most of the pipes have a diameter less than 15 inches. The values of the Hazen-Williams roughness coefficient (unitless) for most pipes are larger than 100.

The target (label) is the recovery time, which is taken as the number of days from the date of the earthquake to the date when the pipe is repaired.

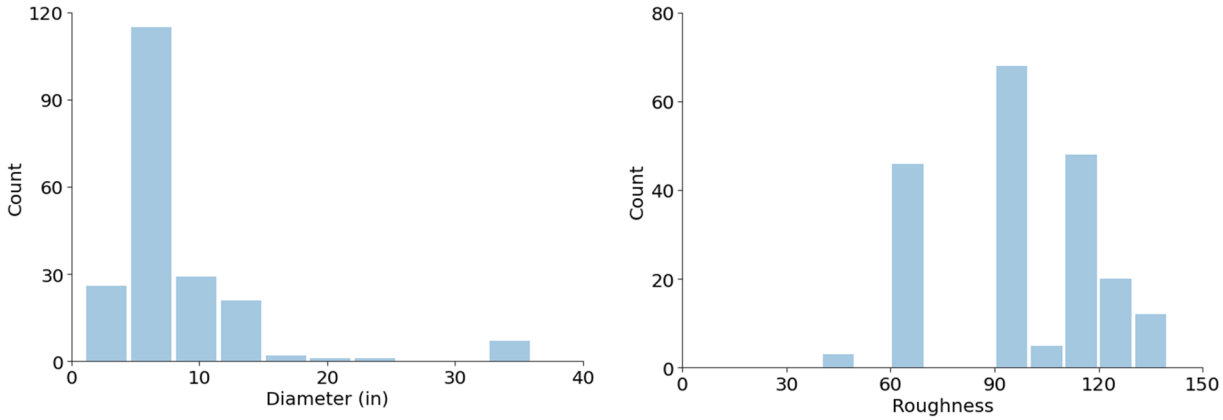
Hence, the objective of the recovery model is to establish a statistical relationship between the input features and the repair completion time, while developing the spatial distribution of the repair over the considered region. The underlying assumption is that components that are closer in distance would tend to have similar recovery times.



(a)



(b)



(c) (d)

Figure 4.3 Distribution of key attributes for the damaged pipe subset: (a) material, (b) length, (c) diameter and (d) roughness

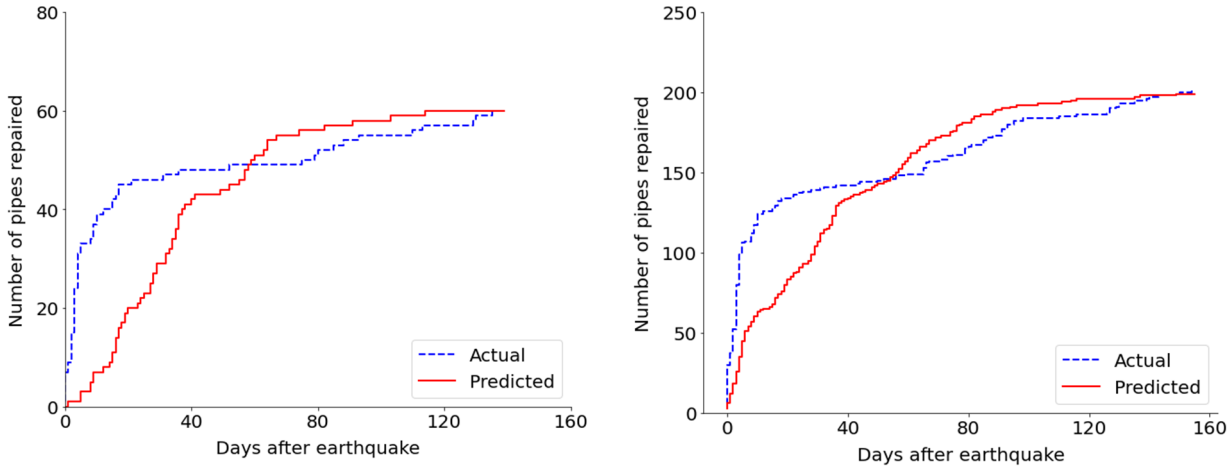
4.4 Model Development

In this section, the goal is to develop a general model for predicting the spatial distribution of the recovery time over the network and recovery trajectory of the system in future earthquakes. We do this by building g a Gaussian Process model which captures the statistical relationship between the recovery time and explanatory variables influential to the pace of the recovery, while accounting for the spatial distribution of the repair time over the region.

4.4.1 Gaussian Process Model with RBF kernel

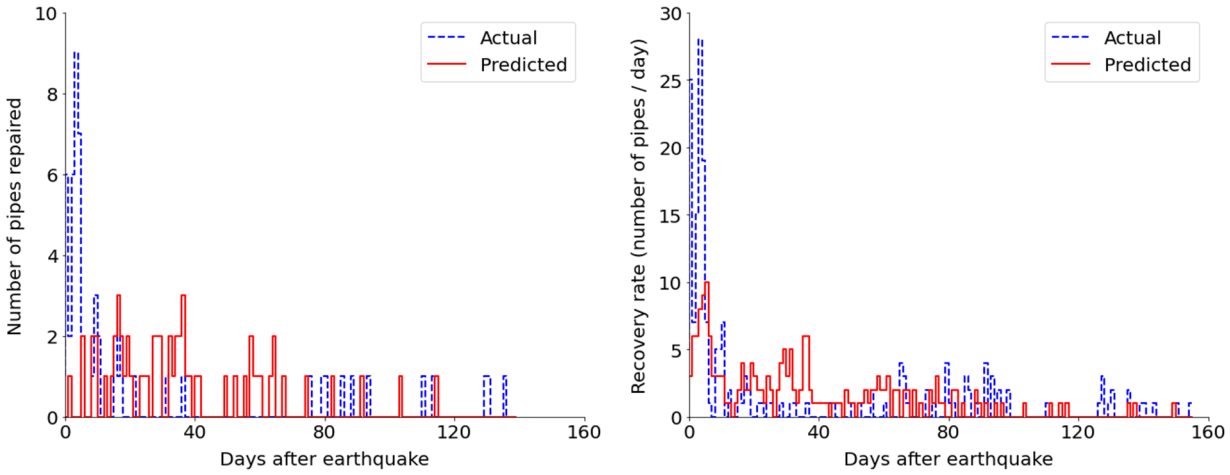
A GP model with the most commonly-used RBF kernel is first developed. The model is implemented using the scikit-learn machine learning library (Pedregosa et al. 2011). The L-BGFS algorithm is used to optimize the kernel parameters. For the learning purpose, both the input features are normalized to have zero mean and unit variance before training. The damage dataset is randomly split into 70% training set and 30% testing set. A GP model is developed on the training set and evaluated on the test set to demonstrate its capability in predicting unseen data.

Given the developed posterior distribution over the test functions f^*, we first consider the mean $E[f^*]$ as the prediction $\hat{y}$. Given the mean prediction, the corresponding recovery trajectory is generated and then compared to the actual recovery curve, as shown in Fig. 4.4. The prediction captures the overall trend of the recovery trajectory, including the gradual reduction in the rate of recovery with time. However, the model underestimates the repair rate in the early stage and underpredicts the recovery level up to about the time when 70% of the pipes have recovered. Fig. 4.5 plots the recovery rate in terms of the number of pipes repaired per day. It can be observed that the model significantly underestimates the rate in the first 10 days. Nevertheless, the predicted recovery rates after 10 days are generally in the same level with the actual rates.



(a) (b)

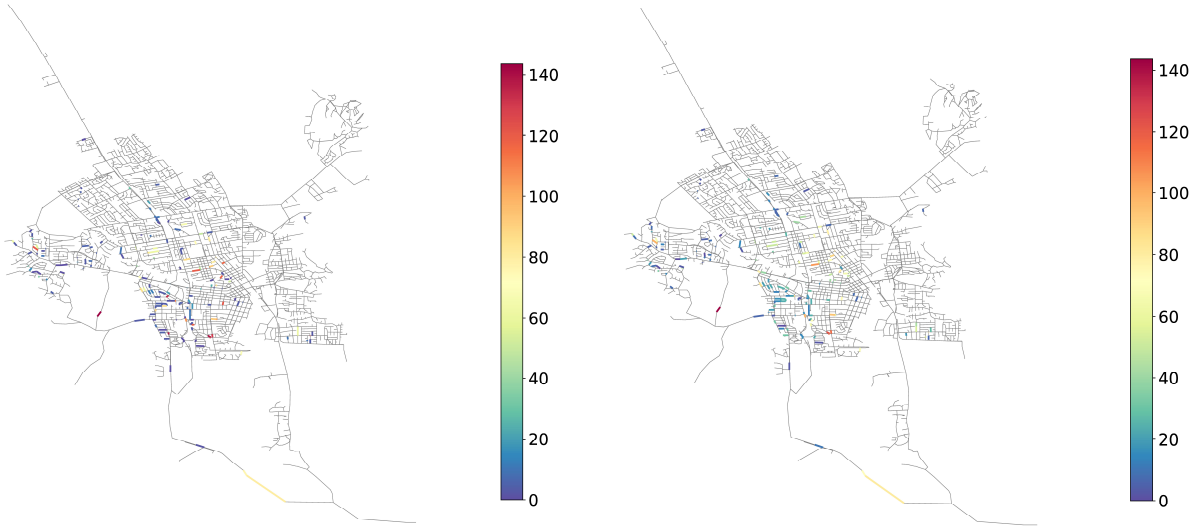
Figure 4.4 Predicted versus actual recovery trajectory: (a) test set; (b) entire set



(a) (b)

Figure 4.5 Predicted versus actual recovery rate: (a) test set; (b) entire set

Fig. 4.6 (a) and Fig. 4.6 (b) show the recovery time for the damaged pipelines in the 2014 event and the predictions from the GP model, respectively. The spatial distribution of the recovery time is captured well by the GP model. The developed spatial distribution can be used for estimating the recovery time for future events with scale and type similar to the 2014 south Napa earthquake.



(a) (b)

Figure 4.6 Spatial distribution of the recovery time (days)

4.4.2 Sampling from the Developed Recovery Model

As the GP model develops a probabilistic distribution of the recovery time, the uncertainty in the recovery can be observed by generating multiple realizations of the recovery trajectory. In this study, 1000 realizations are obtained by sampling from the posterior distribution from the GP model, as shown in Fig. 4.7. The grey lines show individual recovery trajectories, and the black line represents the average recovery curve. Fig. 4.8 (a) shows the probability mass function (PMF) and fitted normal probability density function (PDF) for the recovery time, and the cumulative distribution of the total recovery time is shown in Fig. 4.8 (b). The average total recovery time is 150.8 days, and the standard deviation is 15.9 days.

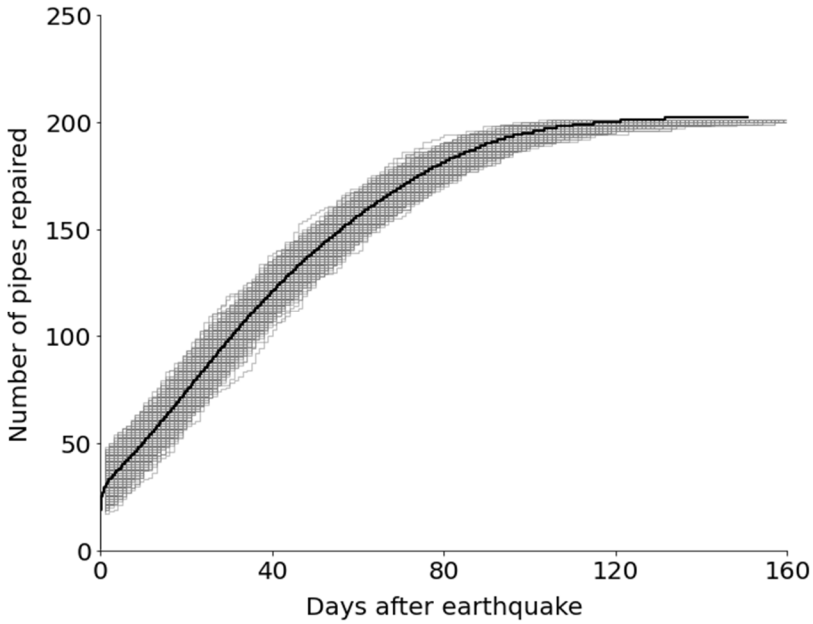
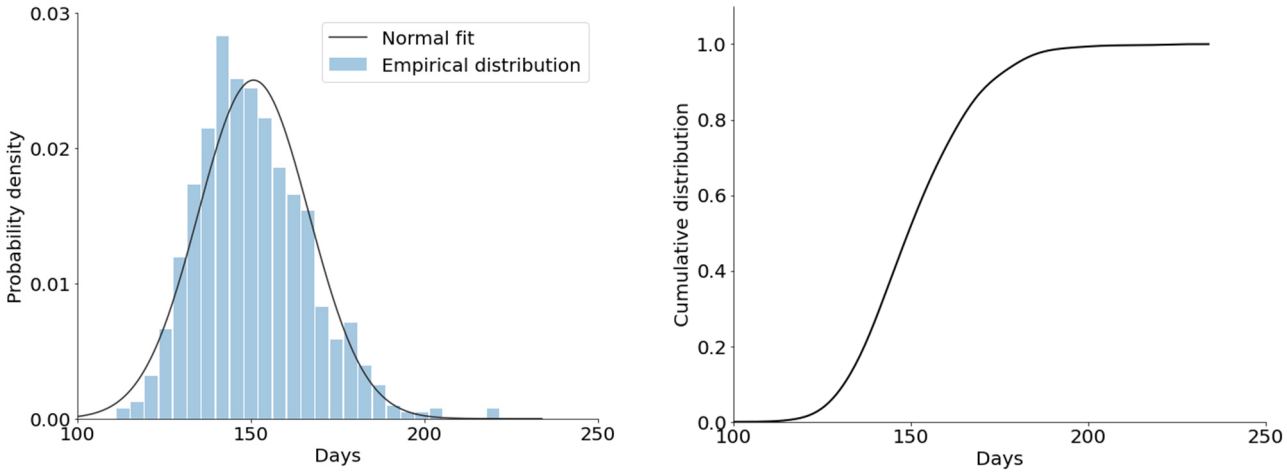


Figure 4.7 Sampled recovery trajectories from the GP model



(a) (b)

Figure 4.8 Distribution for the time to repair all pipes: (a) probability density and (b) cumulative distribution

4.4.3 Comparison of the Effect of Different Kernels

To investigate the influence of the selection of kernel functions on the performance of the GP model, a comparison study is performed. RBF, Matern and RQ kernels are considered for the

comparison. GP models developed with different kernels are evaluated using 10-fold cross-validation. Specifically, in K-fold cross validation, the dataset is randomly split into K equal-sized sub-folds and a train-and-validate process is performed K times: in the k^{th} round, the k^{th} sub-fold is used as a validation set to calculate the performance of the model trained on the other (K-1) sub-folds of data. The root mean squared error (RMSE) is used to evaluate performance.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{4.7}$$

where y_i and $\hat{y}_i$ are the actual and estimated response of the ith data point, respectively. The median absolute relative deviation (MARD) is also used as a performance metric

$$MARD = Median\left(\frac{y_i - \hat{y}_i}{y_i}\right) \tag{4.8}$$

The cross-validation results are shown in Fig. 4.9. It can be concluded that, the performance of the RBF kernel is similar to the Matern kernel for the dataset used. The RQ kernel has better average performance in $RMSE$ than the other two kernels, while having larger dispersion across the 10 folds. For the $MARD$ score, the RQ kernel has both lower average value and lower dispersion compared to the other kernels.

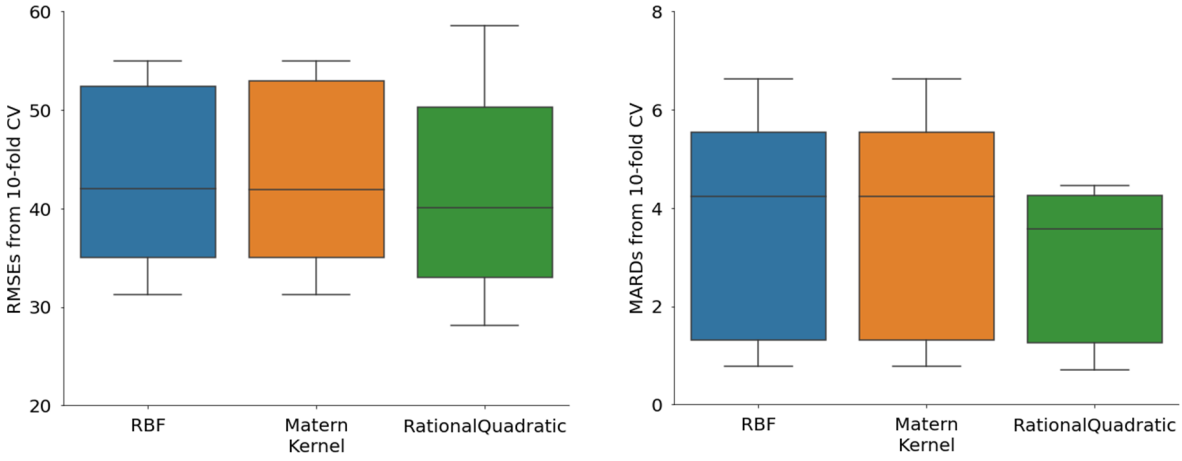


Figure 4.9 Performance of GP models with different kernel from 10-fold cross-validation

4.4.4 Within Event Prediction Using Location Parameters

The previously developed GP model includes the location and features of the individual components, which is useful for extrapolating the recovery estimate to future events and other network systems.

On the other hand, within the scope of estimating/interpolating the recovery time of individual components for the same event using available recovery time information, it is possible to consider the spatial correlations only, and the input features can include the location parameters. For example, during the recovery process, the repair time of a subset of the damaged pipes have been collected, we can use the spatial correlation to interpolate the repair time to other damaged pipes.

Therefore, a GP model is fitted using only the latitude and longitude of the pipes. The train/test split scheme is the same as in Section 4.4.1. The generated recovery trajectories are shown in Fig. 4.10. It can be observed that when considering only the spatial correlation of the recovery time, the prediction on both test set and entire set is better in the early stages of the recovery but overestimates the recovery rate in the later stage.

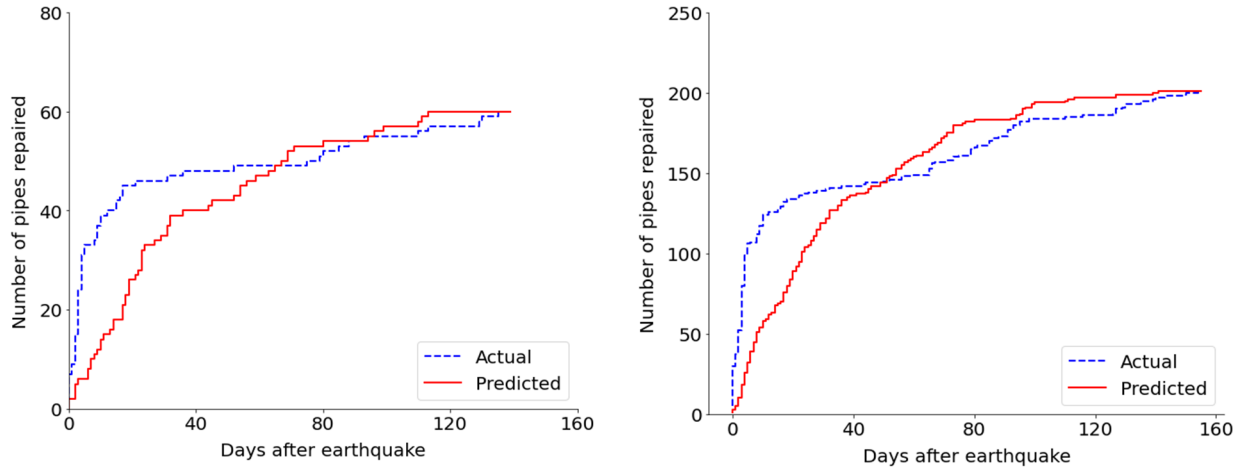


Figure 4.10 Predicted versus actual recovery trajectory using only location parameters: (a) test set; (b) entire set

4.5 Recovery Estimation for Future Earthquakes

For the developed GP recovery model, its ability to generalize to unseen data has been demonstrated in Section 4.4. For the purpose of applying the model, however, it is desirable to examine the performance of the model in future events, where the data does not necessarily follow the same distribution to the data we trained the model on. Due to data availability, this can be done on simulation of an earthquake that represents potential future event. A $\mathbf{M_w}$ 6.7 earthquake occurring on the West Napa fault with an epicentral location of 38°13′12″N; 122°18′36″W is used as the scenario event, as mentioned in Section 3.5.5. The Campbell and Bozorgnia (Campbell and Bozorgnia 2014) ground motion attenuation relationship is used to obtain the median shaking intensities. The inter- and intra-event residuals for PGV are taken as 0.297 and 0.578 cm/s, respectively. Spatially correlated shaking intensities are generated using the model by Jayaram and Baker (2009). 1000 damage maps are generated from the spatially correlated shaking intensities. Additional details about generating the shaking intensities can be found in (Tomar et al. 2020).

For the 1000 simulated maps, the one with the maximum number of damaged pipes is selected for demonstration. A GP model is then fitted to the simulated dataset, using procedures described in Section 4.4. Fig. 4.11 shows the predicted versus actual recovery trajectory from the simulated data.

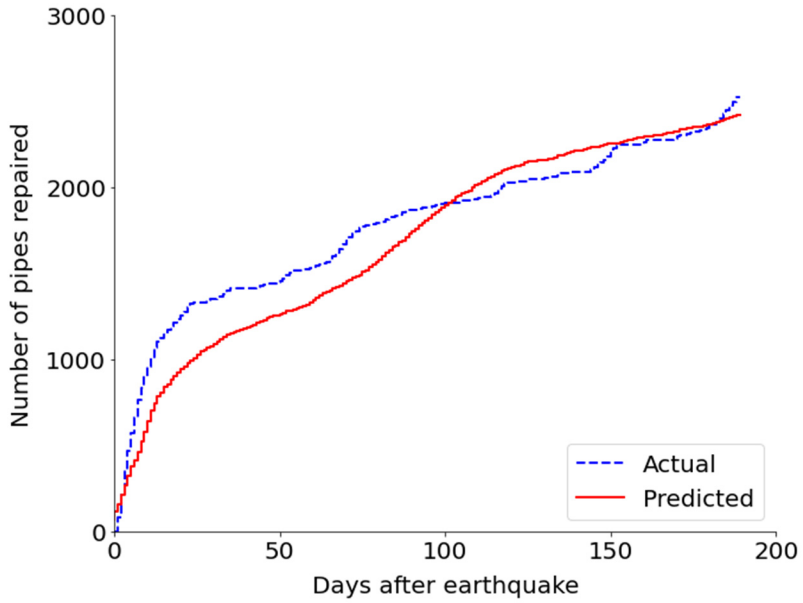
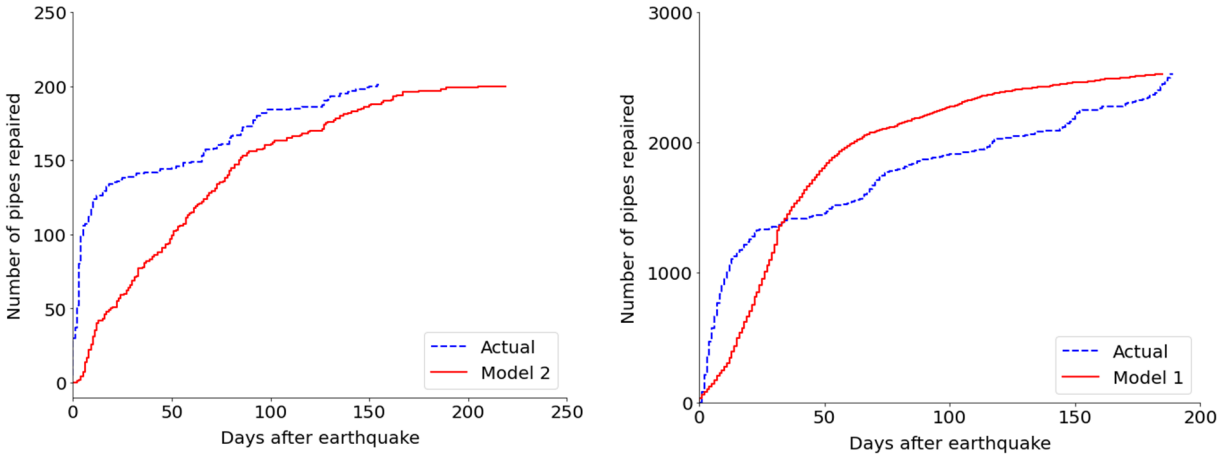


Figure 4.11 Predicted versus actual recovery trajectory for the simulated M_w 6.7 event

The capability of the GP models for predicting future earthquakes is examined as follows: we denote the fitted model in section 4.4 as Model 1, and the model fitted in this section as Model 2. Then we estimate the recovery trajectory of the 2014 Event using Model 2 and estimate the recovery trajectory of the simulated event using Model 1. To consider the effect of the different available inspection and repair sources, the initial number of inspection and repair crew are added to the features.

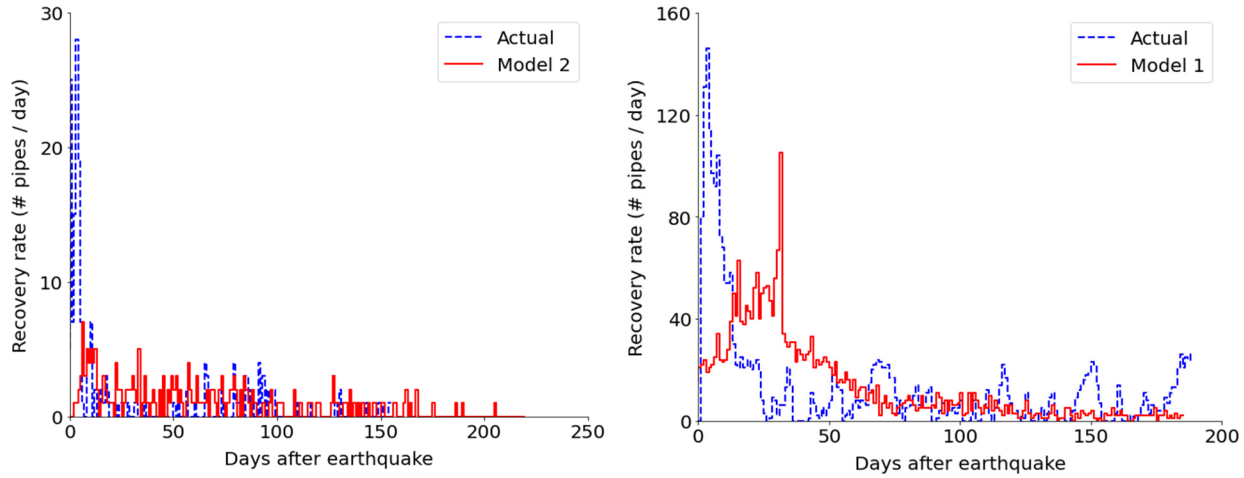
Fig. 4.12 (a) shows the restoration trajectories for the 2014 event. Model 2 underpredicts the recovery rate in the initial phase (first 10 days), which is similar to the issue mentioned in Section 4.4. In addition, the model overestimates the recovery time of several pipes, which results

in the longer total recovery time. However, the estimated recovery rate is close to the actual after 90 days following the earthquake. Fig. 4.12 (b) shows the restoration curve of the simulated 6.7 event. Model 1 underestimates the recovery rate in the first 30 days and overestimates the recovery rate in 30 days to 60 days. To sum up, the model developed on a large number of damaged pipes tends to overestimate the total recovery time for realistic events with less damage. Nevertheless, the estimated recovery rate is relatively accurate.



(a) (b)

Figure 4.12 Predicted versus actual recovery trajectories: (a) 2014 event; (b) simulated M 6.7 event



(a) (b)

Figure 4.13 Predicted versus actual recovery rates: (a) 2014 event; (b) simulated M 6.7 event

4.6 Conclusion

This study developed a spatially explicit recovery model for the post-earthquake restoration of distributed infrastructure. To encode the spatial correlation of the recovery time over the system components, the Gaussian Process (GP) was utilized for modeling.

The GP model took the location information of the components and several key attributes as input, and output the spatial distribution of the recovery time over the network. The evaluation results showed that the model was able to capture the overall recovery trajectory for the 2014 event, but underestimated the high recovery rate observed in the initial phase of the recovery. An alternative way of utilizing the GP model was proposed that used only the location parameters, to represent the case of interpolating the recovery time for the network within the same event given available recovery time information for some damaged pipelines.

To further demonstrate the capability of the GP model for predicting the recovery for future earthquakes, a hypothetical $\mathbf{M_w}$ 6.7 scenario was simulated. A GP model was fitted to the simulated

data. Next, the prediction from the model fitted to the simulated data was applied to the 2014 event, and the prediction from the model fitted using the 2014 event was applied to the simulated data. In both cases, comparisons were made with the actual recovery trajectory. The results showed that both models were able to make a reasonable prediction of the trajectory for the data coming from another event. It can also be concluded that increasing the number of data points can improve the extrapolation capability of the model.

The developed spatially-explicit recovery model can be used to estimate the recovery for future earthquakes while incorporating the uncertainty, which is helpful for pre-event planning and enhancing community resilience. For future applications, the accuracy of the model can be further improved by incorporating the data from a set of different events and including more useful explanatory features.

5. Optimizing Post-earthquake Functional Restoration of Distributed Infrastructure Through Deep Reinforcement Learning

5.1 Introduction

Water distribution networks (WN) have been identified as one of several critical infrastructure systems (CIS) (DHS 2013). Following a major seismic event, the physical damage to a water distribution network can cause significant disruption to its core functions (e.g. delivering water to residential commercial and industrial facilities), resulting in adverse socioeconomic impacts to the affected population (Lee et al. 2018; Tabucchi et al. 2008, 2010a). Resilience has been defined as the ability of a social and/or physical entity to withstand, adapt to and recover rapidly from disruptive events (Bruneau et al. 2003). The growing societal expectations towards infrastructure resilience urges for better post-hazard-event restoration policies (Sarkale et al. 2018). Therefore, developing reliable frameworks and tools to support optimal decision-making for the post-earthquake recovery of water distribution systems is one of the critical steps towards enhancing infrastructure resilience.

The post-earthquake recovery of CIS usually involves several tasks: inspection, damage assessment and repair. The repair task usually takes the longest duration and requires significant allocation of resources (Xu et al. 2019). In the literature, the repair scheduling problem for CIS typically takes on one of two formulations. The first utilizes mixed integer linear programming (MILP), whereby the system operation model (equations) is embedded within the optimization constraints. Such formulation is usually seen in applications to electric power transmission and transportation systems. This approach was adopted by Cavdaroglu et al. (2013), who formulated the restoration planning and scheduling of power and telecommunication infrastructure as a mixed-

integer program and proposed a heuristic solution. Xu et al. (2019) reviewed a series of optimization methods based on mixed integer linear programming and proposed two novel heuristic methods for optimizing the repair sequence of electric power transmission systems. Gomez and Baker (2019) employed two-stage stochastic mixed-integer programming to optimize the retrofit and post-event repair of transportation networks.

Another approach is to formulate the repair scheduling as a Markov Decision Process (MDP), which is characterized by a decision maker (agent) that interacts with an environment in finite time steps with the goal of improving its policy actions. The MDP-based formulation is straightforward once the core elements (state, action, transition rule) are identified. When a model/simulator of the environment is available (e.g. a water network environment) to provide the reward/feedback at each time step, the MDP formulation is feasible because the numerous variables/constraints inside the system operation model do not need to be considered explicitly in the optimization. Numerous studies have demonstrated the suitability of MDP for modeling infrastructure decision-making problems and the capability of reinforcement learning/approximate dynamic programming to achieve a solution: Nozhati et al. (2018, 2019, 2020) and Sarkale et al. (2018) formulated the post-earthquake restoration planning of various CISs in the MDP framework and used the rollout algorithm to obtain near-optimal repair policies. A stochastic dynamic programming solution was proposed by Lee and Labadie (2007) to optimize water allocation in a multi-reservoir system. Medury and Madanat (2013) used approximate dynamic programming to optimize the maintenance, rehabilitation and replacement of a pavement system. However, the classical dynamic programming and reinforcement learning methods often suffer from the curse of dimensionality (Nozhati et al. 2019; Sutton and Barto 2018), which limit their effectiveness in large scale problems (Arulkumaran et al. 2017).

Recently, the classical reinforcement learning methods achieved significant breakthrough after being combined with deep learning methods, which is described as deep reinforcement learning (DRL). The DRL methodology has accelerated the progress of reinforcement learning agents in large-scale complex decision-making problems, e.g., the games of Go (Silver et al. 2016, 2017) and StarCraft II (Vinyals et al. 2017), as well as various robotic control problems (Das and Yip 2020; Wu et al. 2019a; Zhu et al. 2017). The deep neural networks addressed the challenge associated with having a large state space through a layered representation of the high dimensional states (e.g., images, video frames). Also, the deep neural networks are used as function approximators for the value or policy functions, which is key to solving the MDP. Because of these advantages, the engineering community has started to explore DRL in applications such as drug design (Popova et al. 2018; Sanchez-Lengeling and Aspuru-Guzik 2018) and electric circuit design (Liao et al. 2019). Current applications of DRL in civil engineering are mostly on the maintenance and operation management of infrastructure systems. Andriotis and Papakonstantinou (2019) proposed a Deep Centralized Multi-agent Actor Critic (DCMAC) algorithm to optimize the maintenance policy of multi-component systems. Memarzadeh and Pozzi (2019) proposed a Safe Q-Learning algorithm for the optimal management of infrastructure systems considering the risk of extreme events. Yao et al. (2020) applied Deep Q Network (DQN) algorithm to optimize the maintenance and rehabilitation of pavement.

To the authors' knowledge, the DRL methodology has not yet been applied to the post-earthquake restoration of water distribution networks. Therefore, this study serves as one of the initial attempts to address this specific challenge. The remainder of the paper is structured as follows: First, we briefly reviews the Markov Decision Process for sequential decision-making problems and the solution via reinforcement learning; Next, we formulate the restoration process

for water networks based on the Markov Decision Process, presents the model implementation details and a general framework for training the model and obtaining the post-earthquake repair policy; A case study on a hypothetical water network is then presented and discussed; Lastly, we summarize the contributions of this paper and provides suggestions for future studies.

5.2 Review of the Markov Decision Process and Reinforcement Learning Methodology

5.2.1 Markov Decision Process

The finite Markov Decision Process (MDP) is a mathematical framework that is used to represent a random process where a decision maker (agent) interacts with an environment at finite time steps $t = 0,1,2 \dots, T$ (Sutton and Barto 2018). The finite MDP is characterized by the following elements: $\mathcal{S}$ is the finite set of states; $\mathcal{A}$ is the finite set of actions; $\mathcal{P}: \mathcal{S} \times \mathcal{S} \times \mathcal{A} \rightarrow [0,1]$ is the state transition matrix with each entry $p(s_{t+1} = s' | s_t = s, a_t = a)$ defining the probability of transitioning from s to s' after taking action a; $\mathcal{R}$ is the reward function, $r(s,a) = E[r_{t+1} | s_t = s, a_t = a]$ represents the expected reward received after taking action a at state s; γ is the discount factor for controlling the decay of importance for future rewards. At each time step t, the agent observes the current state s_t, takes action a_t, receives a reward r_{t+1} from the environment, which then transitions to the next state s_{t+1}. The process continues until a terminal state is reached. The process from an initial to a terminal state is called an episode. On the other hand, some MDP-based applications involve modeling a continuing task without a terminal state, e.g., real-time process control of a water distribution system. The scope of this study is on episodic tasks.

In the MDP, the goal is to learn the policy π, which maps states to a probabilistic distribution over the actions: $\pi(a|s)$ denotes the probability of choosing action a at state s.

Defining the return function $G_t = \sum_{k=0}^{T-t-1} \gamma^k r_{t+k+1}$ as the total discounted reward from time step t to the terminal state T, the goal of the agent is to maximize the expected return

$$\pi^* = \underset{\pi}{\operatorname{argmax}}\, E_{s\sim p,\, a\sim\pi}[G_0] \tag{5.1}$$

where p is the state transition probability. This implies the philosophy of long term planning.

The other core component of the MDP is the value function. Specifically, the state value function $V^\pi(s)$ is defined as the expected return starting from state s and following π onwards (Bellman 2003; Sutton and Barto 2018):

$$V^\pi(s) = E_\pi[G_t|s_t = s] \tag{5.2}$$

Similarly, the state-action value function $Q^\pi(s,a)$ is the expected return after executing action a at state s and following π (Bellman 2003; Sutton and Barto 2018):

$$Q^\pi(s,a) = E_\pi[G_t|s_t = s, a_t = a] \tag{5.3}$$

5.2.2 Reinforcement Learning

The value functions have a recursive property that is reflected in the Bellman equation (Bellman 2003; Sutton and Barto 2018):

$$V^\pi(s) = E_\pi[r_{t+1} + \gamma V^\pi(s_{t+1})|s_t = s] \tag{5.4}$$

which is fundamental to the MDP solution. When all properties (the transition matrix $\mathcal{P}$ and reward function $\mathcal{R}$) of the MDP are explicitly defined, it can be solved by dynamic programming. Specifically, starting from an initial policy, we iterate the policy between two steps (Bellman 2003; Sutton and Barto 2018):

1. Policy evaluation: evaluate the values for all states $s_0, \dots, s_n$ by the Bellman equation. At iteration step $k+1$, update the value $V_{k+1}^\pi(s)$ by

$$V_{k+1}^\pi(s) = E_\pi[r_{t+1} + \gamma V_k^\pi(s_{t+1})|s_t = s] \tag{5.5}$$

2. Policy improvement: improve the policy based on the estimated values

$$\pi'(s) = \underset{a}{\operatorname{argmax}}\, E_{\pi}[r_{t+1} + \gamma V^{\pi}(s_{t+1}) | s_t = s, a_t = a] \tag{5.6}$$

When the properties are unavailable, we need to let the agent learn by interacting with the environment, which is referred to as model-free reinforcement learning. There are two general paradigms for model free reinforcement learning: value-based methods and policy-based methods.

The value-based methods focus on estimating the value functions introduced in Eqs. 5.2 and 5.3. A non-parametric policy is adopted here and updated with respect to the value function, which is similar to Eq. 5.6 but with additional stochastic exploration (ϵ-greedy) (Sutton and Barto 2018). Common methods for estimating value functions include Monte Carlo (MC) and Temporal Difference (TD) methods. The key idea behind the MC method is using the empirical return to approximate the expected return. High variance is associated with the MC estimate because of the sampling from complete episodes (Arulkumaran et al. 2017). Regarding applications in post-hazard-event recovery, Nozhati et al. (2018, 2019, 2020) and Sarkale et al. (2018) performed a series of studies that use MC value estimation and the rollout algorithm (Bertsekas 1999) to solve the MDP for interconnected water, electric power, and food retailer networks within a testbed community located in Gilroy, California. It is worth noting that rollout algorithms are considered as an optimization (decision-time planning) algorithm instead of a learning algorithm, because they do not store the value or policy estimates, and the aim is to find an improved rollout policy (Sutton and Barto 2018). The property of the improved policy depends on factors such as the adopted base policy (Sutton and Barto 2018).

TD methods do not generate samples of episodes, instead, the estimate $\hat{Q}^{\pi}(s_t, a_t)$ is updated towards the "target" Q value

$$\hat{y} = r_{t+1}(s_t, a_t) + \gamma \max_{a'} \hat{Q}^{\pi}(s_{t+1}, a') \tag{5.7}$$

which can be constructed immediately after executing $a_t \sim \pi$, thus $\delta = \hat{y} - \hat{Q}^{\pi}(s_t, a_t)$ is called the TD error. The update is given by

$$\hat{Q}^{\pi}(s_t, a_t) \leftarrow \hat{Q}^{\pi}(s_t, a_t) + \alpha\delta \tag{5.8}$$

where α denotes the step size. Eqs. 5.7 and 5.8 are essentially the value function update steps in the Q-learning algorithm (Watkins 1989), which is the foundation for various value-based deep reinforcement learning methods proposed in recent years. The TD method embeds less variance, but high bias is introduced by bootstrapping, which is likely to cause instability in the learning process (Arulkumaran et al. 2017).

In contrast to the value-based methods like Q-learning, the policy-based methods express the policy explicitly as a parametric distribution $\pi_\theta(a|s)$, and the agent takes actions by sampling from $\pi_\theta(a|s)$ (Sutton et al. 1999). The objective is expressed as a function of θ (Sutton et al. 1999):

$$J(\theta) = E_{\pi_\theta}[G_0] \tag{5.9}$$

Taking the derivative with respect to θ produces the policy gradient (Sutton et al. 1999):

$$\nabla_\theta J(\theta) = E_{\pi_\theta}[\nabla_\theta \log \pi_\theta(a|s) Q^{\pi_\theta}(s, a)] \tag{5.10}$$

and θ is updated through (Sutton et al. 1999):

$$\theta \leftarrow \theta + \alpha \nabla_\theta \log \pi_\theta(a|s) Q^{\pi_\theta}(s, a) \tag{5.11}$$

Policy-based methods have better convergence properties as the policy is parameterized and the algorithm approaches a local maximum. However, high variance is imposed on the gradient (Arulkumaran et al. 2017), as the value needs to be estimated by the MC methods discussed earlier.

5.2.3 Deep Reinforcement Learning

The Q-learning algorithm and many other classical reinforcement learning methods require the value function to be stored for all possible state-action pairs in a tabular form, which is iteratively updated. Hence, the storage requirement grows exponentially with the expansion of the state and action spaces and becomes intractable for large-scale problems. Mnih et al. (2015) introduced the Deep Q Network (DQN) algorithm, which is a benchmark for embedding deep neural networks as function approximators in the reinforcement learning agents. Fig. 5.1 shows the schematic structure for the DQN model, where the solid lines represent the interaction process, and the dashed lines represent the learning process.

In the interaction process, at each time step, the agent observes the current state s, which is input to the Q network to approximate the value $Q_\theta(s, a)$ for possible actions, then takes action a based on the Q value, receives a reward r from the environment, and transitions to the next state s'. The deep neural network enables layered representation of the high dimensional states such as the raw pixels of video games, and outputs the values $Q_\theta(s, a)$ for all available actions a at state s via a forward pass. Moreover, instead of storing the Q values for large state-action pairs, only the parameters of the neural networks need to be saved and updated. Each transition tuple (s, a, r, s') is stored to a replay memory buffer with storage capacity N, i.e., new transitions are taken in and old transitions are expelled.

In the learning process, a target Q network $Q_{\theta^-}(s', a')$ is used to construct the "target" Q value for learning (Arulkumaran et al. 2017; Mnih et al. 2015):

$$\hat{y} = r(s, a) + \gamma \operatorname*{argmax}_{a'} Q_{\theta^-}(s', a') \tag{5.12}$$

The parameters θ^- of the target network are inherited from the Q network at delayed steps. Through temporal fixation, the target network provides a more stable target that addresses the main

issue with classical Q learning. At each learning step, the agent randomly samples a mini-batch, $B = \{(s_i, a_i, r_i, s_i')\}$, from the replay memory buffer, and updates θ through stochastic gradient descent on the loss function (Mnih et al. 2015)

$$L(\theta) = \frac{1}{K}\sum_{i=1}^{K}(r_i + \gamma \underset{a'}{\operatorname{argmax}}\, Q_{\theta^-}(s_i', a_i') - Q_\theta(s_i, a_i))^2 \tag{5.13}$$

where K denotes the batch size. This modification reduces the correlation of samples used for computing the TD error, thus reducing the bias in the update.

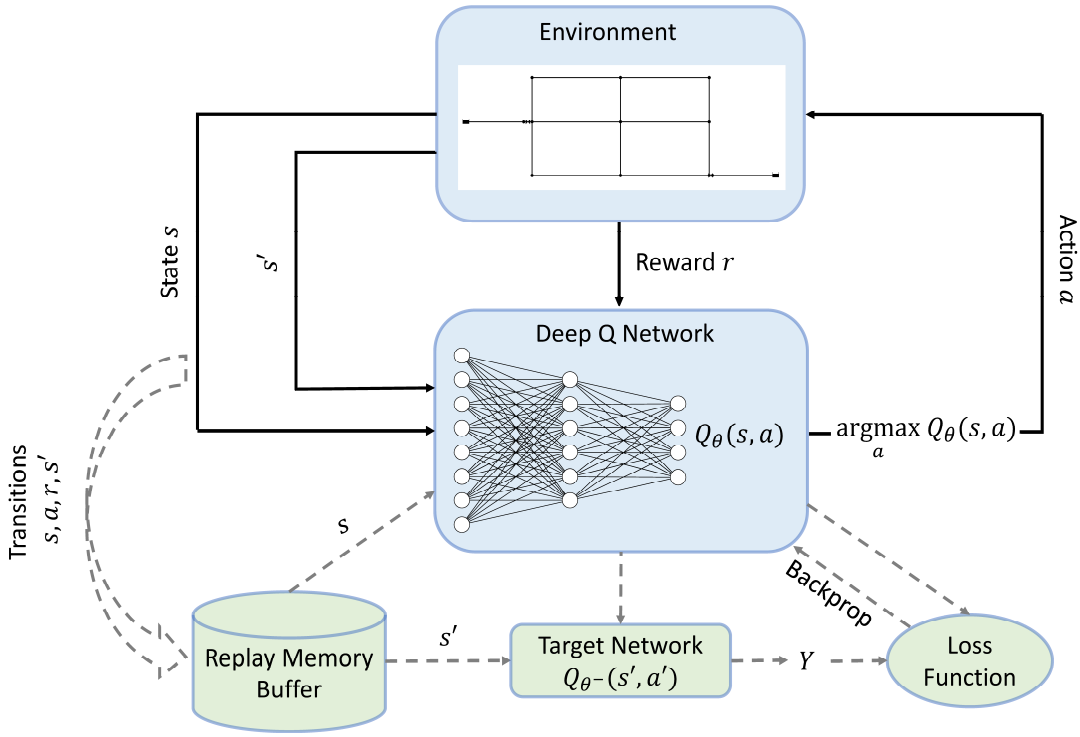


Figure 5.1 Schematic representation of the Deep Q Network

The Deep Actor Critic (DAC) combines the value-based methods with policy-based methods. More specifically, it considers the trade-off between the bias introduced in the value-based methods and the variance in policy-based methods (Grondman et al. 2012; Konda and Tsitsiklis 2003). The model consists of the two main components shown in Fig. 5.2: an actor

network that approximates the policy function $\pi_{\theta_a}(a|s)$ and takes action by sampling from the policy $a_t \sim \pi_{\theta_a}(a|s)$; a critic network that approximates the value function $V_{\theta_c}(s)$ and evaluates the advantage of the action taken by the actor.

The actor is updated through a policy gradient with respect to θ_a

$$\theta_a \leftarrow \theta_a + \alpha_a \nabla_{\theta_a} \log \pi_{\theta_a}(a|s) A^{\pi_{\theta_a}}(s,a) \tag{5.14}$$

where $A^{\pi_{\theta_a}}(s,a) = Q^{\pi_{\theta_a}}(s,a) - V^{\pi_{\theta_a}}(s)$ is called the advantage function, which measures the advantage of taking action a with respect to the average value of state s. Subtracting the baseline $V^{\pi_{\theta_a}}(s)$ reduces the variance in the policy gradient methods (Grondman et al. 2012; Konda and Tsitsiklis 2003; Sutton et al. 1999). Note that the advantage function can be estimated by $\hat{A}^{\pi_{\theta_a}}(s,a) = r(s,a) + \gamma V_{\theta_c}^{\pi_{\theta_a}}(s') - V_{\theta_c}^{\pi_{\theta_a}}(s)$, i.e., the values of the current state s and next state s' are both outputs by the critic, as shown in Fig. 5.2.

The critic is updated through the gradient of the squared TD error with respect to θ_c (similar to the Q network):

$$\theta_c \leftarrow \theta_c + \alpha_c \delta \nabla_{\theta_c} V_{\theta_c}(s) \tag{5.15}$$

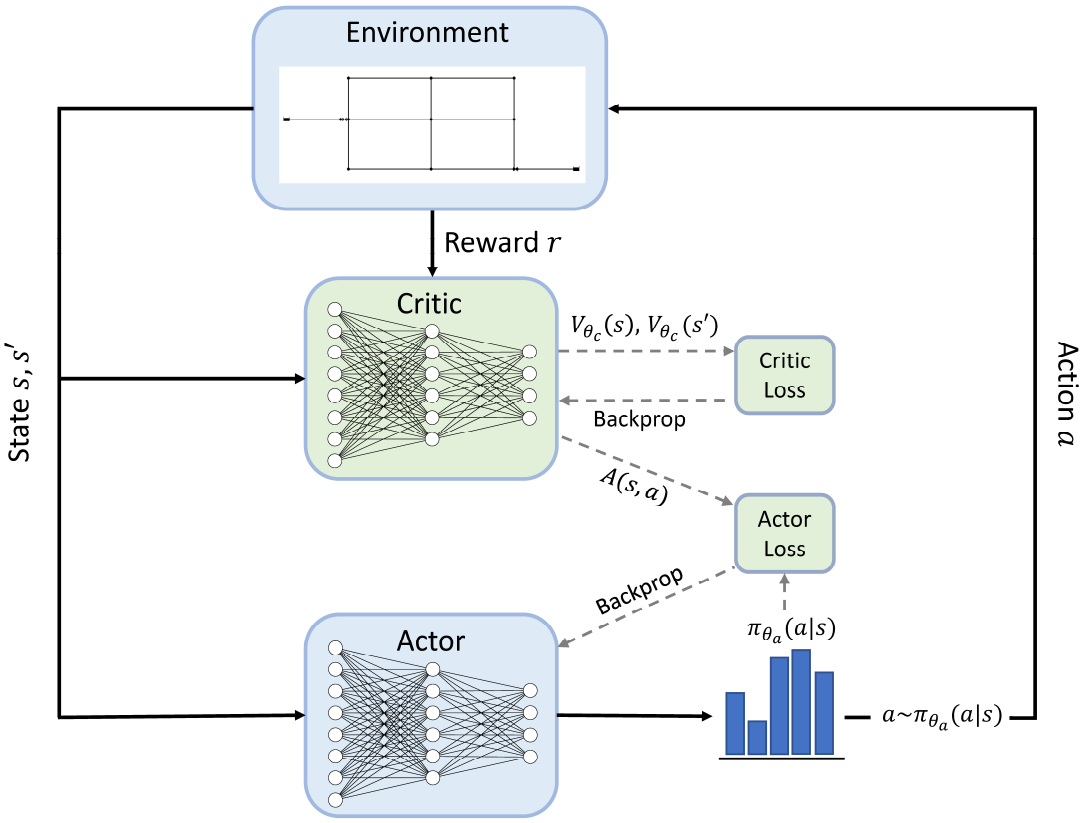


Figure 5.2 Schematic representation of the Deep Actor Critic

5.3 Implementation of the Deep Reinforcement Learning Methodology

5.3.1 Formulation of Post-earthquake Recovery Optimization Problem

The components of a water distribution network can remain intact or undergo damages after the occurrence of earthquakes. Different decisions on the sequence of the available set of repair actions from the time immediately after the earthquake ($\tilde{t}_0$) to when the system recovers to full functionality ($\tilde{t}_{end}$) results in different restoration curves. Fig. 5.3 shows a schematic representation of the restoration curves, where the horizontal axis is the time following the seismic event, and the vertical axis is the System Serviceability Index (SSI). For the water distribution system, the SSI is defined as the ratio between the actual water supply to a junction to the desired demand (Lee et al. 2018):

$$SSI(\tilde{t}) = \frac{\sum_{i=1}^{N}\sum_{j=1}^{L} q_{i,j}(\tilde{t})}{\sum_{i}^{N} d_i\,(\tilde{t})} \tag{5.16}$$

where $q_{i,j}(\tilde{t})$ denotes the actual flow of water from pipe j into junction i, $d_i(\tilde{t})$ is the expected demand (desired flow) of junction i.

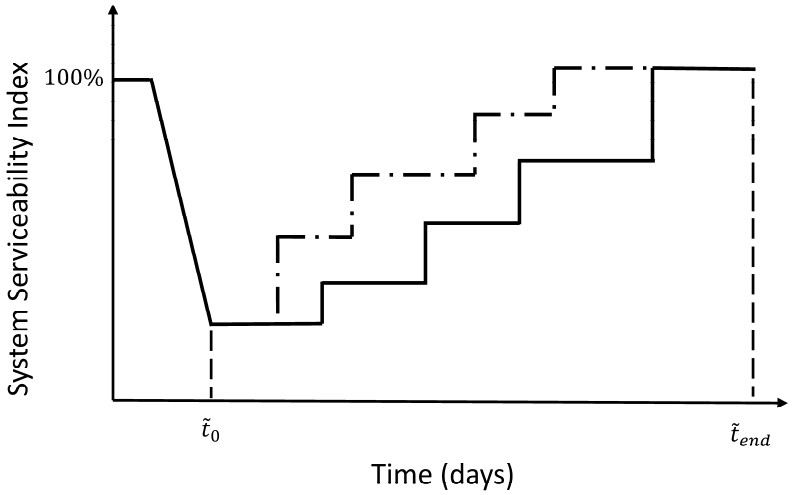


Figure 5.3 Schematic representation of a restoration curve for the system serviceability index

In this study, we consider the primary optimization objective as minimizing the cumulative loss of serviceability normalized by the total recovery time T (denoted as Objective 1)

$$\underset{\pi}{\text{minimize}}\; E_{\pi}\left[\frac{\int_{\tilde{t}_0}^{\tilde{t}_{end}}(1 - SSI(\tilde{t}))d\tilde{t}}{T}\right] \tag{5.17}$$

subject to implicit constraints from the hydraulic simulation model. Eq. 5.17 can be equivalently expressed as maximizing the normalized area under the restoration curve (Cimellaro et al. 2010):

$$\underset{\pi}{\text{maximize}}\; E_{\pi}\left[\frac{\int_{\tilde{t}_0}^{\tilde{t}_{end}} SSI(\tilde{t})d\tilde{t}}{T}\right] \tag{5.18}$$

If $SSI(\tilde{t})$ is a tractable function (with explicit expression) of the damaged state of the network, the solution can be obtained by dynamic programming. However, SSI is obtained from

real-time hydraulic analysis on the physical model of the water network. Hence, feasible solution need to be obtained by learning from interactions.

In this study, two damage states are considered: DS_0 (Undamaged) and DS_1 (Major Leak) (Lee et al. 2018). The MDP framework described earlier is adapted to model the restoration process of the water network, as follows:

- State: $s_t = [s_{t1}, s_{t2}, \dots, s_{tL}]$

  where $s_{tj} \in \{0, 1\}$, 0 corresponds to DS_0 and 1 corresponds to DS_1

- Action: $a_t = [a_{t1}, a_{t2}, \dots, a_{tK}]$,

  where K denotes the number of available repair crews, $a_{tj} \in \{1,2, \dots, L\}$ determines the corresponding pipe to be repaired.

- Reward: $r_{t+1}(s_t, a_t) = SSI(s_t, a_t) \cdot \tilde{t}_{rep}(a_t)$, where $SSI(s_t, a_t)$ is the SSI after executing action a_t at state s_t (obtained from hydraulic simulation), and $\tilde{t}_{rep}(a_t)$ denotes the duration of the repair action a_t. In this formulation, only two damage states are considered, and the time needed to repair a pipeline from DS_1 to DS_0 is assumed to be the same for all teams. In future studies, this formulation can be modified to incorporate stochastic repair duration for each team and more than two damage states.

- Discount factor: $\gamma = 0.99$ is used for the agents.

5.3.2 Solution Framework

A general framework for implementing the proposed DRL methodology, which incorporates the seismic hazard model for obtaining the shaking intensity and initial damage. The main steps are as follows:

Step 1: Damage simulation: simulate a representative number of possible damage cases to the

WN using a suite of earthquake scenarios.

Step 2: Model training: starting from each damage case as the initial state of the environment, let the DRL agent learn by interacting with the environment for multiple episodes. Within each episode,

1. Episode start: the water distribution network is reset to the state s_0, which represents the initial damage case;

2. At each time step of the episode:

- The agent observes the water system state s_t and decides the next repair action a_t;
- The environment receives the repair action a_t, repairs the corresponding pipe, transitions to the next state s_{t+1} and obtains the $SSI(s_t, a_t)$ through hydraulic simulation, which is then transformed and passed to the agent as the reward r_{t+1};
- The agent learns and updates its policy.

3. Episode terminates: all pipes are repaired, and the serviceability of the water distribution network is fully restored (100%).

Step 3: Obtain repair policy: after training is complete, given a damage case of interest, the model outputs the optimal repair sequence and the associated restoration curve.

For the damage simulation step, the repair rate (RR) is computed as a function of the peak ground velocity (PGV) using the empirical equation recommended by the American Lifelines Alliance (2001), which takes the form of a linear model

$$RR = K_1 \cdot 0.002416 \cdot PGV \tag{5.19}$$

or a power model

$$RR = K_1 \cdot 0.001187 \cdot PGV^{1.173} \tag{5.20}$$

where the units of RR and PGV are the number of repairs/km and cm/s, respectively. K_1 is a modification factor based on the type of material, connection, soil and pipe diameter. In practice, the fragility function of the pipelines is determined from the repair rate. Specifically, the number of repairs X for a pipeline of length L is assumed to follow a Poisson distribution given by (Eidinger 2001):

$$P(X = k) = \frac{(RR \cdot L)^k}{k!} e^{-RR \cdot L} \tag{5.21}$$

Hence, the probability of failure (without specifying the extent) for a single pipe with length L is expressed as follows (Choi et al. 2018):

$$P_f = 1 - P(X = 0) = 1 - e^{-RR \cdot L} \tag{5.22}$$

For a given PGV value, Eq. 5.19-22 is used to perform multiple damage simulations to the water network.

5.3.3 Description of the Environment and the Deep Reinforcement Learning Model

In this study, the water distribution system model is constructed using the Python-based Water Network Tool for Resilience (WNTR) (Klise et al. 2018). The damage state DS_1 (Major Leak) is realized by creating a leak area with a diameter that is 90% of the pipe diameter (Lee et al. 2018) and the leak demand d_{leak} is computed as follows (Crowl et al. 2002; Lee et al. 2018):

$$d_{leak} = C_d A p^{\alpha} \sqrt{\frac{2}{\rho}} \tag{5.23}$$

where C_d is the discharge coefficient taken as 0.75, A is the leak area, p is the pressure in the pipeline, α is the discharge coefficient, ρ is the fluid density. It is worth noting that the water network model is used to relate the system state to its damaged condition, and provides an outcome that is a function of the real-time hydraulic simulation analysis.

To implement the DRL methodology, a simulation-based environment needs to be constructed which embeds the water distribution network model. The water network environment, named "WaterNet-v0", is implemented as a customized environment (subclass) of the OpenAI gym environments (Brockman et al. 2016), which is a platform for developing and testing reinforcement learning models. The corresponding class methods are written in the standard format of gym environments (e.g, a "Step" method to perform state transition after receiving an action). The DRL models (Deep Q Network and Deep Actor Critic) are implemented in Python using the TensorFlow framework (Abadi et al. 2016). The model structures are implemented as follows:

Q network:

- Input layer: 14 units (corresponding to the state dimension, the state is a vector recording whether each pipe is damaged, as defined earlier)
- Hidden layer 1: fully-connected layer with 128 units and rectified linear unit activation
- Hidden layer 2: fully-connected layer with 128 units and rectified linear unit activation
- Output layer: fully-connected layer with 14 units and linear activation

Actor network:

- Input layer: 14 units
- Hidden layer 1: fully-connected layer with 128 units and leaky rectified linear unit activation
- Hidden layer 2: fully-connected layer with 128 units and leaky rectified linear unit activation

- Output layer: fully-connected layer with 14 units (corresponding to the number of pipes) and softmax activation

Critic network:

- Input layer: 14 units
- Hidden layer 1: fully-connected layer with 128 units and rectified linear unit activation
- Hidden layer 2: fully-connected layer with 128 units and rectified linear unit activation
- Output layer: fully-connected layer with a single unit and linear activation

The Adam optimizer (Kingma and Ba 2014) is used to perform the gradient descent on the neural network parameters. For DQN, a learning rate of 5e-4 is used, and for DAC, 8e-4 is used for the actor and 8e-3 is is used for the critic.

It is worth noting that the action space $\mathcal{A}$ is not constant because after repairing each pipeline, the corresponding action becomes ineffective at the next time step. To address this issue, two approaches are explored when implementing the models. For the DQN agent, no constraints are imposed on the action space at each time step. Instead, a negative reward is introduced to punish the selected illegal action. For the Actor Critic model, when sampling an action, a filter is added to the output distribution from the actor by removing the illegal actions and renormalizing the distribution by

$$\pi'(a|s) = \frac{\pi(a|s)}{\sum_{a' \in \mathcal{A}_L} \pi(a'|s)} \tag{5.24}$$

where $\pi'(a|s)$ is the adjusted legal policy, $\mathcal{A}_L$ is the legal action space at state s. Although this adjustment would potentially impose ill-conditions for the exact convergence of the policy, it

updates the policy towards the near optimal direction.

5.4 Case Study on a Hypothetical Water Network

In this section, a case study is performed where the DRL methodology is applied to the hypothetical water distribution network shown in Fig. 5.4, which has been used in several prior studies (Jayaram and Srinivasan 2008; Larock et al. 2010; Lee et al. 2018). Additional details on the configuration and composition of the network can be found in the aforementioned references.

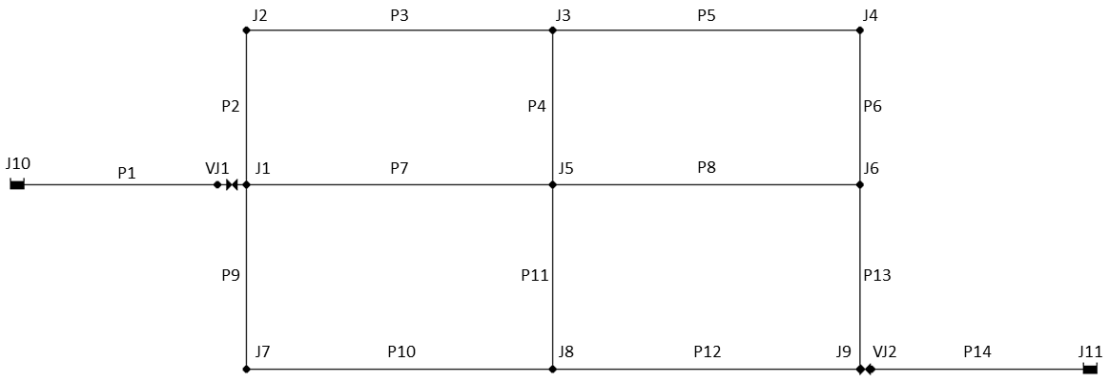


Figure 5.4 Hypothetical water distribution network (adapted from Jayaram and Srinivasan (2008), Larock et al. (2010), and Lee et al. (2018))

5.4.1 Evaluation Results

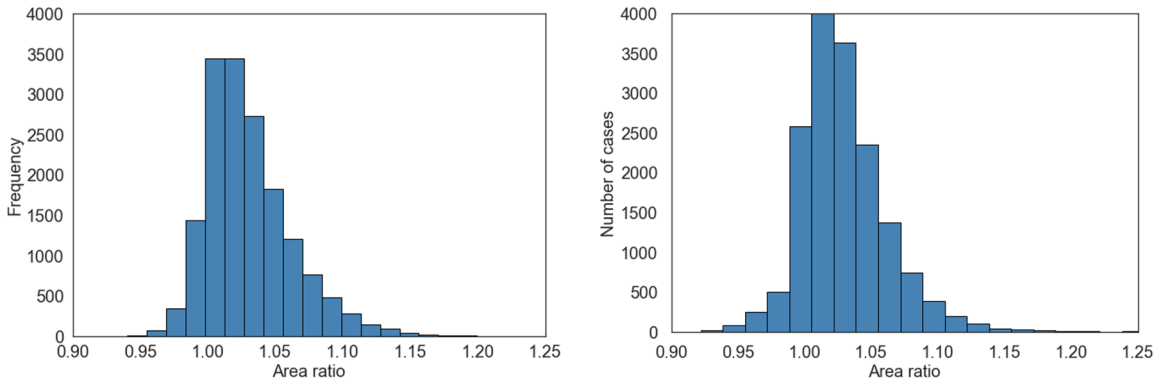
The hypothetical water distribution network consists of 14 distribution pipelines. Therefore, the number of possible damage cases is $2^{14} - 1 = 16383$, which is a tractable number. For this evaluation, a single repair team is assumed. Hence, the total recovery time for a given case is deterministic, the objective in Eq. 5.18 is equivalent to maximizing the expected area under the restoration curve. As mentioned in Section 5.3.3, a random sample of damage cases is first generated, and the agents are trained using multiple episodes for each damage case. In this study, 1200 damage cases are used for the DQN agent and 1000 cases for the DAC agent. The number of episodes per case is set at 50. Once the training is completed, the performance of the agent is

evaluated and compared with a "random" agent, i.e., an agent with no knowledge that selects a random pipeline to repair at each step. The evaluation results are presented in two ways:

1. Test on all possible (16383) cases. This evaluation does not require an earthquake ground motion simulation model. It simply compares the relative performances of the DRL agents and random agent starting from any initial state of the water distribution system. When the DRL method is applied to larger systems, the number of possible damage cases can be intractable. In cases like these, a reasonable number of damage cases can be generated from scenario-based simulation.

2. Test on the damage cases simulated by taking PGV values from 91.92 to 900 $\mathrm{cm/s}$. 1000 damage cases are simulated for each PGV value. The range of PGV is adopted from the site hazard curve at Los Angeles City Hall (Lee et al. 2018). Because the focus of this study is not on life-cycle assessment, an integration over the entire range of the hazard curve is not performed. The goal here is to perform the evaluation by considering an extensive range of PGV values under which the hypothetical network starts to exhibit significant component damage (from a preliminary simulation).

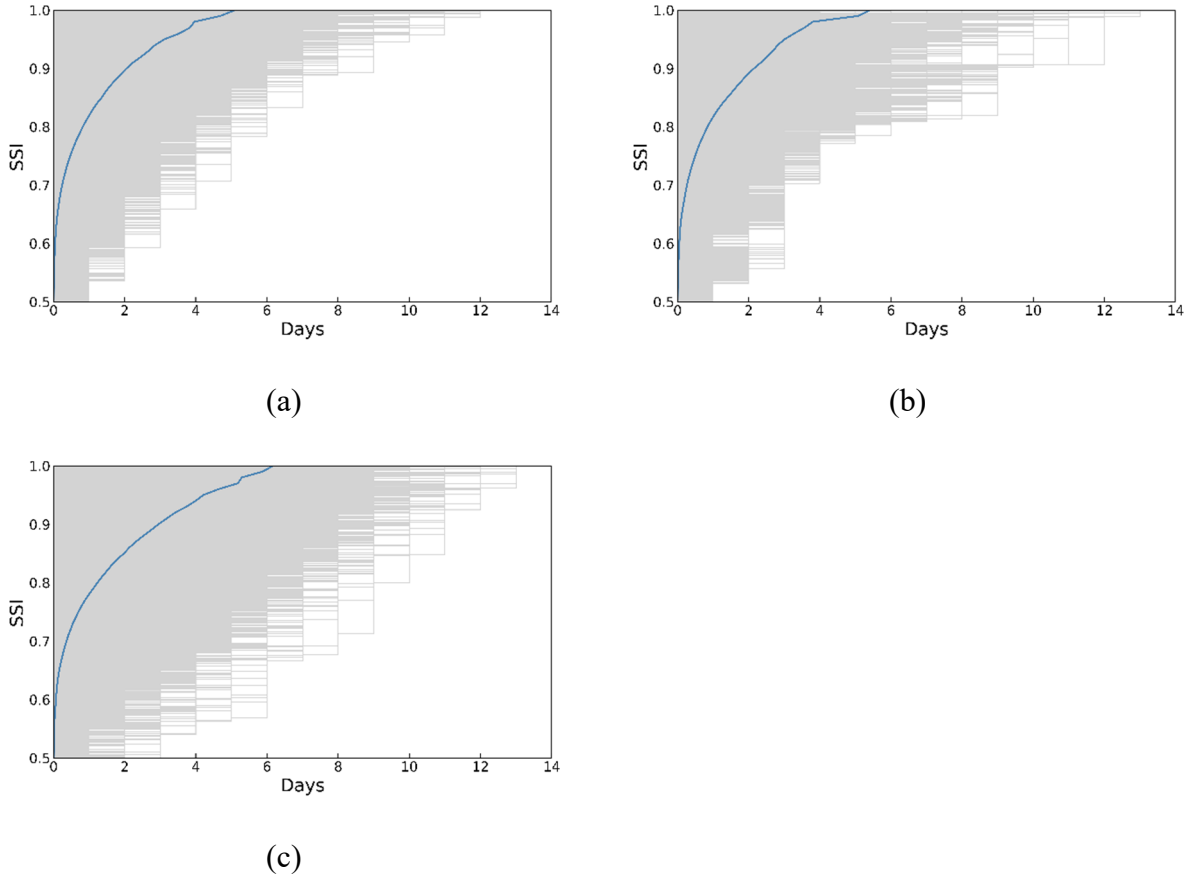
Among the 16383 evaluated cases on metric 1 (area under the restoration curve (A^{res})), the DQN agent outperforms the random agent in 13834 cases (winning rate 84.4%), while the DAC outperforms random in 14029 cases (winning rate 85.6%). Fig. 5.5 shows the distribution of the ratio of the area under the restoration curve following the DRL policy to that following the random policy.



(a) (b)

Figure 5.5 Ratio of the area under the restoration curve: (a) Deep Q Network to random; (b) Deep Actor Critic to random

Fig. 5.6 presents the evaluation results for the DRL and random agents considering all possible cases, where the light grey lines are the restoration curves for individual cases, and the blue line presents an average restoration curve computed by averaging the required time to reach a given SSI level (at an increment of 0.01). It can be shown that the restoration curves following the repair policy by the DQN agent have much smaller dispersion and the area under the restoration curves are all greater than half of the area associated with full functionality, i.e., exhibits a "concave" shape. The restoration curves from the DAC agent exhibit slightly larger dispersion than those of the DQN, which is expected since the Actor outputs a distribution over actions and samples from the distribution, while DQN takes greedy actions with respect to the Q value. Nevertheless, the DAC has much better performance than the random agent.



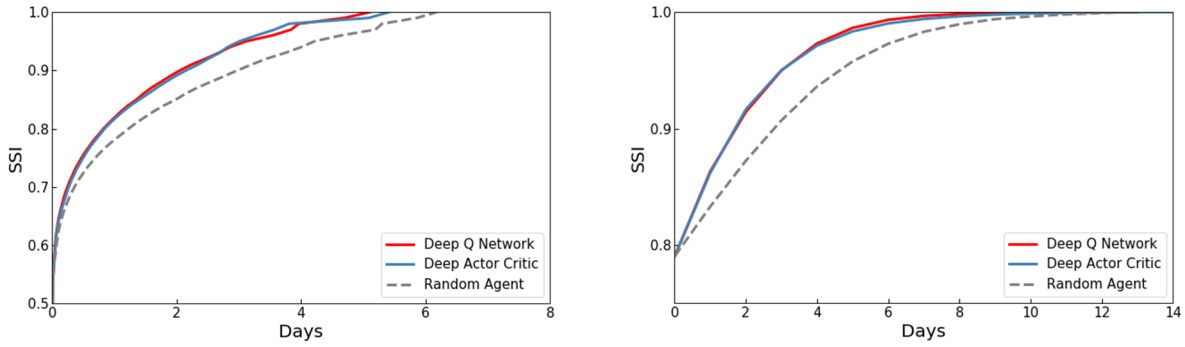
(a) (b)

(c)

Figure 5.6 SSI-based restoration curves for all possible damage cases: (a) Deep Q Network; (b) Deep Actor Critic; and (c) Random Agent

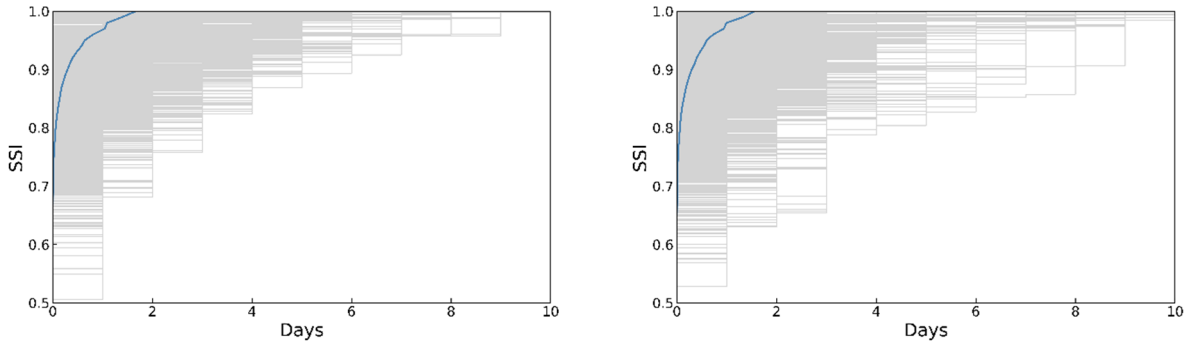
A comparison of the average performances of the three agents is shown in Fig. 5.7. Fig. 5.7 (a) presents the results given by averaging the required time to reach a given SSI level. On average, the performance of the DAC is similar to the DQN. The average time required to recover to full serviceability following the policy from the DRL agents are approximately 5 days, while 6 days are needed on average when following the random agent policy. Despite the small difference in terms of absolute values (because of the small network), the average recovery time for the random agent is 20% higher than the DQN. Fig. 5.7 (b) presents the restoration curves computed by averaging the SSI at specific time-points after damage has occurred. The average SSIs achieved following the DQN and DAC are much higher than the random agent case.

The test results on the hazard-generated damage cases are presented in Figs. 5.8 and 5.9, from which similar conclusions to the ones discussed earlier are drawn.

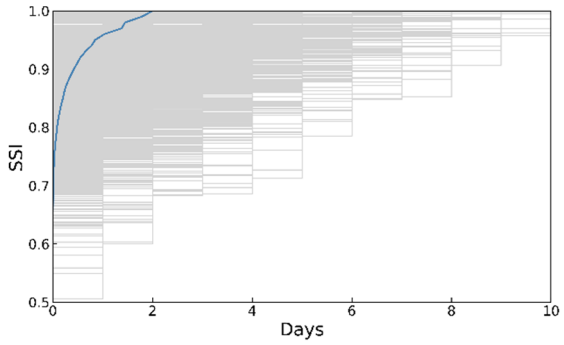


(a) (b)

Figure 5.7 Results from all possible cases: (a) average days required to achieve a given SSI level; (b) average SSI achieved at a given day

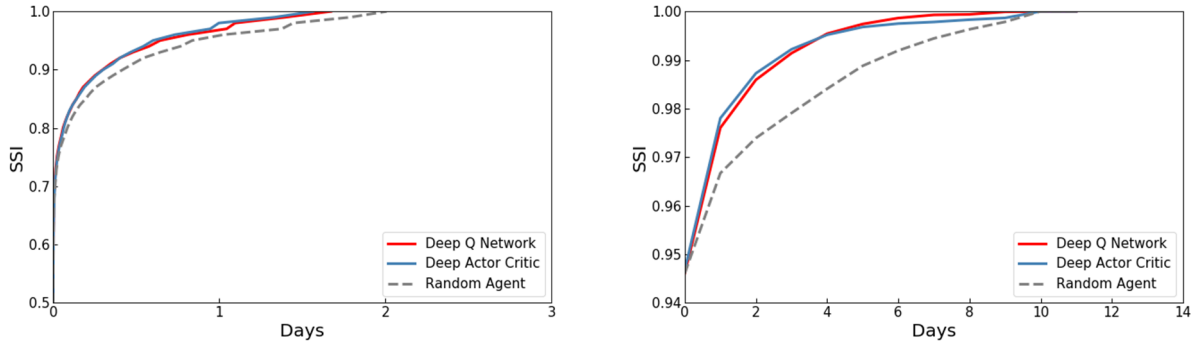


(a) (b)



(c)

Figure 5.8 Restoration curves for damage cases generated from the hazard model: (a) Deep Q Network; (b) Deep Actor Critic; and (c) Random Agent



(a) (b)

Figure 5.9 Restoration curves for damage cases generated from different PGVs: (a) Average days required to achieve a given SSI level; (b) Average SSI achieved at a given time-point

The proposed method using the Deep Q Network can be applied to small to moderate-sized networks and when the number of repair teams is relatively small, since the Q network needs to evaluate the value of all possible actions at a given state. For a WN with L pipelines and K repair teams, the output dimension of the Q network is $\binom{L}{K}$, thus the computation could become intractable when L and K become large. In general, the actor critic model structure would be a more suitable choice when scaling to a large discrete action space. While the exact model of the actor critic network implemented in this study cannot be directly applied to tasks of large discrete action space, some modifications that include employing a multi-actor single-critic structure (Lowe et al. 2017), using a single actor critic with a deterministic policy gradient and mapping the continuous action space to a discrete one (Silver et al. 2014; Dulac-Arnold et al. 2015), can potentially address the challenge.

5.4.2 Influence of the Training Scheme on the Performance of the Deep Q Network Model

To gain a better understanding of the influence of the training scheme of the performance of the DQN agent (with respect to the random agent), a series of experiments is performed using number of randomly sampled damage cases from 200 to 1600, and with 50 and 100 episodes per case, as shown in Fig. 5.10.

When trained for 50 episodes per case (blue line), the performance improves from 200 to 1200 cases, then starts to decrease. This can be due to the stochastic exploration mechanism now adds noise to the near-optimal behavior of the agent at 1200 cases, thus leading to over-exploration. On the other hand, increasing the episodes per case to 100, the agent tends to over-exploit the experience (overfitted the policy on a sub-environment) from each case. Although the performance becomes better after 1200 cases, but still worse than the performance with 1200 cases and 50 episodes. Therefore, using 1200 cases and 50 episodes per case achieves a reasonable trade-off between exploration and exploitation.

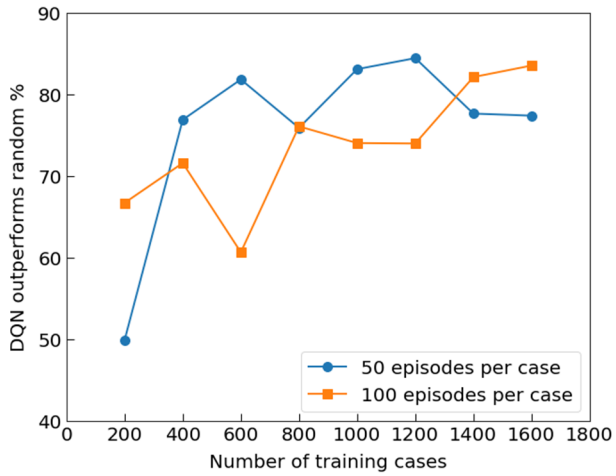


Figure 5.10 Sensitivity analysis on the number of training cases and episodes

5.4.3 Additional Experiments Using the Time to Reach 90% SSI as the Resilience Metric

In the previous analyses, the objective was to maximize the normalized area under the

restoration curve and the performance of the DRL agents was evaluated on all possible damage cases. In practice, it is of interest to consider other objectives, for example, minimizing the time to restore the system to a pre-defined percentage of functionality, or the time for certain percentage of housing units to have access to water. In this subsection, we consider the objective (denoted as Objective 2) of minimizing the time to reach 90% SSI (denoted as Metric 2: $\tilde{T}^{90}$). The optimization problem can be formulated as follows:

$$\underset{\pi}{Minimize} \quad E_{\pi}[\sum_{t=0}^{T} \tilde{t}_{rep}(a_t)] \qquad (5.25)$$

$$\text{Subject to} \qquad SSI(s_T,\ a_T) \geq 90\%$$

where T is the terminal time step and $\tilde{t}_{rep}(a_t)$ is the duration for repair action a_t. Hence, we can define an MDP with the following objective function: $\underset{\pi}{\text{Maximize}} \quad E_{\pi}[-\sum_{t=1}^{T} \tilde{t}_{rep}(a_t)]$.

The reward function is then designed as $r_{t+1}(s_t, a_t) = -\tilde{t}_{rep}(a_t)$, and the episode is terminated once reaching 90% SSI. Other elements are the same as the MDP defined in Section 5.3.1.

Based on Metric 2, the DQN agent is trained for 1600 cases with 50 episodes per case. For evaluation, 13912 cases are used because there are 2471 cases in which the initial SSIs are higher than 90%, and the performance of the DQN is compared to the random agent (in this case, the focus is having the time for the DQN to reach 90% SSI being less than that for the random agent). Fig. 5.11 shows the distribution of the ratio of the time to reach 90% SSI following the DRL policy to that following the random policy. The DQN agent outperforms random in (67.6%), while the performance is the same for both agents in 28.8% cases, which suggests that for this small system, the time to reach 90% SSI might not vary significantly in many cases (the sequences can vary significantly, but the total time tends to be stable).

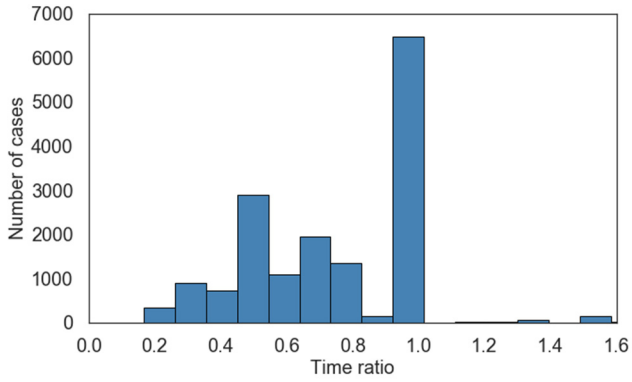


Figure 5.11 Distribution of the ratio of the time to reach 90% SSI following the DRL policy to that following the random policy.

Next, a series of experiments of training the DQN on Objective 2 with 1200 to 2000 training cases is performed with 50 episodes per case (because from the previous sensitivity analysis 1200 was the best), and evaluated on both Metric 1 and Metric 2. As shown in Fig. 5.12, the performance of the DQN agent improves significantly from 1200 to 1400 cases, but does not improve much afterwards.

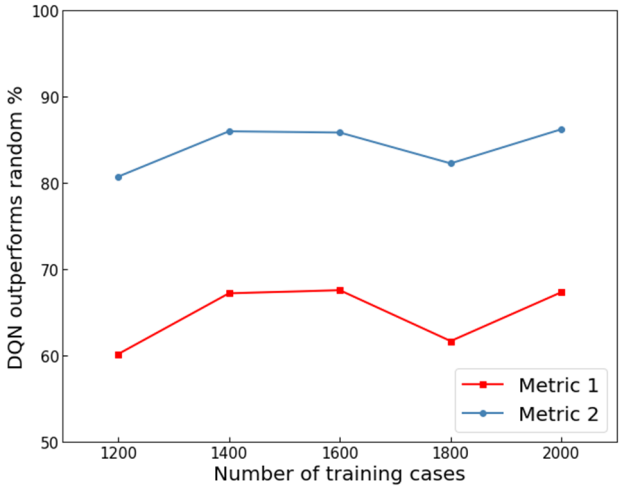


Figure 5.12 Sensitivity analysis on the number of training cases and episodes

5.5 Conclusion

This study explored the application of the deep reinforcement learning methodology to

address the challenge of post-earthquake repair scheduling for water distribution systems. The Markov Decision Process was adapted to define the restoration process. A system model was constructed using the Water Network Tool for Resilience (WNTR) package and used to perform real-time hydraulic analysis. The network model was then embedded in a customized reinforcement learning environment written using the OpenAI gym platform. Two types of deep reinforcement learning algorithms, namely Deep Q Network (DQN) and Deep Actor Critic (DAC) were adopted to learn the near optimal repair policy by interacting with the water network environment. A general framework for applying the deep reinforcement learning methodology was proposed which can be extended to large water networks while incorporating scenario-based earthquake simulation.

The proposed methodology was applied to a hypothetical network consisting of 14 pipelines. During the model training phase, 1200 and 1000 damage cases were randomly sampled from the entire space (16383 damage cases) for the DQN and DAC agents respectively, and 50 episodes are considered in each case. The trained model was then evaluated on all possible damage cases for the water network and compared with an agent that has no knowledge and employs a random assignment policy of repair actions. Evaluation results demonstrated that the deep reinforcement learning agents successfully learned better repair policies than the random assignment policy, both in terms of application across a wide range of damage scenarios and the average performance.

Next, a sensitivity analysis was conducted to explore the influence of the number of cases and episodes per case used for training on the performance of the DQN agent. It was determined that using 1200 training cases and 50 episodes per case results in the best performance. A similar procedure can be used to determine a reasonable training scheme when applying the DRL method

to other systems. To gain better understanding of the effectiveness of different resilience metrics used to design the model, a series of additional experiments were performed by training the DQN agent with the objective of minimizing the time needed to restore to 90% of the system serviceability index (SSI) for the water distribution system. The evaluation results showed that the best DQN agent outperforms the random agent in 67% cases, while the restoration time for the two agents were the same in 29% cases, which suggested that the variation in the time to reach 90% SSI for this water system is small.

As one of the initial applications of the deep reinforcement learning methodology to inform decisions associated with water network systems, this study showed its capability to enhance infrastructure resilience. Nevertheless, there remain several limitations that need to be addressed in future studies. The proposed framework was evaluated on a simple hypothetical water network for demonstration purposes and only pipeline damage and a single repair team are considered. Evaluations on larger networks should be performed with realistic consideration of the possible damage states, number of repair crews, repair time. In addition, theoretical modifications of the implemented Actor Critic model need to be carefully studied to further enhance its scalability to large water networks and other infrastructure systems. Finally, aftershocks could happen during the restoration process which can cause new damage to the components of the infrastructure. Therefore, accounting for a representative mainshock-aftershock pattern would enhance the robustness of the proposed methodology.

6. A Deep Actor Critic Based Framework to Support the Repair Optimization for Distributed Infrastructure

6.1 Introduction

Earthquakes and other natural hazards can cause damages and undesirable functionality loss to critical infrastructure systems (CIS) (Nozhati et al. 2019). To meet the growing societal expectations towards infrastructure resilience, it is important to developing efficient decision making for post-earthquake recovery of water distribution systems and other CISs (Bruneau et al. 2003; Sarkale et al. 2018).

In the literature of developing decision-making frameworks for post-earthquake recovery of CIS, the emphasis is often placed on optimizing the scheduling of repair actions to minimize the resilience loss over the restoration time horizon, which is a sequential decision making problem. Therefore, the restoration process can be modeled as a Markov Decision Process (MDP), which is characterized by a decision maker (agent) that interacts with an environment with the goal of improving its strategy for taking actions. The MDP-based formulation is straightforward once the core elements (state, action, transition rule) are identified. When a model/simulator of the environment is available (e.g., a water network model) to provide the reward/feedback at each time step, the MDP formulation is feasible because the numerous variables/constraints inside the system operation model do not need to be considered explicitly in the optimization and can be generalized to different systems. The suitability of this approach has been demonstrated in various studies for optimizing the restoration of CIS (Lee and Labadie 2007; Nozhati et al. 2018, 2019, 2020; Sarkale et al. 2018).

The deep reinforcement learning (DRL) method provides a strong tool for solving complex large-scale MDP problems (Silver et al. 2016, 2017; Vinyals et al. 2017). Chapter 5 explored the

applicability of the DRL method in optimizing the repair actions for the post-earthquake recovery on a hypothetical small water distribution network. It needs to be noted that the typical setting of DRL applications usually involves a large state space (e.g., video frames, images), but a small to moderate discrete action space (e.g., action set in the Atari games), or a continuous action space (e.g., movements of robots). In the application of post-earthquake restoration, a large discrete action space is often encountered, which is less studied in typical DRL applications. For example, for a system consisting of 100 pipelines and 5 repair teams are available, the action space is on the order of 10^{10}. Starting from the actor critic structure, some modifications can be made to address such challenge, such as the Wolpertinger model, in which the continuous actions produced by the actor are mapped to discrete legal actions (Dulac-Arnold et al. 2015; Silver et al. 2014).

In this study, we develop a deep reinforcement learning based framework to improve repair scheduling for the post-earthquake restoration of water distribution systems. Section 6.2 provides the general formulation of the post-earthquake recovery optimization problem. The technical background of the MDP and DRL is briefly reviewed in Section 6.3; A MDP based formulation of the restoration process and a general framework for training the model and obtaining the post-earthquake repair policy are presented in Section 6.4; To demonstrate the capability of the DRL framework for problems with large state and action space, the Wolpertinger algorithm is adapted in Section 6.5 and evaluated on a realistic system in Novato, CA, based on damage cases generated from scenario-based damage simulation. Section 6.6 provides a summary of the work and limitations.

6.2 Formulation of Post-earthquake Recovery Optimization Problem

The components of an infrastructure system can remain intact or undergo damage after the occurrence of an earthquake. Different decisions on the sequence of the available set of repair

actions from the time immediately after the earthquake ($\tilde{t}_0$) to when the system recovers to full functionality ($\tilde{t}_e$) results in different restoration curves. $F(\tilde{t})$ is the functionality at time $\tilde{t}$. In this study, we consider the optimization objective as the area under the restoration curve $A_{res} = \int_{\tilde{t}_0}^{\tilde{t}_e} F(\tilde{t})d\tilde{t}$ normalized by the duration T, following the earthquake scenario and a repair scheduling policy π:

$$J(\pi) = E_{\pi}[A_{norm}] \tag{6.1}$$

where $A_{norm} = \frac{A_{res}}{T}$. Note that this formulation is not restricted to the water distribution system, but can be generalized to other infrastructure systems.

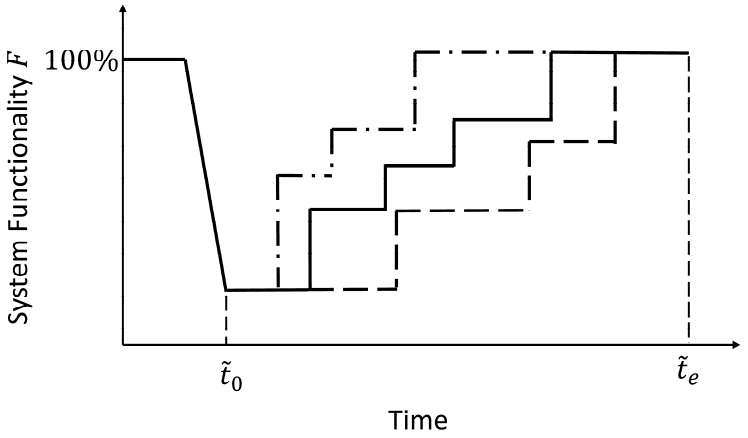


Figure 6.1 Schematic representation of the restoration curves of an infrastructure system after earthquake damage

6.3 Description of the Solution Methods

6.3.1 Markov Decision Process

The Markov Decision Process (MDP) is used to model the restoration process. The MDP is a general framework for sequential decision making, which is defined by the set of states $\mathcal{S}$, the set of actions $\mathcal{A}$, the probabilistic transition rule $\mathcal{P}: \mathcal{S} \times \mathcal{S} \times \mathcal{A} \rightarrow [0,1]$ and the reward function $\mathcal{R}$. The decision maker (agent) interacts with an environment at finite time steps $t = 0,1,2 \ldots, n$.

At each time step t, the agent observes the current state s_t, takes action a_t, as a result receives a reward r_{t+1} from the environment, and then observes the next state s_{t+1}. The process continues until a terminal state is reached (in episodic MDP) (Sutton and Barto 2018). The detailed review of MDP and reinforcement learning is provided in Chapter 5.

6.3.2 Deep Deterministic Policy Gradient Model

As an extension to the actor critic model as described in Chapter 5, Silver et al. 2014 proposed the deterministic policy gradient (DPG) algorithm, in which the actor utilizes a deterministic policy $\mu_\theta(s)$, and it is proved that a deterministic policy gradient exists (Silver et al. 2014). Lillicrap et al. (2015) further proposed the Deep Deterministic Policy Gradient (DDPG) algorithm to embed neural networks as function approximators in DPG. The algorithm includes an actor network $\mu(s|\theta^a)$, a critic network $Q(s,a|\theta^Q)$, and utilizes a replay buffer for learning. In addition, target networks are used for both actor and critic, denoted by $\mu'(s|\theta^{\mu'})$ and $Q'(s,a|\theta^{Q'})$. At each learning step, a random mini-batch is sampled, and the actor network is updated by the sampled gradient

$$\nabla_{\theta_\mu} J = \frac{1}{N}\sum_i \nabla_a Q(s,a|\theta^Q)|_{s=s_i,a=\mu(s_i)} \nabla_{\theta_\mu}\mu(s|\theta^\mu)|_{s_i} \tag{6.2}$$

The critic is updated similar to DQN, except that the target value is calculated using target the actor and critics

$$L = \frac{1}{N}\sum_i \left(r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'}) - Q(s_i,a_i|\theta^Q)\right)^2 \tag{6.3}$$

6.3.3 Extension to Problems with Large Discrete Action Space

In the application of post-earthquake restoration, a large discrete action space is

encountered, which is uncommon in typical DRL applications. For example, for a system consisting of 100 pipelines and 5 repair teams, the action space is on the order of 10^{10}. The problem then requires the output dimension of the network to be 10^{10}, and this can easily explode, and the softmax is not able to capture such huge set of actions.

The Wolpertinger algorithm proposed by Dulac-Arnold et al. (2015) is aimed at addressing the challenge of a large discrete action space. The core idea is to use DDPG as a base model which outputs a "proto" continuous action $\hat{a} = f_{\theta^\pi}(s)$, and find a k-nearest neighbor mapping to the discrete action space by the L_2 distance: $A_k = g_k(\hat{a}) = \underset{a \in A}{\overset{k}{\operatorname{argmin}}} \|a - \hat{a}\|_2$. The final output action is selected from A_k which has the highest Q value

$$a = \underset{a_j \in A_k}{\operatorname{argmax}} Q_{\theta^Q}(s, a_j) \tag{6.4}$$

For our application problem, we made several adaptations to the original Wolpertinger algorithm, as shown in Fig. 6.2:

1. When searching for candidate actions, first select k closest sub-actions dimension-wise

$$A_c^i = \underset{a_i}{\overset{k}{\operatorname{argmin}}} |a_i - \hat{a}_i| \tag{6.5}$$

where $\hat{a}_i$ is the ith element of the proto action. Then construct the candidate action set using the Cartesian product

$$A_c = \{(a_c^1, \dots a_c^d) | a_c^i \in A_c^i\} \tag{6.6}$$

In other words, the action forming step is decentralized and the action evaluation step is centralized.

2. The epsilon-greedy mechanism is used for action exploration. DDPG and Wolpertinger use a random process for exploration in the continuous action space, which is more appropriate when subsequent actions are highly correlated and within a neighboring region in nature. Such

exploration could be ineffective when the goal is to select from a set of discrete actions not necessarily correlated with the previous one. Hence, we apply epsilon-greedy mechanism directly in the discrete legal action space for exploration.

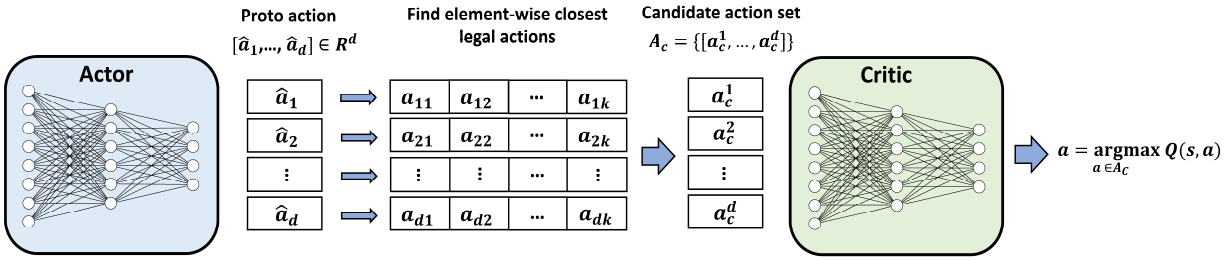


Figure 6.2 Schematic representation of the adapted Wolpertinger model

6.4 Framework

For the optimization of repair scheduling of the water network, a Markov Decision Process can be formulated as follows:

State: $s_t = [s_t, s_{t2}, \ldots, s_{tL}]$, where s_{tj} is the damage state of the jth pipeline

Action: $a_t = [a_{t1}, a_{t2}, \ldots, a_{td}]$, where d denotes the number of available repair crews, $a_{tj} \in \{1,2,\ldots,L\}$ determines the corresponding pipe to be repaired.

Reward: $r_{t+1}(s_t, a_t) = \sum_{j=1}^{d} F(s_t, a_{t[j-1]}) \cdot \left(\tilde{t}_{rep}(a_{t[j]}) - \tilde{t}_{rep}(a_{t[j-1]})\right)$, where $\tilde{t}_{rep}(a_{tj})$ denotes the duration of the repair of the jth team at time step t, $\{a_{t[1]}, \ldots, a_{t[d]}\}$ is the ordered set of a_t such that $\tilde{t}_{rep}(a_{t[1]}) \leq \tilde{t}_{rep}(a_{t[2]}) \leq \cdots \leq \tilde{t}_{rep}(a_{t[d]})$, $F(s_t, a_{tj})$ is the functionality after executing action a_{tj} at state s_t (obtained from hydraulic simulation). Let $F(s_t, a_{t[0]}) = F(s_{t-1}, a_{t[d]})$ and $\tilde{t}_{rep}(a_{t[0]}) = 0$.

We present a general framework for implementing the proposed DRL methodology as shown in Fig. 6.3:

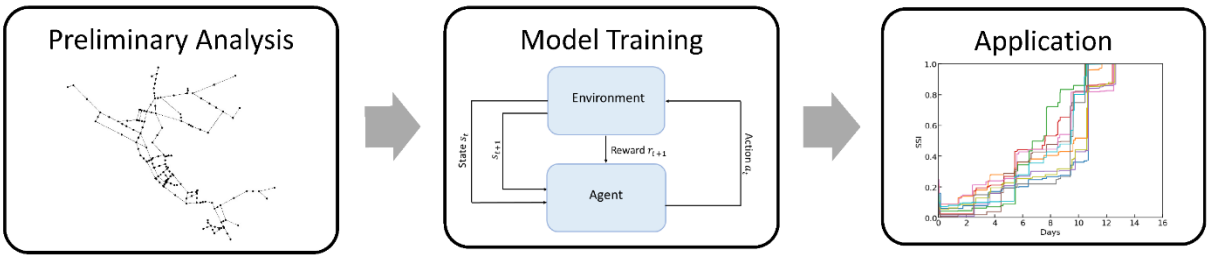


Figure 6.3 Framework of the DRL-based optimization

1. **Preliminary analysis**
   Based on the infrastructure system of interest, determine a reasonable initial distribution or method for resetting the system state at the beginning of each MDP episode, such that the trained agent can be generalized to a wide range of damage cases. Based on the size of the system, determine the appropriate model structure.

2. **Model training**
   Once the model is created, we train the model in the following process to the specified number of episodes:

   1) Episode start: the water distribution network is reset to the state s_0 from the designed initial distribution;

   2) At each time step of the episode:

   - The agent observes the water system state s_t and decides the next repair action a_t;
   - The environment receives the repair action a_t, repairs the corresponding pipes, obtains the $F(s_t, a_{tj})$ through hydraulic simulation, and transitions to the next state s_{t+1}, and gives r_{t+1} to the agent
   - The agent learns and updates its policy

   3) Episode terminates: all pipes are repaired, and the functionality of the water distribution network is fully restored (100%).

3. **Application**
   Generate a representative number of damage cases based on a possible suite of earthquake

scenarios for the network system of interest, apply the trained agent on each case.

6.5 Case study: Net3 water network

6.5.1 Network Description

In this section, the proposed DRL framework is illustrated using a case study on the benchmark system called "Net3", which is a skeleton version of the actual water distribution system in the city of Novato, Marin County, California. The Net3 system is used as examples by EPANET, WNTR and various studies for hydraulic and topological analysis (Diao et al. 2016; Klise et al. 2018; Meng et al. 2018).

The Novato Water System serves primarily the City of Novato and surrounding unincorporated areas in Marin County (North Marin Water District 2019). The estimated service area population is 61,381. The North Marin Water District (NMWD) water supply for the Novato Service area is derived from two sources: 1) surface water stored in Stafford Lake; and 2) Russian River water supplied by Sonoma County Water Agency (SCWA). The transmission system consists of 16- through 30-inch diameter pipelines strategically located to convey water supply to the distribution system. Larger diameter transmission system piping is generally constructed of steel or reinforced concrete pressure pipe. The majority of the distribution system is comprised of 6-, 8-, 10- and 12-inch diameter pipelines to distribute water from the transmission mains. Distribution system pipelines are constructed primarily of PVC, asbestos cement, and cast iron (North Marin Water District 2019).

The configuration of the Net3 system is shown in Fig. 6.4. The system has 2 Reservoirs, 3 tanks, 2 pump stations and 115 distribution pipelines. For the purpose of the current study, we need to generate damage from realistic earthquake scenarios. In this regard, the original hypothetical coordinates in the EPANET model of Net3 are calibrated to match with the Novato region to obtain

realistic locations of the components. The calibrated network coordinates ranges from 38.06° N to 38.12° N and 122.51°W to 122.61°W.

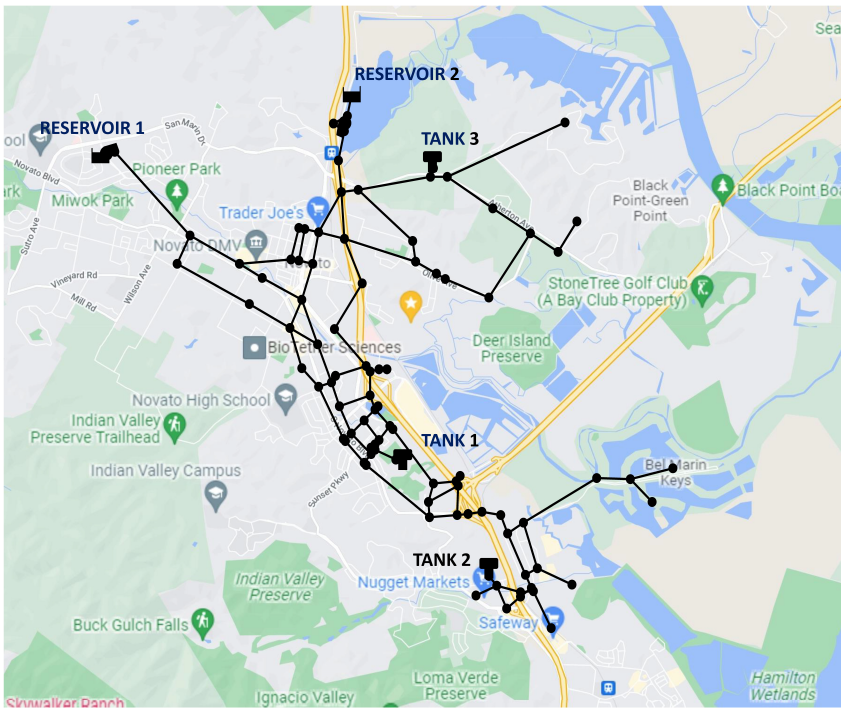


Figure 6.4 Water network system (Net3) in the City of Novato (Map data © 2021 Google)

In this case study, damage to pipelines are considered, and 3 repair teams are used for demonstration. The repair time for the pipeline are assumed to follow a triangular distribution with parameters listed in Table 6.1, as recommended by HAZUS-MH (FEMA 2012).

Table 6.1 Triangular distribution for the pipeline repair time

Damage State	Min	Mode	Max
Minor leak	3 hours	4 hours	6 hours
Break	4 hours	6 hours	12 hours

6.5.2 Model Implementation and Training

The simulation environment, named "WaterNet-v1", is implemented using OpenAI gym (Brockman et al. 2016). The core functionalities are similar to those of the environment described

in Section 5.3.3.

Based on a preliminary analysis, the DRL model is implemented as follows:

Actor Network: 3 fully connected hidden layers with 512, 256, 128 units and leaky ReLU activation. The output layer contains a single unit with sigmoid activation to limit the value to $(0,1)$, and multiplied by the number of pipelines of the network to produce a proto action $\hat{a}$ in the continuous space.

Critic Network: The hidden layer structure is similar to the actor network, except that the 1st hidden layer contains 512 units and concatenate with the input action. ReLU activation is used and a L_2 regularization is applied on the weights with parameter 0.1. The output layer contains a single unit with linear activation to produce the Q value $Q(s, a|\theta^Q)$.

For the training of the agent, the initial state of the environment is generated such that the initial damage can capture a wide range of possible cases, while focusing on a reasonable number of damaged components for effectiveness of training. At the beginning of each episode, the initial state is reset as follows:

1. The number of leaks is generated from $\mathrm{Triangular}(10,\ 60,\ 80)$ and round to the nearest integer
2. The number of breaks is set to 20% of the number of leaks
3. The corresponding number of damaged components is randomly assigned to the pipes

The training process stops after 10000 episodes following the defined initial distribution.

6.5.3 Scenario-based Damage Simulation

The performance of the proposed methodology is evaluated based on damage cases generated from scenario-based damage simulation. The $\mathbf{M_w}$ 7.9 event occurring on the San Andreas fault, with an epicenter location of 37.75° N, 122.55° W (included in UCERF2 maps) is

used as the scenario for demonstration. The median Peak Ground Velocity (PGV) for the grid of the pipelines are computed using Campbell and Bozorgnia (2014) ground motion attenuation equation, and the spatially correlated PGVs are generated using the model by Jayaram and Baker (2009) model considering the inter- and intra-event standard deviations as 0.297 and 0.494, respectively. Fig. 6.5 shows the generated spatially correlated PGV for the Net3 system, where the PGV values range from 66.51 to 334.62 cm/s.

For the Net3 system, three pipe damage states are considered: Undamaged (DS_0), Leak (DS_1) and Break (DS_2). The fragility curve of each damage state is defined by an exponential distribution over $RR \cdot L$, where RR is the repair rate, and L is the pipe length, as described in Eq 5.22. According to HAZUS (FEMA 2012), 80% of pipe damage is assumed as leaks and 20% is assumed as breaks. Given $RR \cdot L$, the probabilities of exceeding DS_1 and DS_2 can be determined, and multiple damage realizations can be generated. For the evaluation, 1000 Monte Carlo damage realizations are generated from the selected scenario. Fig 6.6 shows the distributions of the number of leaks and breaks among the 1000 simulated damage cases. The number of leaks ranges from 30 to 60 and the number of breaks ranges from 11 to 36.

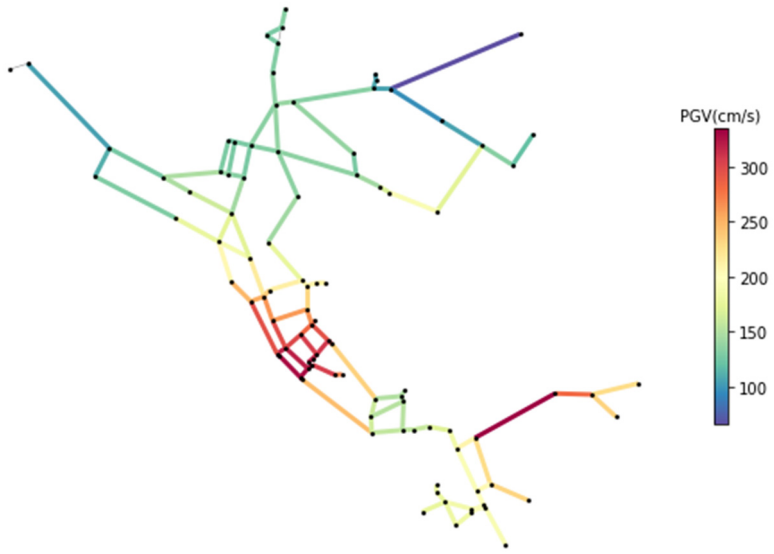
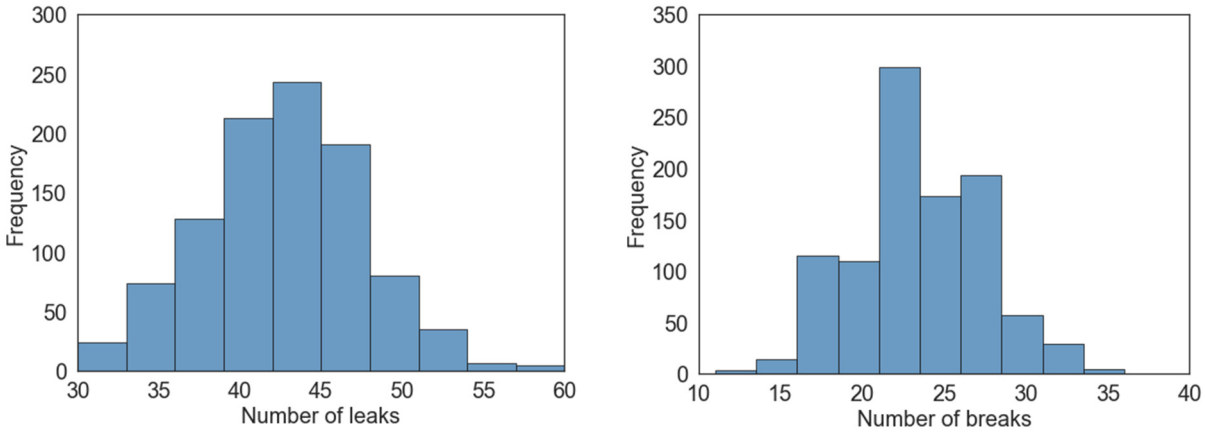


Figure 6.5 Distribution of the simulated PGV values



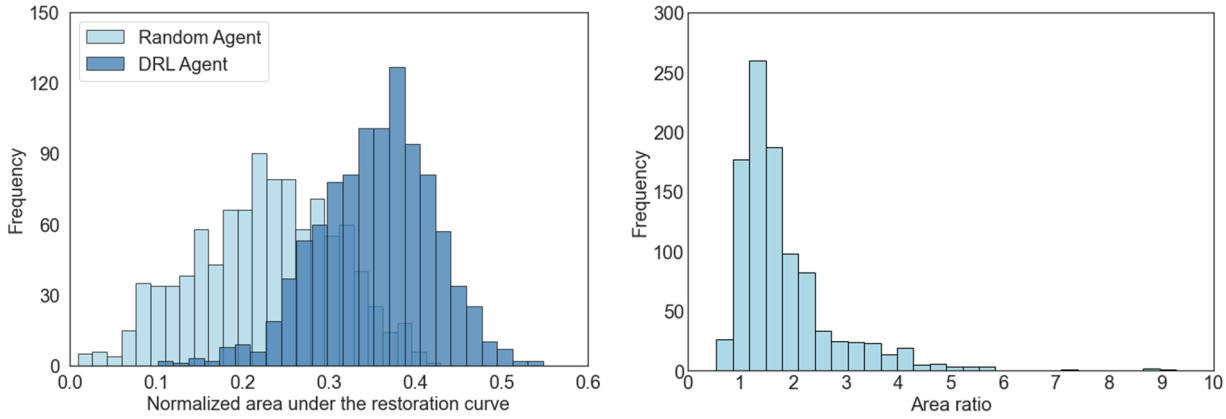
(a) (b)

Figure 6.6 Distributions of (a) number of leaks; and (b) number of breaks in 1000 damage simulations

6.5.4 Evaluation Results

For the evaluation, each damage case is input to the DRL-based optimization model to obtain a recommended sequence of repair, then a hydraulic recovery simulation is performed with the recommended repair sequence as input, assuming that the repair teams work from 9 a.m. to 6

p.m. Among the 1000 evaluated cases, the proportion of Wolpertinger agent winning is 90.9% (909/1000). Fig. 6.7 (a) shows the distributions of A_{norm} following the policies by the Wolpertinger and the random agents. The normalized area for the random agent has mean 0.227 and standard deviation 0.082, while the normalized area following Wolpertinger agent has mean 0.354 and standard deviation 0.066. In terms of the expected normalized area under the restoration curve following the earthquake scenario $E_{\pi}[A_{norm}]$, which is the optimization objective, the Wolpertinger agent improves over random agent 56.1%. In addition, the Wolpertinger agent reduces the dispersion of the normalized area among 1000 cases by 18.8%. Fig. 6.7 (b) presents the distribution of the area ratio ($A_{norm}^{wolp}/A_{norm}^{random}$). Here we observed that most of the area ratios are larger than one, and a great proportion are significantly larger than one.



(a) (b)

Figure 6.7 Distributions of (a) area under the restoration curve; and (b) area ratio between Wolpertinger and Random agent from the 1000 evaluated damage cases

Fig. 6.8 plots the restoration curves for the 1000 damage cases, where the repair crew are assumed to work continuously (day and night). It can be concluded that, the Wolpertinger agent successfully "pushes" the curves towards the upper-left direction. The curves distribute compactly around the diagonal line. The Wolpertinger agent significantly improves the restored SSI at given

time points during the restoration process. Fig. 6.9 shows the restoration curves for final evaluation where the repair crews are assumed to only allowed to work from 9 a.m. to 6 p.m. The improvement of the Wolpertinger agent over the random agent is as significant as in Fig. 6.8. Therefore, the recommended schedules from the optimization model can work well for real situations with more constraints.

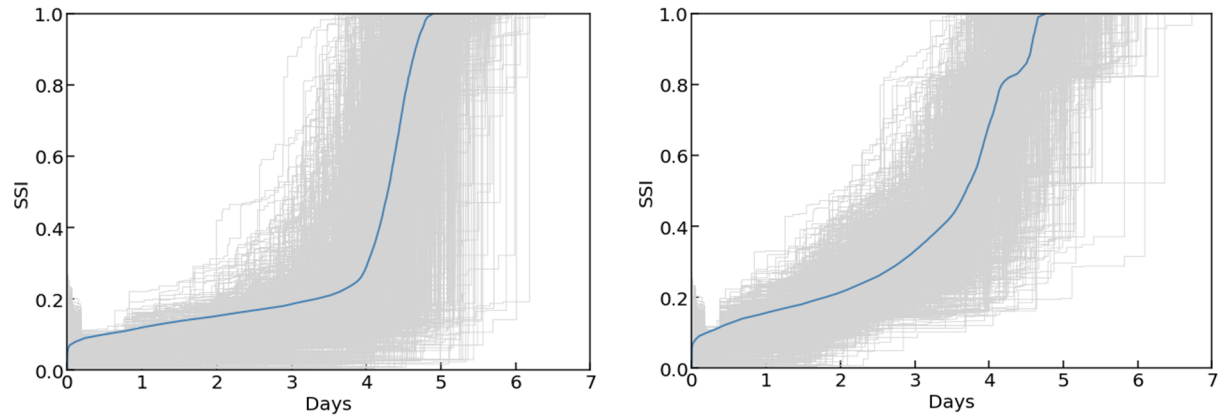


Figure 6.8 Restoration curves for 1000 evaluated damage cases (working continuously): (a) Random Agent; (b) Wolpertinger Agent

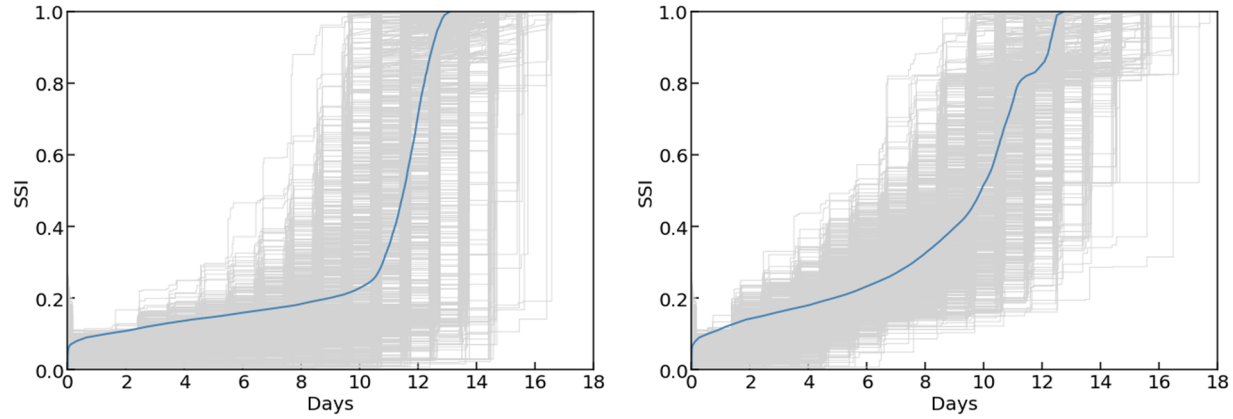


Figure 6.9 Restoration curves for 1000 evaluated damage cases (working from 9 a.m. to 6 p.m.) : (a) Random Agent; (b) Wolpertinger Agent

6.6 Conclusion

This study proposed a general framework for improving the post-earthquake repair scheduling policy based on the deep reinforcement learning (DRL) method. The Markov Decision Process was used to model the restoration process. A system model was constructed using the Water Network Tool for Resilience (WNTR) package and used to perform real-time hydraulic analysis. The network model was then embedded in a customized reinforcement learning environment written using the OpenAI gym platform. The actor-critic based model, Wolpertinger, was adapted to address the challenge of a large discrete action space and learn the near optimal repair policy by interacting with the water network environment.

To evaluate the proposed methodology, a water distribution network in the city of Novato, North Marin, California was used for demonstration, which consists of 115 pipelines. 1000 damage cases were simulated from the scenario-based simulation, and the normalized area under the restoration curve from the DRL policy is compared with the base policy of randomly repairs from the damaged pipes. The Wolpertinger won in 90.9% (909/1000) of the cases, and improved the objective value by 56.1%. The evaluation results demonstrated the effectiveness of the DRL framework for the repair optimization for realistic systems. The proposed framework can be generalized to other hazards (e.g., flood) and distributed infrastructure (e.g., power networks).

For future work, several challenges need to be addressed. The deep reinforcement learning methodology is computationally expensive, which comes from two aspects: learning the parameters of a deep agent model and interaction with a physical model of the system. To address these challenges, the simplification of the agent model structure can be studied; A surrogate model of the system can be developed, to alleviate the expensive computation of the real time response of the system in each interaction. In addition, the framework assumes the damage evaluation phase

has been completed before the repair, and the optimization focuses on the repair sequence. It is important to explore the possibility of incorporating the inspection decision in the optimization.

7. Conclusion

7.1 Overview

Decision-making in the post-earthquake environment is critical to infrastructure and community resilience. The main objective of the research is to develop a general framework that supports reliable damage assessment, recovery estimation and repair action optimization for distributed infrastructure following earthquakes, based on data-driven and machine learning methodologies. The framework can provide insights on the post-earthquake state of infrastructure, improve pre-event planning, and inform optimal decisions for the tasks involved in the recovery process. The methodology was validated through two infrastructure systems: the water distribution networks in the City of Napa and the City of Novato, subjected to real and simulated earthquake scenarios.

7.2 Findings

Chapter 3 presented a general framework for dynamically-updating damage assessments for post-earthquake recovery of distributed infrastructure. The water distribution network for the City of Napa, California was used as an application testbed. Previous studies mostly aimed at developing models to predict damage for future earthquakes i.e., pre-event assessments. However, less effort has been placed on utilizing the information collected at certain time points and a few locations within the network during the recovery to inform the damage estimation of the entire system, and further help update the inspection and repair plans. As a first step of this work, a neural network-based pre-event assessment model was developed, using the data for the pipeline damage from the 2014 south Napa earthquake. Next, a graph neural network model was applied to update the damage estimate via semi-supervised learning given the information incrementally collected during the inspection process.

Chapter 4 developed a spatially explicit recovery model for the water distribution network. Previous models often establish the statistical relationships between the time parameters for recovery and several explanatory variables of individual components. The accuracy of the modeling can be further improved by incorporating the spatial correlations between the components in a distributed infrastructure. The recovery was assessed in this work using the Gaussian Process model where the spatial correlation was explicitly considered. The recovery data from the 2014 south Napa earthquake was used for demonstration purposes. The capability of the model to predicting the recovery for future earthquakes is demonstrated using a simulated $\mathbf{M_w}$ 6.7 event.

Chapter 5 proposed a deep reinforcement learning (DRL) based methodology to optimize the repair scheduling for the post-earthquake recovery of distributed infrastructure. A mathematical formulation of the restoration process and the optimization problem was established based on the Markov Decision Process (MDP). To study the applicability of the methodology, two commonly used DRL models: Deep Q Network (DQN) and Deep Actor Critic (DAC) were applied to a hypothetical water network. Chapter 6 further expanded the proposed methodology by considering the challenge of a large discrete action space for the repair optimization in the recovery of realistic large systems. The Wolpertinger based DRL model was adapted to address the challenge. The water distribution system in Novato was used and 1000 damage cases were generated from scenario-based simulation to evaluate the modified framework.

7.3 Limitations and Future Work

The main objective of this dissertation is to develop a machine learning-based framework that supports various types of assessments and decision-making procedures (damage assessment, recovery estimation and repair optimization). The framework was demonstrated through a series

of scenarios and systems. The potential and effectiveness of the machine learning methodology has been demonstrated through the case studies in the dissertation.

It needs to be noted, however, that the performance of the machine learning methodology largely depends on the quality of data and the selection of appropriate model, and the applicability of such methods in practice is limited by the complexity of use and computational expense.

For the scenario earthquake simulations, the ground motion intensity at mid-point of the pipeline was used to represent the intensity over the entire length. In other words, variations in the intensity along the pipe length were not considered. This assumption may be appropriate for shorter pipes but for the ones with significant length, this could reduce the reliability of the damage assessment. Future work should use a higher fidelity distribution to characterize the shaking intensities.

For the highly imbalanced dataset of earthquake damage in Chapter 3, which is representative for other real earthquake scenarios where only a small portion of the infrastructure is damaged, the machine learning model tends to overestimate the damage, which may lead to conservative decisions and plans. Further studies need to be carried out to address the challenge of imbalanced data.

The damage information used to train the graph neural networks was assumed to be obtained by inspection, but the reliability of the inspection was not considered. In other words, the information from the inspection was assumed to be perfect. Future work should focus on accounting for the possibility of mis-classification of damage during inspections.

Detailed discussions on the interpretation of the graph neural network model and the relative importance of the features were not carried out.

The recovery model in Chapter 4 was developed on a single earthquake scenario, and the

capability for estimating the trajectories from future earthquakes was demonstrated through a simulated scenario. It was found that the transfer learning or extrapolation capability highly depends on the number of data points on which the model is developed. Hence, the model can be further improved by learning from a series of different scenarios. The very high recovery rate in the early stage of the recovery was not well-captured by the Gaussian Process model. Future studies can incorporate information regarding the change of repair resources as a feature in the recovery model to better address this challenge.

The reinforcement learning based optimization model proposed in Chapter 5 and Chapter 6 is computationally expensive, because the model was trained by interacting with an environment. In the case for infrastructure systems, this means performing a response simulation of the complex model of the real system in each interaction, which limits the applicability of the model in practice. Therefore, in this study, the methodology was evaluated on a moderate system where the number of components was on the order of hundreds. To further improve the applicability of the model, measures need to be proposed for both the agent and the environment models. For example, a surrogate model can be used to predict the response of the infrastructure system instead of running real simulations, and the structure of the reinforcement learning model can be simplified to reduce the learning time. In addition, further studies can be carried out to combine the suggestions from the reinforcement learning model and the actions of human engineers in deciding the final repair sequences.

8. References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mane, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viegas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. (2016). "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems."

American Lifelines Alliance. (2001). *Seismic fragility formulations for water systems: Part 1 - Guideline*.

Andriotis, C. P., and Papakonstantinou, K. G. (2019). "Managing engineering systems with large state and action spaces through deep reinforcement learning." *Reliability Engineering and System Safety*, 191.

Arulkumaran, K., Deisenroth, M. P., Brundage, M., and Bharath, A. A. (2017). "Deep reinforcement learning: A brief survey." *IEEE Signal Processing Magazine*.

Baeza-Yates, R., and Ribeiro-Neto, B. (2011). *Modern Information Retrieval - the concepts and technology behind search, Second edition*.

Bagriacik, A., Davidson, R. A., Hughes, M. W., Bradley, B. A., and Cubrinovski, M. (2018). "Comparison of statistical and machine learning approaches to modeling earthquake damage to water pipelines." *Soil Dynamics and Earthquake Engineering*, Elsevier Ltd, 112, 76–88.

Barker, K., and Baroud, H. (2014). "Proportional hazards models of infrastructure system recovery." *Reliability Engineering & System Safety*, Elsevier, 124, 201–206.

Bellagamba, X., Bradley, B. A., Wotherspoon, L. M., and Lagrava, W. D. (2019). "A Decision-

Support Algorithm for Post-Earthquake Water Services Recovery and Its Application to the 22 February 2011 M w 6.2 Christchurch Earthquake." *Earthquake Spectra*, Earthquake Engineering Research Institute, 35(3), 1397–1420.

Bellman, R. E. (2003). *Dynamic Programming*. Dover Publications, Inc., New York, NY, USA.

Bertsekas, D. P. (1999). "Rollout algorithms: an overview." *Proceedings of the 38th IEEE Conference on Decision and Control (Cat. No.99CH36304)*, IEEE, 448–449.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. (2016). "OpenAI Gym."

Bruneau, M., Chang, S. E., Eguchi, R. T., Lee, G. C., O'Rourke, T. D., Reinhorn, A. M., Shinozuka, M., Tierney, K., Wallace, W. A., and von Winterfeldt, D. (2003). "A Framework to Quantitatively Assess and Enhance the Seismic Resilience of Communities." *Earthquake Spectra*, 19(4), 733–752.

Çağnan, Z., and Davidson, R. A. (2007). "Discrete event simulation of the post-earthquake restoration process for electric power systems." *International Journal of Risk Assessment and Management*, Inderscience Publishers, 7(8), 1138–1156.

Cao, Q. D., Miles, S. B., and Choe, Y. (2020). "Infrastructure Recovery Curve Estimation Using Gaussian Process Regression on Expert Elicited Data."

Cavdaroglu, B., Hammel, E., Mitchell, J. E., Sharkey, T. C., and Wallace, W. A. (2013). "Integrating restoration and scheduling decisions for disrupted interdependent infrastructure systems." *Annals of Operations Research*, Springer US, 203(1), 279–294.

Chawla, N. V., Bowyer, K. W., Hall, L. O., and Kegelmeyer, W. P. (2002). "SMOTE: Synthetic minority over-sampling technique." *Journal of Artificial Intelligence Research*, American Association for Artificial Intelligence, 16, 321–357.

Choi, J., Yoo, D. G., and Kang, D. (2018). "Post-earthquake restoration simulation model for water supply networks." *Sustainability (Switzerland)*, MDPI AG, 10(10).

Cimellaro, G. P., Reinhorn, A. M., and Bruneau, M. (2010). "Framework for analytical quantification of disaster resilience." *Engineering Structures*, Elsevier, 32(11), 3639–3649.

Crowl, D. A., Louvar, J. F., and Rocco, C. M. (2002). *Chemical Process Safety: Fundamentals with Applications*. Upper Saddle River, NJ: Prentice Hall.

Crowley, H., and Bommer, J. J. (2006). "Modelling seismic hazard in earthquake loss models with spatially distributed exposure." *Bulletin of Earthquake Engineering*, 4(3), 249–273.

Csáji, B. (2001). "Approximation with artificial neural networks." *MSc. thesis*, 45.

Das, N., and Yip, M. (2020). "Learning-Based Proxy Collision Detection for Robot Motion Planning Applications." *IEEE Transactions on Robotics*.

Decò, A., Bocchini, P., and Frangopol, D. M. (2013). "A probabilistic approach for the prediction of seismic resilience of bridges." *Earthquake Engineering & Structural Dynamics*, John Wiley & Sons, Ltd, 42(10), 1469–1487.

DHS. (2013). *National Infrastructure Preparedness Plan, NIPP 2013*.

Dhulipala, S. L. N., Burton, H. V., and Baroud, H. (2021). "A Markov framework for generalized post-event systems recovery modeling: From single to multihazards." *Structural Safety*, Elsevier, 91, 102091.

Diao, K., Sweetapple, C., Farmani, R., Fu, G., Ward, S., and Butler, D. (2016). "Global resilience analysis of water distribution systems." *Water Research*, Elsevier Ltd, 106, 383–393.

Didier, M., Sun, L., Ghosh, S., and Stojadinovic, B. (2015). "Post-earthquake recovery of a community and its electrical power supply system." *COMPDYN 2015 - 5th ECCOMAS Thematic Conference on Computational Methods in Structural Dynamics and Earthquake*

Engineering, National Technical University of Athens, 1451–1461.

Dulac-Arnold, G., Evans, R., van Hasselt, H., Sunehag, P., Lillicrap, T., Hunt, J., Mann, T., Weber, T., Degris, T., and Coppin, B. (2015). "Deep Reinforcement Learning in Large Discrete Action Spaces."

Eidinger, J. (2001). *Seismic Fragility Formulations for Water Systems Web Site Report Appendices*.

Eidinger, J., Tang, A. K., Fujisaki, E., Sun, J., and Trinh, R. (2016). "Seismic Fragility of Power Distribution Systems." *International Collaboration in Lifeline Earthquake Engineering 2016 - Proceedings of the 7th China-Japan-US Trilateral Symposium on Lifeline Earthquake Engineering*, 212–218.

Eskandarpour, R., and Khodaei, A. (2017). "Machine Learning Based Power Grid Outage Prediction in Response to Extreme Events." *IEEE Transactions on Power Systems*, Institute of Electrical and Electronics Engineers Inc., 32(4), 3315–3316.

Farahani, S., Tahershamsi, A., and Behnam, B. (2020). "Earthquake and post-earthquake vulnerability assessment of urban gas pipelines network." *Natural Hazards*, Springer, 101(2), 327–347.

Federal Emergency Management Agency (FEMA). (2003). "HAZUS-MH MR4 Multi-Hazar d Loss Estimation Methodology – Earthquake Model: Technical Manual. Department of Homeland Security." *Federal Emergency Management Agency, Washington, ….*

FEMA. (2012). "Hazus-MH 2.1 Technical Manual: Earthquake Model." Federal Emergency Management Agency, Mitigation Division Washington, DC.

Fishman, G. S. (1978). "Principles of Discrete Event Simulation." John Wiley & Sons, Inc.

Fishman, G. S. (2001). "Discrete-Event Simulation: Modeling, Programming, and Applications." New York: Springer-Verlag.

Gol, E. A., Erkal, B. G., and Göl, M. (2019). "A Novel MDP Based Decision Support Framework to Restore Earthquake Damaged Distribution Systems." *Proceedings of 2019 IEEE PES Innovative Smart Grid Technologies Europe, ISGT-Europe 2019*, Institute of Electrical and Electronics Engineers Inc.

Gomez, C., and Baker, J. W. (2019). "An optimization-based decision support framework for coupled pre- and post-earthquake infrastructure risk management." *Structural Safety*, Elsevier, 77, 1–9.

Goswami, T., and Roy, U. B. (2019). "Predictive Model for Classification of Power System Faults using Machine Learning." *IEEE Region 10 Annual International Conference, Proceedings/TENCON*, Institute of Electrical and Electronics Engineers Inc., 2019-October, 1881–1885.

Grondman, I., Busoniu, L., Lopes, G. A. D., and Babuška, R. (2012). "A survey of actor-critic reinforcement learning: Standard and natural policy gradients." *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews*.

Isoyama, R., Ishida, E., Yune, K., and Shirozu, T. (2000). "Seismic Damage Estimation Procedure for Water Supply Pipelines."

Jayaram, N., and Srinivasan, K. (2008). "Performance-based optimal design and rehabilitation of water distribution networks using life cycle costing." 44, 1417.

Kang, D., and Lansey, K. (2013). "Post-earthquake Restoration of Water Supply Infrastructure." *World Environmental and Water Resources Congress 2013: Showcasing the Future - Proceedings of the 2013 Congress*, American Society of Civil Engineers, 913–922.

Kang, H. (2018). "From Rupture to Recovery: Integrating Probabilistic Building Performance Assessment, Decision-Making and Socioeconomic Vulnerability to Model Post-earthquake

Housing Recovery." *ProQuest Dissertations and Theses*, (1538747), 157.

Kang, H., Burton, H. V., and Miao, H. (2018). "Replicating the recovery following the 2014 south Napa Earthquake using stochastic process models." *Earthquake Spectra*, Earthquake Engineering Research Institute, 34(3), 1247–1266.

Kingma, D. P., and Ba, J. (2014). "Adam: A Method for Stochastic Optimization."

Kipf, T. N., and Welling, M. (2016). "Semi-Supervised Classification with Graph Convolutional Networks."

Klise, K. A., Bynum, M., Moriarty, D., and Murray, R. (2017). "A software framework for assessing the resilience of drinking water systems to disasters with an example earthquake case study." *Environmental Modelling and Software*, Elsevier, United States, 95, 420–431.

Klise, K. A., Murray, R., and Haxton, T. (2018). "An Overview of the Water Network Tool for Resilience (WNTR)." *WDSA / CCWI Joint Conference Proceedings*.

Konda, V. R., and Tsitsiklis, J. N. (2003). "On Actor-Critic Algorithms." *SIAM Journal on Control and Optimization*, 42(4), 1143–1166.

Kozin, F., and Zhou, H. (1990). "System Study of Urban Response and Reconstruction due to Earthquake." *Journal of Engineering Mechanics*, American Society of Civil Engineers, 116(9), 1959–1972.

Larock, B., Jeppson, R., and Watters, G. (2010). *Hydraulics of Pipeline Systems*. *Hydraulics of Pipeline Systems*, CRC Press.

Laucelli, D., and Giustolisi, O. (2015). "Vulnerability Assessment of Water Distribution Networks under Seismic Actions." *Journal of Water Resources Planning and Management*, American Society of Civil Engineers (ASCE), 141(6), 04014082.

Lee, J.-H., and Labadie, J. W. (2007). "Stochastic optimization of multireservoir systems via

reinforcement learning." *Water Resources Research*, John Wiley & Sons, Ltd, 43(11), W11408.

Lee, J. Y., Tomar, A., and Burton, H. (2018). "A Framework for Water Distribution Systems Exposed to Seismic Events and Evolving Conditions." *11th US National Conference on Earthquake Engineering*, Los Angeles, CA.

Liao, H., Zhang, W., Dong, X., Poczos, B., Shimada, K., and Kara, L. B. (2019). "A Deep Reinforcement Learning Approach for Global Routing."

Lillicrap, T. P., Hunt, J. J., Pritzel, A., Heess, N., Erez, T., Tassa, Y., Silver, D., and Wierstra, D. (2015). "Continuous control with deep reinforcement learning."

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. (2017). "Multi-Agent Actor-Critic for Mixed Cooperative-Competitive Environments."

Van Der Maaten, L., and Hinton, G. (2008). *Visualizing Data using t-SNE*. *Journal of Machine Learning Research*.

MacKenzie, C. A., and Barker, K. (2012). "Empirical Data and Regression Analysis for Estimation of Infrastructure Resilience with Application to Electric Power Outages." *Journal of Infrastructure Systems*, American Society of Civil Engineers, 19(1), 25–35.

Mangalathu, S., Asce, A. M., and Jeon, J.-S. (2020). "Regional Seismic Risk Assessment of Infrastructure Systems through Machine Learning: Active Learning Approach."

Mangalathu, S., and Jeon, J.-S. (2019). "Machine Learning–Based Failure Mode Recognition of Circular Reinforced Concrete Bridge Columns: Comparative Study." *Journal of Structural Engineering*, American Society of Civil Engineers (ASCE), 145(10), 04019104.

Martell, M., Miles, S., and Choe, Y. (2020). "Modeling of Lifeline Infrastructure Restoration Using Empirical Quantitative Data." *Natural Hazards Review*, American Society of Civil

Engineers (ASCE), 22(4), 03121001.

Masoomi, H., Burton, H., Tomar, A., and Mosleh, A. (2020). "Simulation-Based Assessment of Postearthquake Functionality of Buildings with Disruptions to Cross-Dependent Utility Networks." *Journal of Structural Engineering*, American Society of Civil Engineers (ASCE), 146(5), 04020070.

Medury, A., and Madanat, S. (2013). "Incorporating network considerations into pavement management systems: A case for approximate dynamic programming." *Transportation Research Part C: Emerging Technologies*, Pergamon, 33, 134–150.

Mehralian, H., and Azarbakht, A. (2020). "Seismic loss assessment: the case study of the power distribution network in Arak city, Iran." *Journal of Civil Engineering and Materials Application*, 4(4), 195–207.

Memarzadeh, M., and Pozzi, M. (2019a). "Model-free reinforcement learning with model-based safe exploration: Optimizing adaptive recovery process of infrastructure systems." *Structural Safety*, 80, 46–55.

Memarzadeh, M., and Pozzi, M. (2019b). "Model-free reinforcement learning with model-based safe exploration: Optimizing adaptive recovery process of infrastructure systems." *Structural Safety*, Elsevier, 80, 46–55.

Meng, F., Fu, G., Farmani, R., Sweetapple, C., and Butler, D. (2018). "Topological attributes of network resilience: A study in water distribution systems." *Water Research*, Elsevier Ltd, 143, 376–386.

Mishalani, R. G., and Madanat, S. M. (2002). "Computation of Infrastructure Transition Probabilities Using Stochastic Duration Models." *Journal of Infrastructure Systems*, 8(4), 139–148.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. (2015). "Human-level control through deep reinforcement learning." *Nature*, 518, 529–533.

Mojtahedi, M., Newton, S., and Von Meding, J. (2017). "Predicting the resilience of transport infrastructure to a natural disaster using Cox's proportional hazards regression model." *Natural Hazards*, Springer Netherlands, 85(2), 1119–1133.

Monsalve, M., and de la Llera, J. C. (2019). "Data-driven estimation of interdependencies and restoration of infrastructure systems." *Reliability Engineering & System Safety*, Elsevier, 181, 167–180.

Murphy, K. P. (2012). "Machine learning: a probabilistic perspective."

Napa Water Division. (2017). *Urban water management plan: 2015 update*. Napa, CA.

Nayak, M. A., and Turnquist, M. A. (2016). "Optimal Recovery from Disruptions in Water Distribution Networks." *Computer-Aided Civil and Infrastructure Engineering*, John Wiley & Sons, Ltd, 31(8), 566–579.

Nejat, A., and Damnjanovic, I. (2012a). "Modeling Dynamics of Post-Disaster Recovery." *Construction Research Congress 2012: Construction Challenges in a Flat World, Proceedings of the 2012 Construction Research Congress*, American Society of Civil Engineers, 2200–2210.

Nejat, A., and Damnjanovic, I. (2012b). *MODELING DYNAMICS OF POST-DISASTER RECOVERY*.

North Marin Water District. (2019). *2018 Novato Water System Master Plan Update*.

Nozhati, S., Sarkale, Y., Chong, E. K. P., and Ellingwood, B. R. (2020). "Optimal stochastic

dynamic scheduling for managing community recovery from natural hazards." *Reliability Engineering and System Safety*, 193.

Nozhati, S., Sarkale, Y., Ellingwood, B., K.P. Chong, E., and Mahmoud, H. (2019). "Near-optimal planning using approximate dynamic programming to enhance post-hazard community resilience management." *Reliability Engineering and System Safety*, 181, 116–126.

Nozhati, S., Sarkale, Y., Ellingwood, B. R., Chong, E. K. P., and Mahmoud, H. (2018). "A Modified Approximate Dynamic Programming Algorithm for Community-level Food Security Following Disasters."

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Köpf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). "PyTorch: An Imperative Style, High-Performance Deep Learning Library." *arXiv*, arXiv.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., and Dubourg, V. (2011). "Scikit-learn: Machine learning in Python." *the Journal of machine Learning research*, JMLR. org, 12, 2825–2830.

Pineda-Porras, O., and Najafi, M. (2010). "Seismic Damage Estimation for Buried Pipelines: Challenges after Three Decades of Progress." *Journal of Pipeline Systems Engineering and Practice*, American Society of Civil Engineers (ASCE), 1(1), 19–24.

Popova, M., Isayev, O., and Tropsha, A. (2018). "Deep reinforcement learning for de novo drug design." *Science Advances*, American Association for the Advancement of Science, 4(7), eaap7885.

Sanchez-Lengeling, B., and Aspuru-Guzik, A. (2018). "Inverse molecular design using machine learning: Generative models for matter engineering." *Science (New York, N.Y.)*, American

Association for the Advancement of Science, 361(6400), 360–365.

Sarkale, Y., Nozhati, S., Chong, E. K. P., Ellingwood, B. R., and Mahmoud, H. (2018). "Solving Markov decision processes for network-level post-hazard recovery via simulation optimization and rollout." *IEEE International Conference on Automation Science and Engineering*, Munich, Germany.

Seeger, M. (2004). "Gaussian processes for machine learning." *International journal of neural systems*, World Scientific, 14(02), 69–106.

Sheibani, M., and Ou, G. (2020). "The development of Gaussian process regression for effective regional post-earthquake building damage inference." *Computer-Aided Civil and Infrastructure Engineering*, Blackwell Publishing Inc., 36(3), mice.12630.

Sherstinsky, A. (2018). "Fundamentals of Recurrent Neural Network (RNN) and Long Short-Term Memory (LSTM) Network." *Physica D: Nonlinear Phenomena*, Elsevier B.V., 404.

Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., van den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M., Dieleman, S., Grewe, D., Nham, J., Kalchbrenner, N., Sutskever, I., Lillicrap, T., Leach, M., Kavukcuoglu, K., Graepel, T., and Hassabis, D. (2016). "Mastering the game of Go with deep neural networks and tree search." *Nature*, Nature Publishing Group, 529(7587), 484–489.

Silver, D., Lever, G., Heess, N., Degris, T., Wierstra, D., and Riedmiller, M. (2014). "Deterministic policy gradient algorithms." *31st International Conference on Machine Learning, ICML 2014*, 605–619.

Silver, D., Schrittwieser, J., Simonyan, K., Antonoglou, I., Huang, A., Guez, A., Hubert, T., Baker, L., Lai, M., Bolton, A., Chen, Y., Lillicrap, T., Hui, F., Sifre, L., van den Driessche, G., Graepel, T., and Hassabis, D. (2017). "Mastering the game of Go without human knowledge."

Nature, Nature Publishing Group, 550(7676), 354–359.

Soleimani, N., Davidson, R. A., Davis, C., O'Rourke, T. D., and Nozick, L. K. (2021). "Multihazard Scenarios for Regional Seismic Risk Assessment of Spatially Distributed Infrastructure." *Journal of Infrastructure Systems*, American Society of Civil Engineers (ASCE), 27(1), 04021001.

Sun, H., Burton, H. V., and Huang, H. (2020). "Machine Learning Applications for Building Structural Design and Performance Assessment: State-of-the-Art Review." *Journal of Building Engineering*, Elsevier BV, 101816.

Sun, H., Burton, H., Zhang, Y., and Wallace, J. (2018). "Interbuilding interpolation of peak seismic response using spatially correlated demand parameters." *Earthquake Engineering & Structural Dynamics*, John Wiley and Sons Ltd, 47(5), 1148–1168.

Sutton, R. S., and Barto, A. G. (2018). *Reinforcement learning : an introduction*. MIT Press, Cambridge, MA.

Sutton, R. S., McAllester, D., Singh, S., and Mansour, Y. (1999). "Policy gradient methods for reinforcement learning with function approximation." *Proceedings of the 12th International Conference on Neural Information Processing Systems*, MIT Press, 1057–1063.

Tabucchi, T., Davidson, R., and Brink, S. (2008). "Restoring the Los Angeles water supply system following an earthquake." *14th World Conference on Earthquake Engineering*.

Tabucchi, T., Davidson, R., and Brink, S. (2010a). "Simulation of post-earthquake water supply system restoration Simulation of post-earthquake water supply system restoration." *Civil Engineering and Environmental Systems*, 27(4), 263–279.

Tabucchi, T., Davidson, R., and Brink, S. (2010b). "Simulation of post-earthquake water supply system restoration." *Civil Engineering and Environmental Systems*, Taylor & Francis , 27(4),

263–279.

Tabucchi, T. H. P., and Davidson, R. A. (2008). *Post-Earthquake Restoration of the Los Angeles Water Supply System*.

Tomar, A., and Burton, H. V. (2021a). "Risk-based assessment of the post-earthquake functional disruption and restoration of distributed infrastructure systems." *International Journal of Disaster Risk Reduction*, Elsevier Ltd, 52.

Tomar, A., and Burton, H. V. (2021b). "Active learning method for risk assessment of distributed infrastructure systems." *Computer-Aided Civil and Infrastructure Engineering*, Wiley, 36(4), 438–452.

Tomar, A., Burton, H. V., Mosleh, A., and Yun Lee, J. (2020). "Hindcasting the Functional Loss and Restoration of the Napa Water System Following the 2014 Earthquake Using Discrete-Event Simulation." *Journal of Infrastructure Systems*, 26(4), 04020035.

Varga, A. (2001). "Discrete event simulation system." *Proc. of the European Simulation Multiconference (ESM'2001)*.

Veličković, P., Casanova, A., Liò, P., Cucurull, G., Romero, A., and Bengio, Y. (2018). "Graph attention networks." *6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings*, International Conference on Learning Representations, ICLR.

Vinyals, O., Ewalds, T., Bartunov, S., Georgiev, P., Vezhnevets, A. S., Yeo, M., Makhzani, A., Küttler, H., Agapiou, J., Schrittwieser, J., Quan, J., Gaffney, S., Petersen, S., Simonyan, K., Schaul, T., van Hasselt, H., Silver, D., Lillicrap, T., Calderone, K., Keet, P., Brunasso, A., Lawrence, D., Ekermo, A., Repp, J., and Tsing, R. (2017). "StarCraft II: A New Challenge for Reinforcement Learning." *arXiv preprint arXiv:1708.04782*.

Watkins, C. (1989). *Learning From Delayed Rewards*.

Winkler, D., Haltmeier, M., Kleidorfer, M., Rauch, W., and Tscheikner-Gratl, F. (2018). "Pipe failure modelling for water distribution networks using boosted decision trees." *Structure and Infrastructure Engineering*, Taylor and Francis Ltd., 14(10), 1402–1411.

Wu, B., Akinola, I., and Allen, P. K. (2019a). "Pixel-Attentive Policy Gradient for Multi-Fingered Grasping in Cluttered Scenes." *IEEE International Conference on Intelligent Robots and Systems*, 1789–1796.

Wu, J., and Baker, J. W. (2020). *Statistical Learning Techniques for the Estimation of Lifeline Network Performance and Retrofit Selection*.

Wu, Y. N., Gao, R., Han, T., and Zhu, S.-C. (2018). "A Tale of Three Probabilistic Families: Discriminative, Descriptive, and Generative Models." *Quarterly of Applied Mathematics*, 77(2), 423–465.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S. (2019b). "A comprehensive survey on graph neural networks." *arXiv*.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S. (2019c). "A comprehensive survey on graph neural networks." *arXiv*, arXiv.

Xu, M., Ouyang, M., Mao, Z., and Xu, X. (2019). "Improving repair sequence scheduling methods for postdisaster critical infrastructure systems." *Computer-Aided Civil and Infrastructure Engineering*, 34(6), 506–522.

Xu, N., Guikema, S. D., Davidson, R. A., Nozick, L. K., Çağnan, Z., and Vaziri, K. (2007). "Optimizing scheduling of post-earthquake electric power restoration tasks." *Earthquake Engineering & Structural Dynamics*, John Wiley & Sons, Ltd, 36(2), 265–284.

Yao, L., Dong, Q., Jiang, J., and Ni, F. (2020). "Deep reinforcement learning for long-term

pavement maintenance planning." *Computer-Aided Civil and Infrastructure Engineering*, Blackwell Publishing Inc., mice.12558.

Zhou, J., Cui, G., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., and Sun, M. (2018). "Graph Neural Networks: A Review of Methods and Applications." *arXiv*.

Zhu, Y., Mottaghi, R., Kolve, E., Lim, J. J., Gupta, A., Fei-Fei, L., and Farhadi, A. (2017). "Target-driven visual navigation in indoor scenes using deep reinforcement learning." *Proceedings - IEEE International Conference on Robotics and Automation*, 3357–3364.