Physics-guided Information Acquisition and Learning For Urban Infrastructure Systems With Constrained Sensing Capabilities

Submitted in partial fulfillment of the requirements for

the degree of

Doctor of Philosophy

in

Civil and Environmental Engineering

Susu Xu

B.E., Environmental Engineering, Tsinghua University
M.S., Advanced Infrastructure Systems, Carnegie Mellon University
M.S., Machine Learning, Carnegie Mellon University

Carnegie Mellon University
Pittsburgh, PA

August, 2019

ACKNOWLEDGMENTS

Throughout my Ph.D. life, I have received a great deal of support and assistance. There are many people that I need to thank.

I would first like to thank my advisor, Professor Hae Young Noh. Thanks for the guidance, encouragement, and support throughout my Ph.D. studies. Thanks for teaching me critical thinking, for encouraging me to be open-minded, for helping me with my presentation and writing skills, for broadening my perspectives and inspiring me, for making me an independent researcher instead of being addicted to the "student mode". Thank you for organizing this great team. Thank you for being such a frank and great advisor.

A special thanks to Professor Pei Zhang. Thanks for continuously challenging me, pushing me and helping me, for bringing those exciting interdisciplinary collaboration opportunities, for all the valuable suggestions about both research and life.

Thanks to all the other committee members. Thank you Professor Chris Hendrickson for your valuable suggestions on my research and career path. Thank you Professor Xue (Steve) Liu for the insightful comments and suggestions on my thesis. I am especially grateful to Professor Carlee Joe-Wong, I learned a lot from our discussions and thank you for all the great advice. I would like to also thank Professor Susan Finger and Professor Sean Qian. I was really grateful for the support throughout my Ph.D. life.

Thank you Shijia Pan, Adeola Bannis, Mostafa Mirshekari and Jonathan Fagert for being great lab mates and for all the support during the past 5 years. Thank you Guannan He, Bingyin Hu, Wei Ma, Shuguan Yang, Pinchao Zhang, Shuai Ye, Weiguang Mao. Thanks for all of the wonderful times we have had together and I am glad that all of us survive this long Ph.D. journey.

Thank you Hongyang Zhang, Yaodong Yu, Jiantao Jiao, Pengtao Xie, Xinlei Chen, Jingxiao Liu, Weiwei Jiang, and Tong Yu, it was my pleasure to have such great collaborators like you. Thank you George Lederman, Irem Velibeyoglu, Peng Gong, Matineh Eybpoosh, Frank Mokaya, Carlos Ruiz, Amelie Bonde, Ankit Shrivastava, Shuxin Yao, Mike Lam, Biwei Huang for all the help and advise. Thanks Yan Li, Junmeng Hou, Liang Yu, Yunxiang Qiu, Yue Zhao, Cong Wang, Xulan Zhang for all the support.

Last but not least, I would like to thank my parents for their unconditional love, support, and understanding. Thanks my grandparents. I always miss the days to be with you and always remember what I was taught. Thanks Xidong, for always standing behind me. Thanks to all my families. This fantastic journey is impossible without all your support.

All the works presented in this thesis have been partially supported by NSF CMMI-1653550, NSF CNS-1149611, NSF CNS-1751075, DARPA D11AP00265, University Transportation Center (DTRT13-G-UTC26), United States Geological Survey, Intel, CMU CIT Dean's Fellowship, CMU Dowd Fellowship, and CMU Liang Ji-dian Graduate Fellowship.

ABSTRACT

Urban infrastructure monitoring is essential to ensure the safe and efficient functioning of urban services. Recently, a lot of advanced sensing systems have been developed to improve the management and maintenance of urban infrastructure systems. Timely and accurate information acquisition and learning for urban infrastructures, such as structural health, traffic conditions, surrounding air quality, etc., is the main goal of these urban sensing systems.

To learn the infrastructure information from these large set of sensing data, many conventional data-driven approaches have been introduced using statistical models or machine learning techniques. However, urban infrastructure monitoring systems often have constrained sensing capabilities due to improper deployment conditions, budget limits, or the complexity of physical infrastructure systems. The constrained sensing capabilities include 1) noisy data from complex physical systems, 2) lack of labeled data limiting the accuracy of data-driven models, 3) inefficient sensor deployment with low sensing coverage, 4) a lack of proper sensors to be deployed in the target infrastructures to collect the target information from the infrastructures. These constrained sensing capabilities reduce the quality of information acquisition, which may significantly degrade the performance of information learning using conventional data-driven methods.

To address these challenges, my research objective is to utilize physical knowledge to acquire and learn high-fidelity urban infrastructure information. The physical interactions between different physical components of infrastructure systems, between the physical components of infrastructures and sensors, and between the physical components of infrastructures and the ambient environment, are always governed by the physical properties of the urban infrastructure systems and thus subjected to consis-

tent physical principles. Such underlying physical knowledge provides an important aspect to better understand the infrastructure systems and thus improve information acquisition and learning under sensing constraints. Specifically, my research focuses on the following four topics:

a) To address the challenge of noisy data in complex physical systems, I introduce an information-theoretic approach to extract the changes of wave propagation patterns between different physical components of infrastructures.

b) To address the challenge of lack of labeled data, I propose a new knowledge transfer scheme across different infrastructures based on the physical understanding of how sensing data changes with different infrastructures' physical properties.

c) To address the challenge of inefficient deployment of sensors, I introduce an incentivizing algorithm to optimize the sensing distribution considering vehicle mobility and human mobility.

d) To address the challenge of lack of proper sensors, I introduce an indirect sensing method to monitor the target infrastructures using ambient infrastructure sensing systems.

The physical knowledge contains the understanding of the physical patterns that urban infrastructure systems subject to. These physical patterns are governed by universal and classical laws of physics. The physical patterns are validated to reflect the underlying physical processes happened inside urban infrastructure systems in previous studies, but difficult to be learned directly from the less informative data collected under constrained sensing capabilities. Under sensing capabilities constraints, the combination of physical knowledge helps to effectively acquire high-fidelity sensing data and learn information from the collected sensing data for urban infrastructure systems.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Smart Urban Infrastructure Systems

In the current world, urban infrastructure systems play a vital role in the operation of the city. Urban infrastructure systems refer to "the physical components of interrelated systems providing commodities and services essential to enable, sustain, or enhance societal living conditions in cities" [54], such as buildings and transportation systems. Nowadays, there are more than 2.9 billion people living in urban area. According to UN estimates, the urban populations will rise to 9.8 billion by 2050 [12]. The huge number of urban populations brings huge demand for living space, mobility, and other urban services. These urban services are often provided by various urban infrastructure systems, like residential buildings, roads, light rails, and etc. Without safe and efficient functioning of urban infrastructure systems, the growth of urban populations might pose a catastrophic risk to city management and the quality of human life.

Accurate and efficient monitoring of urban infrastructures is essential in the management and maintenance of urban infrastructure systems. For example, infrastructure health monitoring systems monitor deterioration of infrastructures to ensure the safety of the human living environment [48, 133, 204], and traffic monitoring systems provide traffic congestion information improve the efficiency of transportation system [6, 42]. Urban infrastructure monitoring provides governing agencies daily information about the infrastructures and the ambient environment that infrastructures exist in and interact with (e.g., air quality, traffic states) [73]. Meanwhile, when disasters or emergencies happen, these monitoring systems assist government and professional teams to better assess the safety and operation conditions of infrastructures, which is important to help reduce the human fatalities and economic cost [246].

Advanced sensing systems automate the processes of urban infrastructure monitoring and enable the development of smart urban infrastructure systems. Real-time data are acquired and fur-

ther analyzed by embedding sensing systems into infrastructures or mobile sensor carriers which infrastructure interacts with. From intelligent building structural health monitoring using vibrations sensors [203], to autonomous traffic monitoring using mobility sensors [36], there are immense improvements enabled by sensing technologies [5]. As pervasive sensing becomes more feasible and affordable recently, our knowledge about the urban infrastructure systems and related environment is enriched with previously uncharted real-time information.

The general framework of urban infrastructure sensing often includes 4 steps: sensor preparation, sensor deployment, data collection, and information learning from the collected data for final tasks. Figure 1.1 gives an overview of the sensing process. In this process, people first select proper sensors which can acquire information about the target infrastructures. The sensors might be mobile or static sensors, depending on the sensing tasks. This selection always involves a trade-off among the economic cost, deployment constraints, and sensing quality. Then sensors are deployed and installed on the target infrastructures or carriers in the target environments. After deployment, the sensor system begins to collect real-time sensing data. Finally, people extract and learn information from collected sensing data to infer the safety/efficiency/conditions of the target infrastructure systems.

An important goal of the infrastructure sensing systems is to acquire and learn high-fidelity real-time information from the complex infrastructure systems. Ideally, efficient information acquisition obtains high-quality desired data related to the target infrastructure systems. Based on a large set of acquired sensing data, there are many conventional pure-data drive approaches developed to learn the infrastructure information. For example, people develop structural damage classification models using features from vibration signals and ground-truth damage state as labels [164, 234]. These pure data-driven approaches often outperform physical models when there are large sets of high-fidelity training data. However, in real-world practices, it is often difficult to acquire the high-fidelity data due to constrained sensing capabilities introduced later. When the collected sensing data is limited or not informative, e.g. noisy, lack of labels, not sufficient, or even no data, the performance of pure data-driven methods is significantly reduced.

1.2 Constrained Sensing Capabilities in Smart Urban Infrastructure Systems

The sensing capabilities of the infrastructure sensing systems are often constrained due to deployment constraints, budget limit and the complexity of the infrastructure systems. In real-world practices, the scale, structure, and functionality of infrastructure systems vary a lot, from city-scale public transit systems, to high-rise residential buildings. The deployment and data collection of

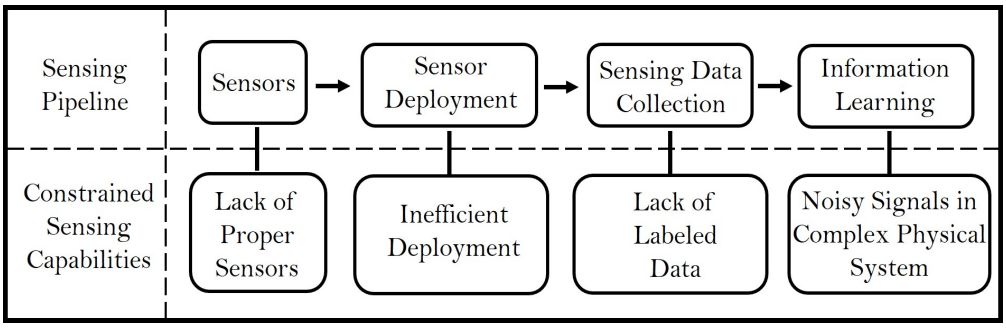


Figure 1.1: The general framework of urban infrastructure sensing and related sensing capability problems.

sensing systems are constrained by various physical environment, which causes different types of sensing capabilities constraints. These sensing capabilities constraints in different stages of the sensing procedure. When learning information from the collected data, the data may contain noise from complex physical systems. More severely, the collected data may lack of labels to indicate the ground truth (e.g. structural damage states and etc.), which makes it difficult to learn the infrastructure knowledge using pure data-driven method. Before data collection, the sensing capabilities constraints may appear in the up-streaming stages. The instrumentation and configuration of sensors may face the constraints of inefficient sensor deployment such that less informative data are collected. Moreover, lack of proper sensors to collect the information we need for urban infrastructure monitoring would make the data-driven methods no longer be able to work. As Figure 1.1 shows, we mainly consider these 4 main types of constrained sensing capabilities including 1) noisy data from complex physical systems, 2) lack of labeled data, 3) inefficient sensor deployment, and 4) lack of proper sensors. These 4 types of constrained sensing capabilities are common in real world urban infrastructure systems and difficult to tackle with the existing works [90, 136, 144, 231].

1.2.1 Noisy Data from Physical Systems

As the sensed data comes from the complex physical infrastructure systems, it is common that the data contains much noise due to the complicated physical processes in the infrastructure and the influence of uncertain environment. For example, in earthquake-induced building damage diagnosis, the building vibration signals are noisy due to the non-structural components, the fast-changing dynamics of earthquake excitation, and so on. The noise contained in infrastructure data is irregular and often tangled with physical environment, which makes it difficult to filter out the noise to extract high-fidelity information about the infrastructure systems.

1.2.2 Lack of Labeled Data

It is possible that the collected data may lack of ground truth as labels due to the deployment constraints, power supplies, and other issues. For example, data-driven approaches for building structural health monitoring have been developed, but it is often difficult to obtain the ground truth of the structural health state due to the high cost of sensors to measure the structural damages [90, 144], especially in the post-disaster scenarios. Without enough labeled data, the accuracy of data-driven approaches would be significantly reduced.

1.2.3 Inefficient Sensor Deployment

In urban infrastructure systems, the sensor nodes are deployed in the static or mobile manner, from city-scale deployment to deployment inside a single infrastructure such as a bridge or a building. Either way faces a problem of inefficient sensor deployment due to dynamically changing physical environment. Inefficient sensor deployment often results in low sensing coverage and increases the deployment cost significantly. For example, in mobile sensing systems, the mobility of sensor carriers poses the challenge to achieve optimal dynamic sensor deployment at each time point. If we want to use this mobile sensing system to collect information (e.g. air quality, traffic congestion) in different areas of the city, the sub-optimal sensor deployment would result in a lack of sufficient data in those areas with few sensor carriers. The inefficient spatio-temporal sensor deployment finally reduces the sensing quality and increases sensing cost.

1.2.4 Lack of Proper Sensors

Although sensing technologies become ubiquitous, it is common that people cannot find proper sensors to collect the desired infrastructure information or can only prepare a limited number of sensors for the target task because of the deployment constraints and cost. For example, dense sensor deployment to monitor the conditions of the light rail tracks is often costly, and the most common way to monitor the track health is manual inspection due to the lack of proper sensors.

The constrained sensing capabilities impair the quality of information acquisition and performance of information learning. Conventional pure data-driven approaches often have the assumptions on the collected dataset, such as that the collected data contains Gaussian noise or the collected data are labeled, which is no longer fulfilled under sensing capability constraints. The noisy sensing data may make the features extracted by statistical models ineffective for infrastructure information prediction or inference. The lack of labeled data violates the requirements of supervised/semi-supervised machine learning techniques on training labels. Inefficient sensor deployment causes a lack of important sensing information about the infrastructure systems. Without proper sensors, it would be difficult to obtain the data about the target infrastructure systems.

Therefore, under sensing capability constraints, pure data-driven approaches hardly work.

1.3 Physics-guided Sensing and Learning in Smart Urban Infrastructure Systems

A key characteristic of urban infrastructure systems is that as physical systems, they obey consistent physical rules. As shown in Figure 1.2, a general urban infrastructure system includes the infrastructures, sensors, and ambient environment. In this system, physical processes, such as structural responses, human occupants' movements, and air pollutants diffusion, happen in and/or around the infrastructures when the physical components of the infrastructures interact with each other, with sensors, or with the environment. We can see that, these physical processes are not isolated, and they are always involved with the physical components of the infrastructures and contain information about infrastructure conditions. Meanwhile, the physical processes inside different infrastructures share common physical principles. By combining the understanding of these physical processes, we can use the underlying physical knowledge to guide the best utilization of the constrained sensing capabilities and limited resources to acquire and learn more sufficient information.

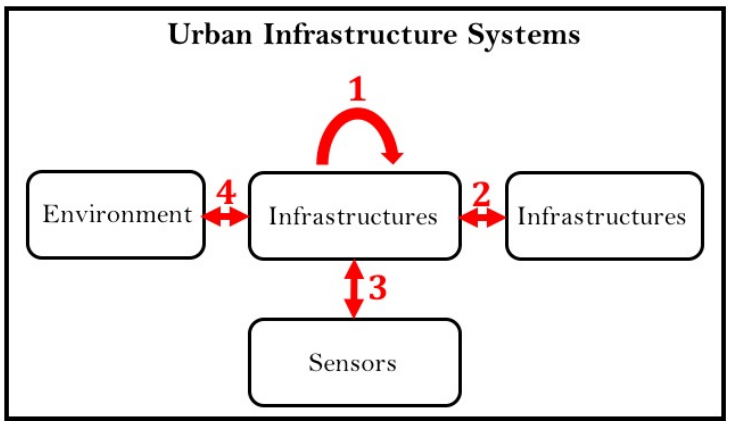


Figure 1.2: The overview of smart urban infrastructure systems. The infrastructure interacts with the sensors, the environment. It also shares similar physical functionality/knowledge with other infrastructures.

To address the aforementioned challenges induced by constrained sensing capabilities (Section 1.2), it is important to have a holistic understanding of the physical processes inside infrastructure systems, between different infrastructure systems, between the infrastructure with the sensors, and between the infrastructure and environment. The general philosophy is to systematically

combine the understanding of the physical processes with the data-driven methodology development. In this way, we can utilize the physical knowledge to complement and regularize the data-driven models of which performance are limited by constrained sensing capabilities. As Figure 1.3 shows, combining the understanding of the physical processes inside infrastructures, I model the wave propagation processes inside infrastructures using information theory, which allows the extraction of information-theoretic features from noisy collected data to infer the health states of infrastructures. By understanding the similarities and differences of the physical processes in different infrastructures, I transfer the implicit knowledge learned from the labeled data collected on other infrastructures to complement the information learning for the target infrastructures without labeled data. By discovering the physical mobility patterns of the sensor carriers, I present an efficient sensors deployment algorithm to collect the most informative data. By observing the physical interactions between the infrastructure and its environment, we can indirectly acquire information about the target infrastructures from the environment without direct instrumentation of sensors on the target infrastructures. Although the sensing capabilities are often constrained, the underlying physical knowledge provides a basic but essential aspect to better understand the infrastructure systems and thus improve the information acquisition and learning under sensing constraints.

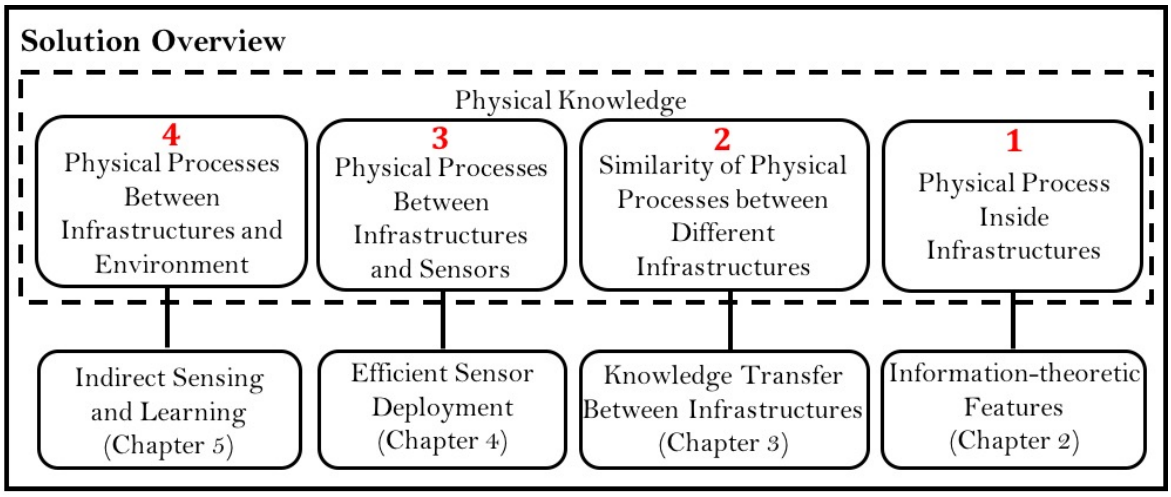


Figure 1.3: The solution to combine physical knowledge to address the challenges of constrained sensing capabilities.

1.4 Contributions

My research objective is to combine the classic physical knowledge with data-driven approaches to acquire and learn high-fidelity information about urban infrastructure systems with constrained sensing capabilities. Specifically, my research aims to integrate the physical understanding about

the generation, propagation and distortion of information to the process of urban infrastructure sensing, and thus sufficiently acquire, extract and learn knowledge about the target infrastructure systems under sensing capability constraints. My research has 4 main contributions:

1. For analyzing noisy sensing data from complex infrastructure systems, we model their physical properties using an information theoretic approach. This approach extracts the changes of the causal relationships between different components of the infrastructures to infer the infrastructures' physical conditions. For example, the model represents the changes in vibration waveform distortion with information exchanges to reflect the structural damages. With the physical understanding of the generation of structural damages, this method reduces the influence of environmental noise in infrastructure information learning.
2. For the infrastructure with incomplete datasets, such as missing ground truth and training samples for learning models, we transfer the knowledge learned from other infrastructures to improve the information learning for the target infrastructure. We combine the effects of physical properties of infrastructures on the sensing data distributions and adapt the models learned from different infrastructures to help monitor the infrastructure with incomplete/very limited datasets.
3. For efficient sensor deployment in mobile crowd sensing frameworks, we incentivize vehicle mobility to match the collected sensing data distribution to the desired sensing distribution. We propose an incentive mechanism considering human mobility and vehicle mobility to optimize sensing coverage while ensuring the utility of vehicle agents. A new optimization algorithm is also developed to efficiently obtain the incentive strategies.
4. For target infrastructures lacking proper sensors, we use indirect sensing through other sensing systems. For example, We use traffic-induced building structural vibrations to monitor nearby traffic events. We consider the influence of different types of traffic events on the building vibration patterns and propose an information-theoretic approach to extract information about traffic events using building vibrations.

1.5 Organization of the Thesis

This thesis proposal is organized as follows: In Chapter 2, we introduce an information-theoretic approach to analyze the noisy sensing data from complex infrastructure systems. We validate the proposed method in the context of post-disaster structural health monitoring. In Chapter 3, we present a new knowledge transfer framework to adapt the knowledge learned across different infrastructures to to address the challenge induced by incomplete sensing data. The proposed method embeds the fuzzy physical knowledge about the similarities between different infrastructure into a

data-driven adversarial knowledge transfer model. We plan to validate our algorithm in the context of post-disaster building damage diagnosis with incomplete vibration sensing datasets. In Chapter 4, we study the problem of efficient sensor deployment in urban mobile crowd sensing systems. We design a new incentive mechanism and optimization algorithm to match the distribution of collected sensing data to the desired target distribution. The proposed algorithm is validated using a real-world spatio-temporal air quality monitoring system in Beijing City. In Chapter 5, we show a new indirect sensing approach using ambient infrastructure sensing system to monitor the target infrastructure systems. The developed method is implemented and evaluated in the scenario of real-world traffic event monitoring using ambient building vibrations. The results show that indirect sensing from related infrastructures can provide useful information about the target infrastructures, which is a new perspective to improve information acquisition in the future.

Chapter 2

Information Exchange Analysis on Noisy Sensing Signals

2.1 Problem Overview

Accurate and timely building structural damage diagnosis is important to help save lives and reduce the reconstruction cost in post-earthquake scenarios. Damage diagnosis techniques can help identifying safe shelters to temporally move in, assessing the building safety conditions for evacuation, and determining to rebuild/repair/reserve buildings in an earthquake zone. For example, on the 2011 Tohoku earthquake, a lot of factories were damaged during earthquake [97]. A fast and accurate inspection of these buildings is critical to accelerate the recovery process of factory production.

Current practices of building structural damage diagnosis are mostly labour intensive, time consuming, or error prone. For example, in the Tohoku earthquake, it took many experts more than 1 year to get a full statistics on the overall building damage through visual inspection [97, 157]. Given the drawbacks of current post-earthquake reconnaissance practice, new sensor-based techniques have been actively explored to automate the earthquake-induced building structural damage diagnosis [135, 205].

Recently, people developed vibration-based structural damage diagnosis methods. Based on automatically collected building vibrations during an earthquake, these methods provide the accuracy and speed needed to quickly evaluate the structural health of a building [37, 48, 85, 134]. Most of the vibration-based methods fall into two categories: physics-based methods [24, 52, 196, 197] and data-driven methods [92, 164, 234]. However, the physics-based methods require much prior knowledge about the building structure (e.g. building geometry and material properties), which is difficult to be obtained in the post-earthquake scenario. The data-driven methods utilize statistical

models to learn a mapping from the collected vibrations to the structural damage states based on the collected vibration data. But in post-earthquake scenario, the collected building vibration data contains complex environmental noises introduced by the fast-changing seismic dynamics. Moreover, conventional data-driven methods require dense sensor instrumentation for detailed and sufficient information to detect the structural damages separately [111, 122], which is labor-intensive and expensive.

To address these challenges, we introduce an information-theoretic approach to detect and quantify the earthquake-induced building structural damage with sparsely deployed vibration sensors and few prior knowledge about buildings. Our method is based on the premises that wave propagation inside structures can be modeled as the process of information exchanges between adjacent locations, and the structural damage will alter information exchange patterns between two locations. By detecting this change, our method detects and quantifies the damage state of each story inside the building. In this chapter, we collect vibration signals at each floor to detect story-level damage states, but the method is generally applicable to any spatial granularity. We then extract the information exchanges between the two vibration signals of the floor and the ceiling of each story based on the principles of information theory. With the information exchanges as features of each story, we estimated the damage state using machine learning techniques. Instead of detecting damages of each location, the presented method detects the damages between sensor pairs, which allows sparsely deployed sensors to infer the structural damages. This method does not require prior knowledge about building. Besides, the bi-directional information exchanges between two collected vibration signals are extracted to provide higher-resolution information about structural properties than conventional correlation-based features [164]. Moreover, we show the analytical relationship between information exchanges and the structural damage of each story to demonstrate that the information exchange is an effective indicator of the structural damage with physical significance.

This work has 3 key contributions:

1. To best of our knowledge, we are the first to model the wave propagation inside the building as information exchanges as defined in information theory, which allows the analysis of groups of noisy vibration data and provides more detailed information about the structural changes.
2. We present the physical insights of the data-driven information-theoretic approach and the analytical relationship between the information exchanges and the structural properties, which gives theoretical supports to using information exchanges to detect and quantify the structural damage state without prior knowledge of the structure.
3. We evaluate our algorithm using both simulation and experimental data of multiple buildings

with varying heights subjected to a series of multiple earthquake excitations. As a result, our approach achieves upto 15.49% improvement in the damage prediction accuracy.

In this chapter, we first discuss the related work in Section 2.2, and provide the physical insight of representing the wave propagation process between adjacent floors as information exchanges in Section 2.3. In Section 2.4, we present the analytical relationships between the story-level structural properties and the extracted information exchanges. Then the algorithm of extracting information exchanges as features to detect and quantify the structural damage state is introduced in Section 2.5. In Section 2.6, we evaluate our algorithm with data from multiple buildings under a series of earthquake excitations. Finally, Section 2.7 concludes the chapter.

2.2 Relate Work on Information Learning for Infrastructure Health Monitoring

Feature extraction based on sensing data is one of the most important parts in Infrastructure Health Monitoring. It is also challenging due to the complex physical process of structural response to various dynamic loading. There are various feature extraction methods used in vibration-based infrastructure health monitoring research.

Most conventional methods are focusing on finding changes in structural mode shapes by using, for example, Hilbert-Huang Transform. [79] firstly introduced Hilbert-Huang transform (HHT) to analyze signals. HHT generally includes two steps: empirical mode decomposition and then Hilbert spectral transform. [128, 186] applied empirical mode decomposition and HHT for infrastructure damage detection. The ensemble empirical mode decomposition is introduced by [221] to partially address the problem of mode mixing induced by intermittent frequencies in the empirical mode decomposition method.

In recent years, people also developed various features which are believed to indicate the structural damage. Wavelet Transform becomes a hot spot in recent years since it can provide information in time-frequency domain. Wavelet Transform has been widely applied in various fields like image processing, robotics and etc. In the field of infrastructure health monitoring, there are many features developed based on Wavelet Transform coefficients as indicator of structural damages [71, 77, 192]. Fourier Transform is another widely used feature extraction methods. By extracting modal parameters, Fourier Transform has been applied in various types of infrastructures, including buildings, bridges and pipelines [33, 117, 237]. The drawback of Fourier Transform is mainly on monitoring real structures under dynamic excitation since the collected signals are highly non-linear and non-stationary. Statistical time-series models are also used to learn the mathematical corresponding relationships between input signals and output. Many popular time-

series models are applied for infrastructure monitoring, e.g. auto-regressive model, auto-regressive moving average model, and auto-regressive integrated moving average model [23, 159]. Statistical time-series models need to firstly identify the model orders and then estimate the coefficients of models, which might be computationally expensive.

In our work, we proposed directed information as feature to quantify the causal relationship between two stochastic processes (i.e., a sequence of random variables). Directed information is developed based on the concept of entropy and mutual information in the field of information theory [37, 47, 89, 94, 95, 142, 200]. In general, entropy quantifies the uncertainty (lack of information) of a random variable. As an example, let random variables X and Y represent the number obtained by tossing a 4-side and 8-side dies, respectively. The entropy of Y will be higher than X, since Y has lower predictability, which means Y has higher uncertainty. When there is dependency relationship between two random variables, the two random variables share part of uncertainties induced by the dependency relationship. Mutual information can quantify this shared uncertainties [215]. This shared information can be computed by looking at the information gain (or reduction in uncertainties) for one variable by knowing another related random variable. As a special dependent relationship, causal relationship represents that one variable or process is part of the reason (source of information) of the other variable or process. When there is causal relationship between two random variables, directed information is an asymmetric measure that quantifies the shared information with directionality [94, 95]. For example, the information shared from one random variable to the other random variable.

2.3 Physical Insights of Information Propagation

This section provides the physical insights of representing the seismic wave propagation as a process of information exchanges between floors inside a building. When the vibration wave propagates from the floor to the ceiling of one story, the waveform is distorted due to energy dissipation inside the story [210]. This wave distortion can be represented by information exchange. The changes of information exchange reflect the changes of energy dissipation, which depends on the changes of the structural properties. Therefore, by extracting this information, the altering of structural properties can be detected. We can further detect and quantify the structural damages inside the building.

The process of wave propagation inside a building structural system is similar to the process of information exchanges in communication systems [107, 142, 236]. As Figure 2.1 shows, when earthquake happens, the seismic wave propagates from one location i to the adjacent location j through the structure between the two locations. In this process, there is noise from the non-

structural components or other sources which interfere the wave propagation. Similarly, in the communication system, the information is encoded by a transmitter, and sent from the transmitter to the receiver through the channel. In this communication process, the signal may be distorted by the noise when passing through the channel. In structural systems, the structure between location i and j corresponds to the communication channel. When the structure between the locations i and j is damaged, the damage changes the distortion of wave propagation. That is, when the structural damage happens, the information received at j, which is also the wave observed at j will be different from before damage. The change of the information exchange pattern from i to j indicates the structural damage. In the field of communication system, people developed information theory to study the information exchanges process [107, 142]. Here we can model and analyze the wave propagation process in the structural system using information theory. The information exchanges are bi-directional between i and j. When structural property changes, the information exchanges in two directions changes. The changes are different in different directions, which is discussed in Section 2.4. Therefore, we introduced directed information to quantify the directional information exchanges between two structural response signals.

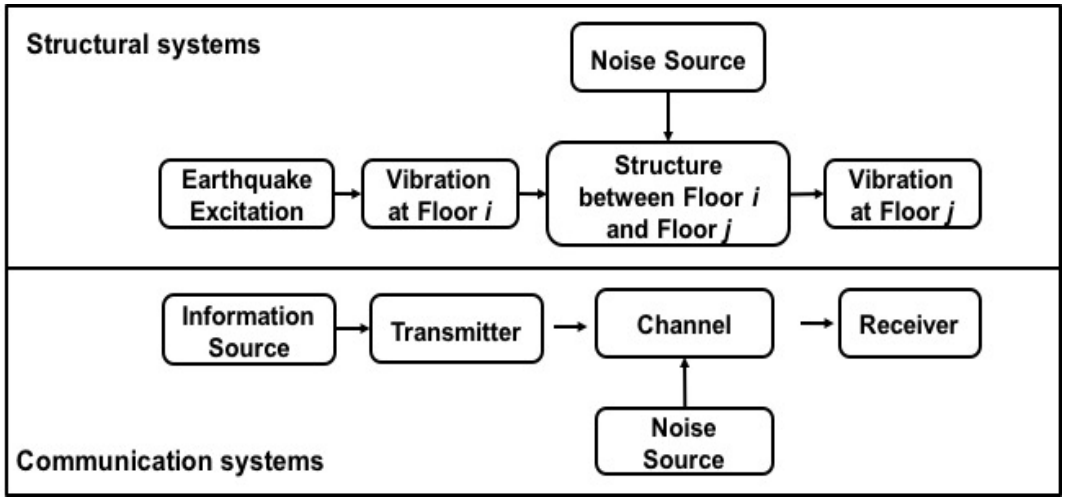


Figure 2.1: The analogy between wave propagation inside the structural system and information exchanges in the communication system.

When earthquake happens, the seismic wave propagates inside the building. In each story, the seismic waves propagating through the building could be separated into two components: up-going and down-going components. We consider a conceptual model as shown in Figure 2.2. Whenever the up-going and down-going waves cross a floor interface, they are partly reflected and partly transmitted into the next floor [210]. The transmitted wave would be attenuated along with the propagation path with multiple times of transmissions and reflections. The reflected component

would be partly reflected back and partly transmitted in the lower floor and finally attenuate as well. Therefore, the waves observed in the floor and the ceiling are different from each other because the wave is distorted when passing through the structures between them. The wave distortion depends on the properties of the structure that the seismic wave passes by. Meanwhile, the wave distortion between two locations can be represented by information exchanges between two locations' vibrations. Therefore, the structural information is embedded in the information exchanges between the two sensing locations.

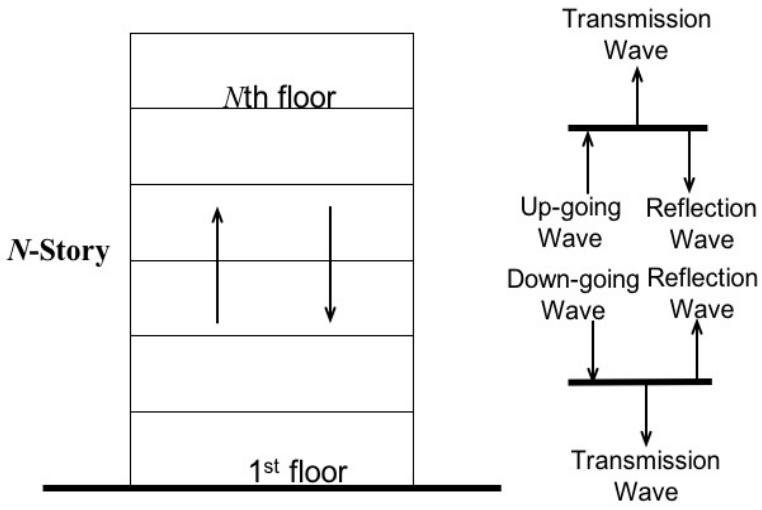


Figure 2.2: Seismic wave propagation inside the building structure. (a) In a N-story building, the seismic propagate inside the building by pass from one story to the next. (b) The seismic wave propagated inside the building can be decomposed into two parts: up-going component and down-going component. At each interface of floor, the propagated wave will be transmit and reflected. The ratio of transmission and reflection is related to the structure of the floor.

When there is earthquake-induced structural damage in some story, the structural properties of the story change, which also change the information exchanges pattern between the ceiling and the floor vibrations. Since the information about the structure is contained in the wave distortion, the wave distortion between the floor and the ceiling change with different structural damages. For example, given a story, suppose there is a crack appearing in one of the columns during the earthquake. When the seismic wave passes through the column, the energy dissipation becomes different from that in a well-conditioned column due to the crack. Compared to when there is no damage in the column, the wave distortion between the ceiling and the floor carries changes, thus, the information exchange between two the ceiling and the floor vibrations also changes.

By extracting the changes of the information exchanges inside each story, we can detect the

changes of the structural properties of the story, and therefore detect and quantify the structural damage state.

2.4 Analytical Interpretation of the Relationship between Directed Information and Structural Parameters

In this section, we discuss the analytical relationship between the information exchanges and the structural properties. We first introduce the concept of directed information to quantify the information exchanges between two vibration signals. For each story, the directed information is extracted from the vibrations of the floor and the ceiling of the story. We present the physical relationship between the extracted directed information and structural properties of the corresponding story.

2.4.1 Directed Information

The change of information exchange patterns between the vibration signals at two floors indicates the change of structural properties between the two floors, as Section 2.3 discussed. Here we use directed information from the field of information theory to quantify the information exchanges between two vibration signals. The information theory is developed to model the information (or uncertainties) contained in random variables (or processes), e.g. seismic wave-induced floor vibrations [242]. Directed information is a concept developed in information theory to measure the directional shared information between two signals.

Directed information is first introduced based on the concept of entropy and mutual information in the field of information theory [89, 142]. In general, entropy quantifies the uncertainty (lack of information) of a random variable. As an example, let random variables X and Y represent the number obtained by tossing a 4-side and 8-side dies, respectively. The entropy of Y will be higher than X, since Y has lower predictability, which means Y has higher uncertainty. When there is dependency relationship between two random variables, the two random variables share part of uncertainties induced by the dependency relationship [37, 47, 200]. Mutual information quantifies this shared uncertainties. This shared information is computed as the information gain (or reduction in uncertainties) for one variable by knowing another related random variable and vice versa [16, 108]. As a natural counterpart, directed information depicts the causal influence that one variable or process (source of information) has on the other variable or process [94, 95, 215]. When there is causal influence between two random variables, directed information is an asymmetric measure that quantifies the shared information with directionality, for example, the

information shared from one random variable to the other random variable. Therefore, compared to the conventional correlation-based features merely focusing on the co-occurrence of two random variables' statistical characteristics, directed information is a more precise measurement providing high-resolution information between two random variables/processes. The concept of directed information has been widely applied in different fields, including identify the pairwise influence in gene networks [195], neuroscience [218], and etc.

In our problem, we define two stochastic processes $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ to represent building vibrations at two different floors i and j from the time point t_1 to the time point t_2, respectively. We define the directed information between them using their joint probability density function (PDF). If $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ are independent, their joint distribution is equivalent to the product of their marginal distributions,

$$P(X^i_{t_1:t_2}; X^j_{t_1:t_2}) = P(X^i_{t_1:t_2})P(X^j_{t_1:t_2}). \tag{2.1}$$

Then, the mutual information of $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ ($I(X^i_{t_1:t_2}; X^j_{t_1:t_2})$) is quantified as the distance (or information discrepancy) between the joint PDF and the product of the marginals by using the concept of Kullback-Leibler divergence (i.e., the mutual information measures the degree of dependency). The distance here represents the information gain when we revise our belief from that $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ are independent to that $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ are dependent:

$$I(X^i_{t_1:t_2}; X^j_{t_1:t_2}) = E[\log \frac{P(X^i_{t_1:t_2}; X^j_{t_1:t_2})}{P(X^i_{t_1:t_2})P(X^j_{t_1:t_2})}]. \tag{2.2}$$

Mutual information is always non-negative, and it becomes zero when $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ are independent. This mutual information does not represent any directionality in information flow. Hence, an alternative factorization in terms of the joint PDF has been introduced to represent the directionality of information feedforward and feedback between $X^i_{t_1:t_2}$ and $X^j_{t_1:t_2}$ [142]

$$P(X^i_{t_1:t_2}; X^j_{t_1:t_2}) = \overleftarrow{P}(X^i_{t_1:t_2}|X^j_{t_1:t_2})\overrightarrow{P}(X^j_{t_1:t_2}|X^i_{t_1:t_2}), \tag{2.3}$$

where $\overleftarrow{P}(X^i_{t_1:t_2}|X^j_{t_1:t_2}) = \prod_{t=t_1}^{t_2} P(X^i_{t+1}|X^i_{t_1:t}; X^j_{t_1:t+1})$ and
$\overrightarrow{P}(X^j_{t_1:t_2}|X^i_{t_1:t_2}) = \prod_{t=t_1}^{t_2} P(X^j_{t+1}|X^j_{t_1:t}; X^i_{t_1:t+1})$. If we consider $X^i_{t_1:t_2}$ as an input and $X^j_{t_1:t_2}$ as an output, $\overleftarrow{P}(X^i_{t_1:t_2}|X^j_{t_1:t_2})$ and $\overrightarrow{P}(X^j_{t_1:t_2}|X^i_{t_1:t_2})$ correspond to information feedback and feedforward, respectively.

Similar to the definition of the mutual information where we compare the true joint PDF to the PDF computed as if the processes are partial dependent, the directed information from $X^i_{t_1:t_2}$ to $X^j_{t_1:t_2}$ is defined as the distribution divergence between the true joint distribution and the distribution computed as if $X^i_{t_1:t_2}$ depends on $X^j_{t_1:t_2}$ but not vice versa. When $X^j_{t_1:t_2}$ does not depend on

$X^i_{t_1:t_2}$, $\overrightarrow{P}(X^j_{t_1:t_2}|X^i_{t_1:t_2}) = P(X^j_{t_1:t_2})$. Thus, the directed information is defined as:

$$I(X^i_{t_1:t_2} \to X^j_{t_1:t_2}) = E[log\frac{P(X^i_{t_1:t_2}, X^j_{t_1:t_2})}{\overleftarrow{P}(X^i_{t_1:t_2}|X^j_{t_1:t_2})P(X^j_{t_1:t_2})}]. \tag{2.4}$$

The directed information is smaller than or equivalent to the mutual information. Note that $I(X^i_{t_1:t_2} \to X^j_{t_1:t_2}) \neq I(X^j_{t_1:t_2} \to X^i_{t_1:t_2})$. By the definition of entropy and conditional entropy, the directed information is expressed as follows:

$$I(X^i_{t_1:t_2} \to X^j_{t_1:t_2}) = H(X^j_{t_1:t_2}) - H(X^j_{t_1:t_2}||X^i_{t_1:t_2}), \tag{2.5}$$

where

$$H(X^j_{t_1:t_2}) = \sum_{t=t_1}^{t_2} H(X^j_{t+1}|X^j_{t_1:t})$$
$$H(X^j_{t_1:t_2}||X^i_{t_1:t_2}) = \sum_{t=t_1}^{t_2} H(X^j_{t+1}|X^j_{t_1:t}, X^i_{t_1:t+1}).$$

The entropy $H(X^j_{t_1:t_2})$ and $H(X^j_{t_1:t_2}||X^i_{t_1:t_2})$ are functionals of the discrete distribution of variables $X^j_{t_1:t_2}$ and $X^j_{t+1}|X^j_{t_1:t}, X^i_{t_1:t+1}$ for $t \in \{t_1, \cdots, t_2\}$. When estimating directed information, we use Equation 2.5 for computational efficiency, instead of Equation 2.4 that involves estimating the joint distribution. The entropy values are estimated using the minimax rate-optimal estimators under $l2$ loss. The minimax estimator minimizes the maximum loss function between estimator and functional of real distribution. The loss function is $l2$ norm of difference between estimator and functional of real distribution. We use empirical D-tuple joint distribution based on the collected data to estimate the functionals of real distribution, and it has been proved that empirical joint distribution of D-tuple converges to the true joint distribution. The estimator converges faster and has less mean square error than conventional MLE (Maximum Likelihood Estimator) [94, 95].

2.4.2 Relationship between Directed Information and Structural Properties

In this section, We show the analytical relationship between the physical properties of building structure and the directed information at each story. The results indicate that directed information extracted between adjacent floor accelerations reflects the structural properties (e.g. stiffness, damping and mass) of the two adjacent two floors, and is a potential damage indicator.

Some assumptions are made to simplify the problem and highlight the important characteristics of the relationship between directed information and structural physical properties. We model the

building as a linear multi-degree of freedom system as shown in Figure 2.3. In the building, the mass is concentrated at each floor. The stiffness of the building is determined by the massless walls and columns.

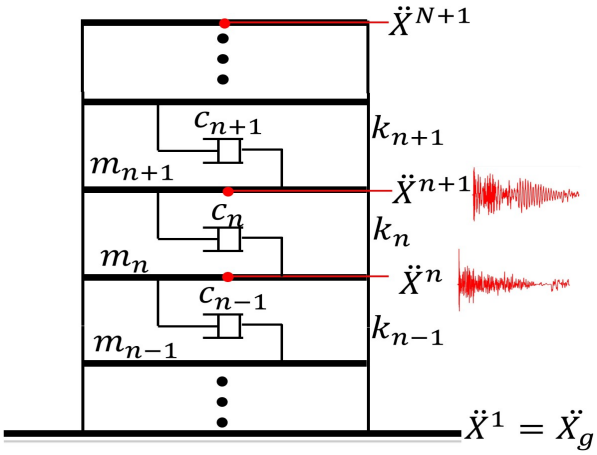


Figure 2.3: Structural model of a building under the earthquake excitation $\ddot{X}_g$.

Given an N-story building, each story is composed of a floor and a ceiling. We collect the earthquake-induced acceleration signal at each floor. We denote the collected acceleration at the nth floor as $\ddot{X}^n$. Similarly, we denote the velocity and relative displacement at the nth floor as $\dot{X}^n$ and X^n, respectively. Given a story n in the building, where $1 \leq n \leq N$, the acceleration at the ceiling of the nth story is also the acceleration at the floor of the $(n+1)$th story. We assume the floor acceleration at the base is the same as the ground motion acceleration, i.e. $\ddot{X}^1 = \ddot{X}_g$, where $\ddot{X}_g$ is the earthquake-induced ground motion acceleration. We denote the mass, shear stiffness, and damping coefficients for the nth story as m_n, k_n and c_n, respectively, as shown in Figure 2.3.

Denote the mass, stiffness and damping matrices of the building as $\mathbf{M}$, $\mathbf{C}$, and $\mathbf{K}$, and use $\mathbf{X}$ to represent the displacement matrix, i.e. $\mathbf{X} = [X^1, X^2, \cdots, X^N]^T$. The equation of motion is then

$$\mathbf{M}\ddot{\mathbf{X}} + \mathbf{C}\dot{\mathbf{X}} + \mathbf{K}\mathbf{X} = -\mathbf{M}I\ddot{X}_g, \tag{2.6}$$

where $\mathbf{I}$ is a vector with all elements as 1. The details of the physical properties matrices are

$$\mathbf{M} = \begin{bmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & m_N \end{bmatrix}, \mathbf{C} = \begin{bmatrix} c_1 + c_2 & -c_2 & 0 & \cdots & 0 \\ -c_2 & c_2 + c_3 & -c_3 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & -c_{N-1} & c_{N-1} + c_N & -c_N \\ 0 & \cdots & \cdots & -c_N & c_N \end{bmatrix},$$

$$\mathbf{K} = \begin{bmatrix} k_1 + k_2 & -k_2 & 0 & \cdots & 0 \\ -k_2 & k_2 + k_3 & -k_3 & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & -k_{N-1} & k_{N-1} + k_N & -k_N \\ 0 & \cdots & \cdots & -k_N & k_N \end{bmatrix}$$

Assume that there exists a zero-mean Gaussian noise V with positive definite covariance matrix Q for floor vibrations. Assume the noises for different stories are independent, i.e. $Q_{2n_1,2n_2} = Q_{2n_1+1,2n_2} = Q_{2n_1+1,2n_2+1} = Q_{2n_1,2n_2+1} = 0, \forall n_1 \neq n_2$ ($Q_{i,j}$ refers to the element in the ith row and jth column of Q). Denote $Z = \left[X^1, \dot{X}^1, \cdots, X^N, \dot{X}^N\right]^T$ as a multivariate variable, we can transform the Equation 2.6 into

$$\dot{Z} = AZ - MI^* \ddot{X}_g + V, \tag{2.7}$$

where $I^* = [0, 1, \cdots, 0, 1]^T$ has dimension of $2N \times 1$, M has the size of $2N \times 2N$ where $M(2n-1, 2n-1) = M(2n, 2n) = m_n$ ($M(i, j)$ refers to the element in the ith row and jth column of M), and A is the coefficient matrix with size of $2N \times 2N$ where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & \cdots & 0 \\ -\dfrac{k_1 + k_2}{m_1} & -\dfrac{c_1 + c_2}{m_1} & \dfrac{k_2}{m_1} & \dfrac{c_2}{m_1} & 0 & 0 & \cdots & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & \cdots & 0 \\ \dfrac{k_2}{m_2} & \dfrac{c_2}{m_2} & -\dfrac{k_2 + k_3}{m_2} & \dfrac{c_2 + c_3}{m_2} & \dfrac{k_3}{m_3} & \dfrac{c_3}{m_3} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & \ddots & 0 & 1 & 0 & 0 \\ 0 & \cdots & \dfrac{k_{N-1}}{m_{N-1}} & \dfrac{c_{N-1}}{m_{N-1}} & -\dfrac{k_{N-1} + k_N}{m_{N-1}} & -\dfrac{c_{N-1} + c_N}{m_{N-1}} & \dfrac{k_N}{m_{N-1}} & \dfrac{c_N}{m_{N-1}} \\ 0 & \cdots & \cdots & \cdots & 0 & 0 & 0 & 1 \\ 0 & \cdots & \cdots & \cdots & \dfrac{k_N}{m_N} & \dfrac{c_N}{m_N} & -\dfrac{k_N}{m_N} & -\dfrac{c_N}{m_N} \end{bmatrix}.$$

Discretizing the Equation 2.7, assuming zero-order hold for the ground motion $\ddot{X}_g$, gives

$$Z_{t+1} = A_d Z_t + M_d I^* \ddot{X}_{g,t} + V_d, \tag{2.8}$$

where each time point represents a sample time of Δt, $V_d \sim N(\mathbf{0}, Q_d)$, and

$$A_d = \exp(A\Delta t);\ M_d = -\Big(\int_{\tau=0}^{\tau=\Delta t} \exp(A\tau)d\tau\Big)M;\ Q_d = \int_{\tau=0}^{\tau=\Delta t} \exp(A\tau)Q\exp(A^T\tau)d\tau.$$

For simplicity, we denote the structural responses at the nth floor at the time point of $t+1$ as $\mathbf{X}_{t+1}^n = [X_{t+1}^n, \dot{X}_{t+1}^n]^T$. Equation 2.8 shows that the structural responses depend on the structural responses of adjacent stories at the previous time point t, i.e., ($\Longleftrightarrow$ refers to the dependency relationship)

$$\cdots \Longleftrightarrow \mathbf{X}_{t:t+1}^{n-1} \Longleftrightarrow \mathbf{X}_{t:t+1}^{n} \Longleftrightarrow \mathbf{X}_{t:t+1}^{n+1} \Longleftrightarrow \cdots . \tag{2.9}$$

With the dependency relationship described in 2.9, the dependencies between the structural responses can be described as

$$\mathbf{X}_{1:t}^{n-1} \perp\!\!\!\perp \mathbf{X}_{1:t}^{i} | X_{1:t}^{n}, \forall i > n. \tag{2.10}$$

2.10 represents that given the vibration at the nth floor, the vibration at the lower floor ($< n$) is independent with the vibration at the higher floor ($> n$). Therefore, we have the lemma describing the directed information from $(n+1)$th floor to nth floor at the time point of t is independent of other stories' information:

Lemma 1. *The directed information at time point of t from $(n+1)$ floor to n floor is independent with the information from other nonadjacent floors, i.e.*

$$\begin{aligned}
&H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n) - H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1}) \\
&= H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) - H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}).
\end{aligned}$$

Proof. Based on the Bayes rule, we can expand the conditional entropy as

$$\begin{aligned}
&H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) - H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) \\
&= H(\mathbf{X}_{t+1}^n, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t}^n) - H(\mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t}^n) \\
&\quad - [H(\mathbf{X}_{t+1}^n, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1}) - H(\mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1})] \\
&= H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n) + H(\mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t+1}^n) + H(\mathbf{X}_{1:t}^{n+2} | \mathbf{X}_{1:t+1}^n) - H(\mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t}^n) - H(\mathbf{X}_{1:t}^{n+2} | \mathbf{X}_{1:t}^n) \\
&- [H(\mathbf{X}_{t+1}^n | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1}) + H(\mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t+1}^n, \mathbf{X}_{1:t+1}^{n+1}) + H(\mathbf{X}_{1:t}^{n+2} | \mathbf{X}_{1:t+1}^n, \mathbf{X}_{1:t}^{n+1}) \\
&\quad - H(\mathbf{X}_{1:t}^{n-1} | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1}) - H(\mathbf{X}_{1:t}^{n+2} | \mathbf{X}_{1:t}^n, \mathbf{X}_{1:t+1}^{n+1})]
\end{aligned}$$

From the independence relationship in Equation 2.10, there is $H(\mathbf{X}_{1:t}^{n+2}|\mathbf{X}_{1:t}^{n}) - H(\mathbf{X}_{1:t}^{n+2}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t+1}^{n+1})$ $= H(\mathbf{X}_{1:t}^{n+2}|\mathbf{X}_{1:t+1}^{n}) - H(\mathbf{X}_{1:t}^{n+2}|\mathbf{X}_{1:t+1}^{n}, \mathbf{X}_{1:t+1}^{n+1})$ and $H(\mathbf{X}_{1:t}^{n-1}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t+1}^{n+1}) = H(\mathbf{X}_{1:t}^{n-1}|\mathbf{X}_{1:t}^{n})$. Therefore, we can remove the terms with $\mathbf{X}_{1:t}^{n+2}$ and $\mathbf{X}_{1:t}^{n-1}$ to obtain

$$\begin{aligned}
&H(\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) - H(\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+1}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) \\
&= H(\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}) - H(\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+1}).
\end{aligned}$$

□

With Lemma 1, we can obtain the directed information from $(n+1)$th floor to nth floor using structural properties and the white noises. Given $\ddot{X}_{g,1:t}$, $V_d \sim N(\mathbf{0}, Q_d)$, and the starting states $\mathbf{X}_0^n, \forall n$, since linear transform of a Gaussian variable is still Gaussian variable, $\mathbf{X}_t^{n+1}$ subjects to a Gaussian distribution. Denote the $(2n-1)$th and $2n$th rows of the matrix A_d as $A_d(2n-1:2n,\cdot) = [A_d^1, \cdots, A_d^N]$, where A_d^n has size of 2×2. Similarly, denote $M_d(2n-1:2n,\cdot) = [M_d^1, \cdots, M_d^N]$ and $I^*(2n-1:2n,\cdot) = [I_n^*, \cdots, I_N^*]$. Therefore, the conditional variables are expressed as

$$\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1} = A_d^n \mathbf{X}_t^n + A_d^{n-1}\mathbf{X}_t^{n-1} + (A_d^{n+1}\mathbf{X}_t^{n+1}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+2}) + M_d^n I_n^* \ddot{X}_g + V_d^n \tag{2.11}$$

$$= A_d^n \mathbf{X}_t^n + A_d^{n-1}\mathbf{X}_t^{n-1} + \left(A_d^{n+1}\right)^t \mathbf{X}_0^{n+1} + f_1(A_d^n, A_d^{n+1}, A_d^{n+2}, \mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+2}) \tag{2.12}$$

$$+ \sum_{j=1}^{t-1} \left(A_d^{n+1}\right)^j V_d^{n+1} + M_d^n I_n^* \ddot{X}_g + V_d^n, \tag{2.13}$$

$$\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t+1}^{n+1}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1} = A_d^n \mathbf{X}_t^n + A_d^{n-1}\mathbf{X}_t^{n-1} + A_d^{n+1}\mathbf{X}_t^{n+1} + M_d^n I_n^* \ddot{X}_g + V_d^n. \tag{2.14}$$

where f_1 is an implicit function involving with the influence of the structural vibrations at nth and $n+2$th floor. A_d^n can be approximated as follows by Euler's method:

$$A_d^n \approx \begin{bmatrix} 1 & \Delta t \\ -\dfrac{k_n + k_{n+1}}{m_n}\Delta t & 1 - \dfrac{c_n + c_{n+1}}{m_n} \end{bmatrix}^T.$$

The variance matrices for the two conditional distributions are

$$Var(\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) = Var\left[\sum_{j=1}^{t-1} \left(A_d^{n+1}\right)^j V_d^{n+1}\right] + Var(V_d^n) \tag{2.15}$$

$$Var(\mathbf{X}_{t+1}^{n}|\mathbf{X}_{1:t}^{n}, \mathbf{X}_{1:t+1}^{n+1}, \mathbf{X}_{1:t}^{n+2}, \mathbf{X}_{1:t}^{n-1}) = Var(V_d^n) = Q_d^n, \tag{2.16}$$

where Q_d^n refers to the covariance matrix for variable $\mathbf{X}^n = [X_t^n, \dot{X}_t^n]$, which is obtained from Equation 2.8. Since it is assumed that the process noise for different floor responses are independent, we can obtain

$$Q_d^n = \int_{\tau=0}^{\tau=\Delta t} \sum_{j=n-1}^{n+1} A_d^{j,n} Q^j (A_d^{j,n})^T d\tau = \int_{\tau=0}^{\tau=\Delta t} A_d^{n,n} Q^n (A_d^{n,n})^T d\tau + \mathcal{O}(\Delta t^3),$$

where

$$A_d^{j,n} = \begin{bmatrix} A_d(2n-1, 2j-1) & A_d(2n-1, 2j-1) \\ A_d(2n, 2j) & A_d(2n, 2j) \end{bmatrix}, Q^n = \begin{bmatrix} Cov(\dot{X}^n, \dot{X}^n) & Cov(\ddot{X}^n, \dot{X}^n) \\ Cov(\dot{X}^n, \ddot{X}^n) & Cov(\ddot{X}^n, \ddot{X}^n) \end{bmatrix}.$$

Let $P_t^{n+1} = Var\left[\sum_{j=1}^{t-1} \left(A_d^{n+1}\right)^j V_d^{n+1}\right]$. P_t^{n+1} depends and only depends on the structural properties of the floor and the ceiling of the $(n+1)$th story and the Gaussian noise on the structural responses of the ceiling of the nth story.

The entropy of multivariate Gaussian distribution with variance matrix of Q is

$$\frac{1}{2} ln \det(2\pi e Q). \tag{2.17}$$

Given the definition of directed information in Equation 2.5, we have the directed information from $(n+1)$th floor to nth floor as

$$I(X_{1:T}^{n+1} \to X_{1:T}^n) = \sum_{t=1}^{T} H(X_{t+1}^n | X_{1:t}^n) - H(X_{t+1}^n | X_{1:t}^n, X_{1:t}^{n+1}) \tag{2.18}$$

$$= \sum_{t=1}^{T} \frac{1}{2} ln \frac{\det(P_t^{n+1} + Q_d^n)}{\det Q_d^n}. \tag{2.19}$$

Similarly, we obtain the inverse directed information from nth floor to $(n+1)$th floor as

$$I(X_{1:T}^n \to X_{1:T}^{n+1}) = \sum_{t=1}^{T} H(X_{t+1}^{n+1} | X_{1:t}^{n+1}) - H(X_{t+1}^{n+1} | X_{1:t}^{n+1}, X_{1:t}^n) \tag{2.20}$$

$$= \sum_{t=1}^{T} \frac{1}{2} ln \frac{\det(P_t^n + Q_d^{n+1})}{\det Q_d^{n+1}}. \tag{2.21}$$

By the definition of P_t^n, it can be found that the directed information from $(n+1)$th floor to nth floor mainly depends on the structural properties of $(n+1)$th floor and nth floor. If directly utilizing the raw vibration signals, each X^n contains the influence of all the other stories' vibrations and

noise during the earthquake. The above proof shows that the directed information effectively help reduce the noise induced by the structural changes in other nonadjacent locations compared to raw vibration signals. Meanwhile, it can be seen that the directed information from the ceiling to the floor of the nth story is different from the directed information in inverse direction. $I(X^n_{1:T} \to X^{n+1}_{1:T})$ focuses more on the properties of the nth floor, while $I(X^{n+1}_{1:T} \to X^n_{1:T})$ focuses more on the properties of the $(n+1)$th floor. In conventional methods, the information in the two directions is combined and extracted as a feature to infer the changes of structural properties. But by differing the information by directionality, the directed information provide more details and enable the analysis of groups of vibration signals.

2.5 Information-theoretic Approach for Structural Damage Diagnosis

Based on the physical insight and analytical relationship, we propose an information-theoretic approach to detect and quantify the earthquake-induced structural damage using the structural vibration responses to earthquake excitations. In this chapter, for simplicity, the approach is explained and implemented for story-level detection and quantification, but the method can be expanded to various scales of detection and quantification, depending on the sensor density. As Figure 4.2 shows, our approach includes three steps: data collection, feature extraction, and damage modeling. In this section, we first describe the collection of the story-level seismic structural responses and corresponding structural drift ratios. Then we describe how to extract directed and inverse directed information between the accelerations of the floor and ceiling of each story and compute the features. Finally, we train kernel-based support vector machine models for damage detection and damage quantification.

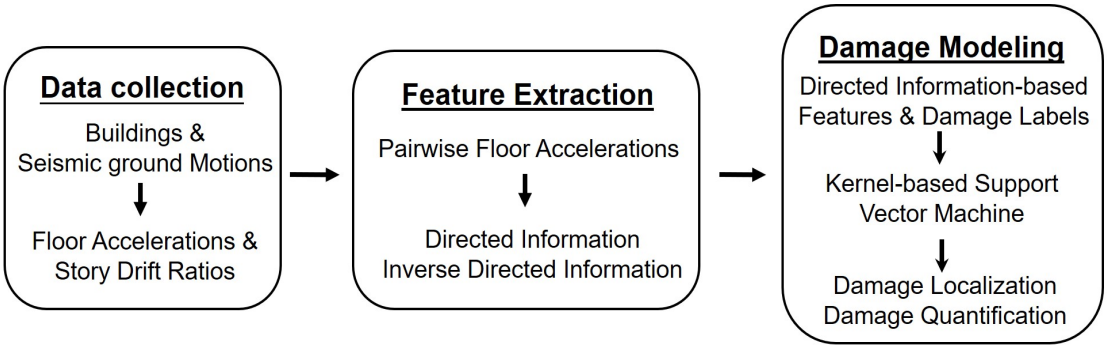


Figure 2.4: The algorithm overview.

2.5.1 Data Collection

We first collect structural responses to obtain sufficient data samples for damage modeling. In our method, the accelerations at each floor are collected during the earthquake excitation. For example, to extract the directed information as the feature of the nth story, we need the collected vibration signals at nth floor and $(n+1)$th floor. The during-earthquake structural responses depend on the dynamic earthquake excitations and the building structural properties, as shown in Section 2.4.

After the acceleration data are collected, we use sliding window to separate the vibration signal for further feature extraction. As shown in Section 3, we assume that in a short time window T, the structural properties are consistent. We use a sliding window with the size of T and the stride of 1 to separate the vibration signals into multiple pieces. In this way, each vibration signal is reshaped as a matrix with the size of $(l-T+1)\times T$ matrix, where l is the length of the vibration signal. Then the directed information is extracted from each pair of pre-processed vibration data collected from two adjacent floors.

2.5.2 Feature Extraction

The next step is to extract the directed information from a floor to the ceiling and from a ceiling to the floor as features of the corresponding story. To ensure the computational efficiency, the signals need to be quantized into S level with the principle of $T \approx \frac{S^{D+1}}{lnS}$, where T is the short time window, D is the order of the Markov process of seismic-induced vibrations, which is 1 in our scenario based on our state-space model described in Section 2.4. The directed information is extracted from each pair of sliding-window vibration signals, i.e. $I(X^n_{kT+1:(k+1)T} \to X^{n+1}_{kT+1:(k+1)T})$ and $I(X^{n+1}_{kT+1:(k+1)T} \to X^n_{kT+1:(k+1)T}), \forall k \in \{0,\cdots,K-1\}, n \in \{1,\cdots,N-1\}$. For simplicity, for each story n, we define the directed information from the bottom floor to the ceiling as "directed information" of the nth story, and the directed information from the ceiling to the bottom floor as "inverse directed information" of the nth story. After computing the directed information and inverse directed information for each sliding window, we obtain the final directed information as well as inverse directed information sequences both with the size of $K\times 1$. The extracted directed information sequences contain the information about how the structural properties change with time evolving.

Given the sliding-windowed vibration signals pair, $X^{n+1}_{kT+1:(k+1)T}$ and $X^{n+1}_{kT+1:(k+1)T}$, we extend the context-tree weighting algorithm [94] to estimate the directed information. As defined in Equation 2.5, to estimate the directed information $I(X^n_{1:T} \to X^{n+1}_{1:T})$, we separately estimate $H(X^{n+1}_{1:T})$ and $H(X^{n+1}_{1:T}||X^n_{1:T})$. After quantization, the random process becomes discrete. Define $P(X^{n+1}_{t+1}|X^{n+1}_{1:t})$ as the conditional probability mass function for X^{n+1}_{t+1} given $X^{n+1}_{1:t}$, which can be

estimated from the vibration signals. We can estimate the entropy as

$$\hat{H}(X_{1:T}^{n+1}) = \frac{1}{T}\sum_{t=1}^{T}\sum_{X_{t+1}^{n+1}} P(X_{t+1}^{n+1}|X_{1:t}^{n+1}) \log \frac{1}{P(X_{t+1}^{n+1}|X_{1:t}^{n+1})} \tag{2.22}$$

$$\hat{H}(X_{1:T}^{n+1}||X_{1:T}^{n}) = \frac{1}{T}\sum_{t=1}^{T} f(P(X_{t+1}^{n}, X_{t+1}^{n+1}|X_{1:t}^{n}, X_{1:t}^{n+1})), \tag{2.23}$$

where $f(P) = -\sum_{x,y} P(x,y)\log P(y|x)$.

To obtain the entropy and conditional entropy estimators, we employ context-tree weighting algorithm with fixed length of T and context tree depth of D. In this algorithm, we initialize the directed information estimator $\hat{I}(X_{1:T}^{n} \to X_{1:T}^{n+1})$ as 0. With the quantized sequences $\hat{X}_{1:T}^{n}$ and $\hat{X}_{1:T}^{n+1}$, we define $Y_t = (X_t^n, X_t^{n+1}), \forall t$. Then $\forall t \in \{D+1, T+1\}$, in the context of $Y_{t-D:t-1}$, we update the context tree for every possible value of Y_t and obtain the estimated probability mass function $P(Y_t|Y_{1:t-1})$. Similarly, we can obtain the estimated probability mass function $P(X_t^{n+1}|X_{1:t-1}^{n+1})$ based on the updated context tree for $X_{t-D:t-1}^{n+1}$. Every update ends with the updating of the directed information estimator

$$\hat{I}(X_{1:T}^{n} \to X_{1:T}^{n+1}) = \hat{I}(X_{1:T}^{n} \to X_{1:T}^{n+1}) + f(P(X_{t+1}^{n}, X_{t+1}^{n+1}|X_{1:t}^{n}, X_{1:t}^{n+1})) - f(P(X_t^{n+1}|X_{1:t-1}^{n+1})). \tag{2.24}$$

After iterating for $T - D$ times, we obtain the final directed information by taking the average, i.e. $\hat{I}(X_{1:T}^{n} \to X_{1:T}^{n+1}) = \dfrac{\hat{I}(X_{1:T}^{n} \to X_{1:T}^{n+1})}{T-D}$. Similarly, we can estimate the inverse directed information using the same algorithm.

2.5.3 Damage Detection and Quantification

Given extracted directed information as features, we conduct supervised learning by training different kernel support vector machines for damage detection and quantification. Here, with directed information quantifying the information exchanges between two locations, the task includes two aspects: damage detection and damage quantification. The damage detection focuses on detect whether there exists damages in each story. The damage quantification aims to quantifying the damage severity, which includes classifying the damage into several levels (classification-based quantification) and directly estimating the structural drift between two floors during the earthquake (regression-based quantification).

Story drift ratio (SDR) is a common index for identifying structural damages [40, 103, 188, 194]. We utilize SDR as the ground truth indicator of structural damages. According to FEMA

P695 [53], there are five damage states defined for the structure in terms of the peak absolute SDR ($\max(|SDR|)$) at each story, which are no damage ($0\% \leq \max(|SDR|) < 1\%$), slight damage ($1\% \leq \max(|SDR|) < 2\%$), moderate damage ($2\% \leq \max(|SDR|) < 3\%$), severe damage ($3\% \leq \max(|SDR|) < 6\%$), and collapse ($6\% \leq \max(|SDR|)$). For the damage detection, according to SDR level, we divide the structural damage state into two classes: damaged ($0\% \leq \max(|SDR|) < 1\%$) and undamaged ($\max(|SDR|) \geq 1\%$). For classification-based quantification, we use the aforementioned 5 classes as the true label. For regression-based damage quantification, we directly use SDR as the true label.

We use kernel support vector machine (SVM) to build the binary-class classification model to detect the structural damage. Given a set of training examples, SVM builds a non-probabilistic binary linear classifier. An SVM model is a mapping from data samples to a new feature representation space so that the examples of the separate classes are divided by a clear gap that is as wide as possible. Kernel support vector machine here is applied for both damage detection and quantification. Kernel support vector machine is good at dealing with the high-dimensional features of data through dimensionality reduction [41, 199]. For damage detection, we use kernel SVM to train the prediction model to detect the damage at each story. For classification-based quantification, we use multi-class kernel SVM. While for regression-based quantification, we use kernel support vector regression to estimate the values of the peak absolute story drift ratio.

2.6 Evaluation

In this section, we evaluate our approach with both simulated and experimental data. The simulated data are collected from 5 buildings with different heights under 40 earthquake excitations. The experimental data are collected from a 4-story building under 4 incrementally scaled of earthquake ground motions. To evaluate the performance of our features, we compare the performance of the same SVM models trained with our information-theoretic features (*DI-based features*), raw vibration signals (*Signal-based features*), and autoregression coefficients as features (*AR coefficient-based features*), which is another widely used features for building damage diagnosis [143].

2.6.1 Simulated Data With Numerical Models

Data Collection

We collect the structural vibration data at each floor from 5 buildings subjected to 40 earthquake excitations. The archetype of the buildings are located in urban California, United States [80]. The simulation is implemented in an open platform OpenSEES.

There are five archetype steel frame buildings with perimeter steel moment-resisting frames (MRFs). These buildings have 2, 4, 8, 12, and 20 stories respectively, with a first-story height

of 4.6m and a typical story height of 4m. More details about archetype buildings' design and geometries are described in the record [147]. The two-dimensional model of each archetype steel building considers the bare structural components of the MRFs. In the analytical model, the steel beams are idealized with an elastic element and a concentrated flexural spring at the center to represent the location of the reduced beam section. Under cyclic loading, the stiffness of steel components and deterioration of flexural strength are captured by modeling the springs with the modified Ibarra-Medina-Krawinkler model. For the first and third mode of all SMFs, Rayleigh damping ratio is assigned with the value of 2%. The natural periods of buildings are recorded.

As specified by FEMA P695, the Far-Field ground motion set is recorded to evaluate the performance of the building models. Horizontal ground-motions are scaled incrementally with respect to the first mode, 5% damped, spectral acceleration $Sa(T_1, 5\%)$ of the steel frame model through collapse. The time histories of floor absolute acceleration and story drift ratio under each incremental ground motion are recorded corresponding to each story of the 5 building models. As an example, Figure 5.10a and 5.10b show the story drift ratio of the 1st story and accelerations at the 1st and 2nd floor of a 12-story building under the ground motion observed at the Las Palmas Ave., Glendale station during the 1994 Northridge earthquake.

Feature characterization

We then extract and characterize directed information as effective features to indicate the structural damages. To extract the directed information, we first quantize the vibration signals. In our case, the signals are quantized into $S = 10$ levels. In general, to effectively estimate the directed information, a large number of quantization level is desirable. This is because with large quantization levels, the signal amplitude range for each quantization level is small (i.e., higher signal resolution) such that more information contained in signals can be extracted. However, this level cannot be too large because of the limitation of $T^* \approx \frac{S^{D+1}}{lnS}$. T^* is the sufficient sample number for calculating directed information between two signals. The sufficient sample number needs to be guaranteed to lower the estimation risk, i.e. $T > T^*$, where T is the final sliding window size we select. With the quantized vibration signals, we obtain the directed information and inverse directed information. By aligning the directed information and inverse directed information, we obtain the feature for each sample as a vector with length of $2(l - T + 1)$.

As an example, Figure 5.10c shows the extracted directed information and inverse directed information between the 1st floor and the 2nd floor of a 12-story building under a ground motion observed from the Northridge earthquake. We can find that between 3.75s and 7s, the absolute story drift ratio increases significantly and a more severe damage happens to the 1st story, as shown in Figure 5.10a. As shown in Figure 5.10b, it is difficult to observe the changes by comparing the

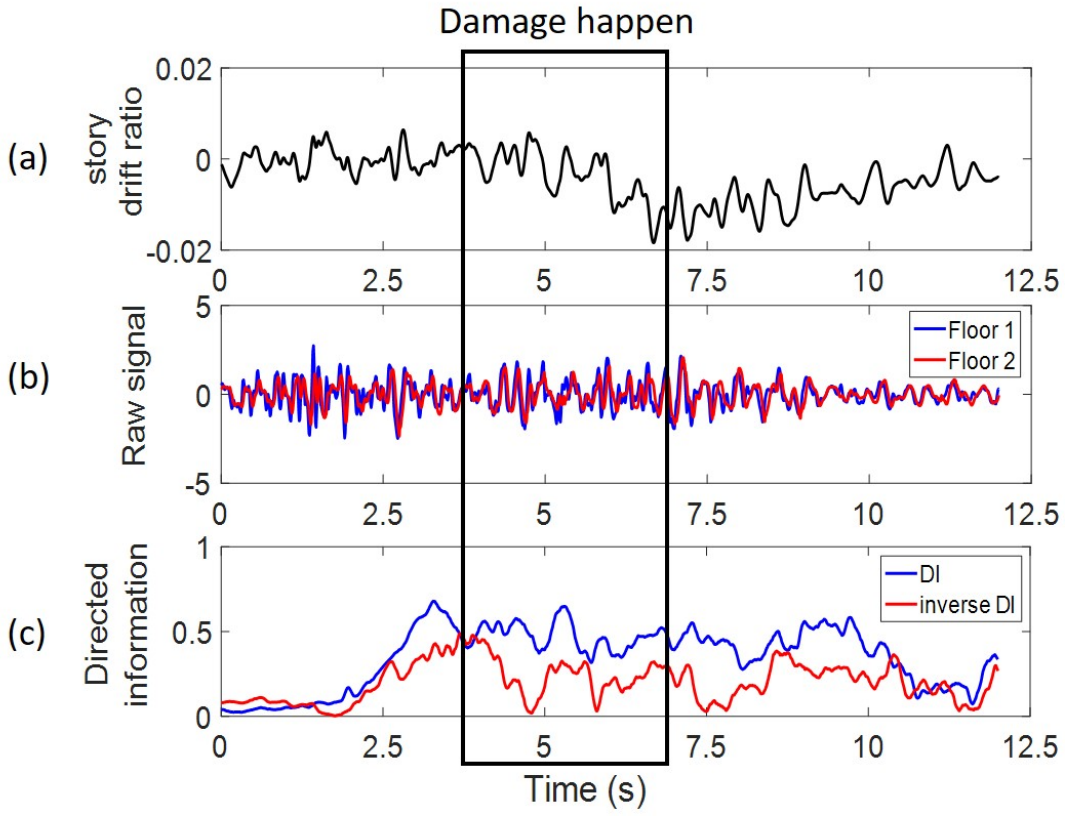
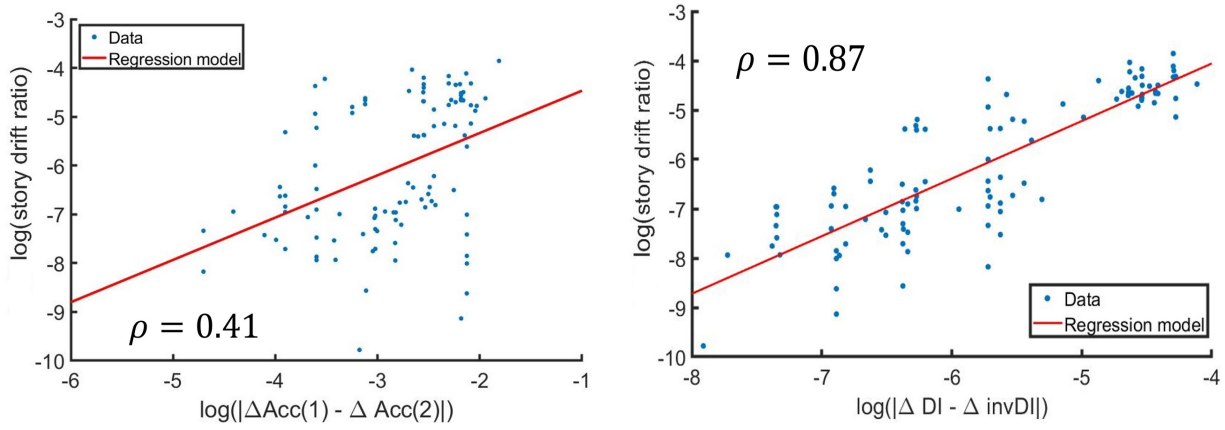


Figure 2.5: The figures visualize the (a) story drift ratio at the 1st story, (b) raw vibration signals at the 1st floor (blue) and the 2nd floor (red), and (c) directed information from the 1st floor to the 2nd floor (blue), and inverse directed information from the 2st floor to the 1nd floor (red) of a 12-story building under the ground motion observed at the Las Palmas Ave., Glendale station during the 1994 Northridge earthquake.

accelerations collected at the bottom floor and the ceiling. However, from Figure 5.10c, we observe that at the time of 3.75s, the difference between the directed information and inverse directed information suddenly increases and exhibit different trends. This shows that directed information is an effective damage index for the structural health conditions.



(a) (b)

Figure 2.6: (a) The logarithmic correlation between the differences between the gradients of accelerations at adjacent floors and the corresponding peak absolute story drift ratios in a 12-story building. (b) The logarithmic correlation between the differences between the gradients of directed information and inverse directed information on each story and the corresponding peak absolute story drift ratios in a 12-story building

To validate the analytical relationship shown in Section 2.4, we investigate the correlation between the directed information and story drift ratio from the collected data. From the Equations 2.19 and 2.21, we can obtain the difference between the gradients of the directed information and inverse directed information as

$$\Delta\text{DI} \triangleq I(X_{1:T}^{n+1} \to X_{1:T}^{n}) - I(X_{1:T-1}^{n+1} \to X_{1:T-1}^{n}) = \frac{1}{2} ln \frac{\det(P_T^{n+1} + Q_d^n)}{\det Q_d^n}.$$

$$\Delta\text{invDI} \triangleq I(X_{1:T}^{n} \to X_{1:T}^{n+1}) - I(X_{1:T-1}^{n} \to X_{1:T-1}^{n+1}) = \frac{1}{2} ln \frac{\det(P_T^{n} + Q_d^{n+1})}{\det Q_d^{n+1}}.$$

$$\Delta\text{DI} - \Delta\text{invDI} = \frac{1}{2} ln \frac{\det(P_T^{n+1} + Q_d^n)}{\det(P_T^n + Q_d^{n+1})} \cdot \frac{\det Q_d^{n+1}}{\det Q_d^n}.$$

It is shown that the difference between the gradients is dominated by the structural properties (P_T^n at the nth story. To explore the correlation between the directed information and the structural damage indicator, i.e., story drift ratio, We plot the pair of peak absolute story drift ratio verses difference of vibrations/directed information for each story in a 12-story building under multiple earthquake ground motions, as shown in Figure 2.6.1. Figure 2.6.1 shows the correla-

tion between the difference of the gradient of accelerations at two adjacent floors in each story. It is shown that the logarithmic correlation between the difference between the gradients of accelerations and the peak absolute story drift ratio is not strong with the correlation coefficient $\rho(\log|\Delta\text{Acc}(1) - \Delta\text{Acc}(2)|, SDR) = 0.41$. In contrast, the differences between the gradients of directed information and inverse directed information have more significant correlation with the story drift ratio with $\rho(\log|\Delta\text{DI} - \Delta\text{invDI}|, SDR) = 0.87$. This is also higher than the correlation coefficient $\rho = 0.69$ of wavelet-based features mentioned in the previous study [81]. Therefore, combining the observations in Figure 5.10 and Figure 2.6, we validate our analytical results which show the effectiveness of directed information as a more powerful feature to indicate the structural damages.

Results and discussion

In this section, we use kernel support vector machine to detect and quantify the structural damage state with the extracted directed information and inverse directed information as features. For the benchmark features based on autoregressive time series modeling of structure's acceleration response, there are several conventional methods for damage sensitive feature extraction. Here, the autoregressive coefficients are extracted by fitting vibration signals in each floor to the autoregressive model, and the coefficients extracted from accelerations in the floor and the ceiling of each story are combined as features for damage estimation. We use the binary-classification accuracy to indicate the damage detection performance, and 5-class classification accuracy to measure the performance of classification-based damage quantification. Meanwhile, to reduce the overfitting of the model, we used cross-validation to calculate the accuracy of the model.

To obtain sufficient data samples, we conduct data augmentation. We use a high-level sliding window with the length of 2000 and the stride of 50 data points to process each vibration signal. The length of the high-level sliding window is decided by the duration it takes the worst damage to happen from the starting time point. In each high-level sliding window, we extracted the directed information and inverse directed information with a local sliding window length of 200 time points, which is 1 second, as features of each story. Meanwhile, we label each samples according to the peak story drift ratio. If the peak story drift ratio is less than 0.01, we label the sample as undamaged, otherwise we label the sample as damaged. For the damage quantification, we label the sample into 5 classes as introduced in Section 2.5.3. In this way, we get the training dataset collected from the same story at different buildings.

Then we utilize kernel support vector machine to train the model and utilize cross-validation to evaluate the performance of our approach. With the high-dimensional directed information-based features, the problem of "curse of dimensionality" makes it difficult to optimize the damage model

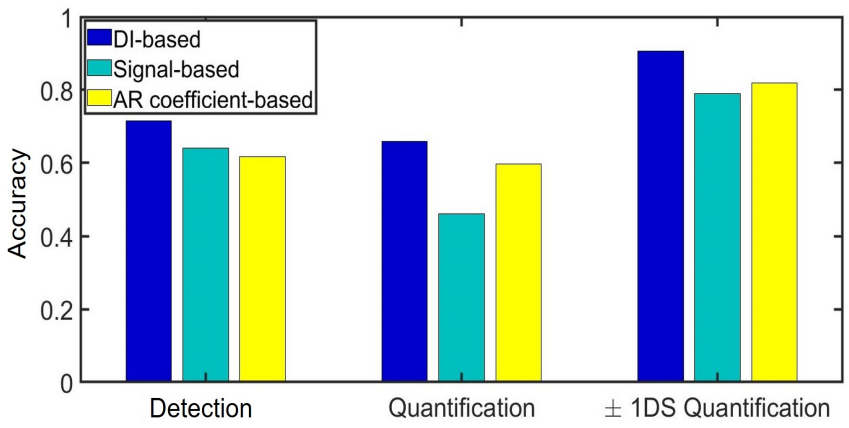


Figure 2.7: The accuracy of binary damage detection, 5-class damage quantification, and within 1 state damage quantification resulted by the DI-based feature(blue), signal-based feature(green), and AR coefficient based feature(yellow). The results show that our directed information-based features are more effective to predict the structural damages compared to other methods.

in the original feature space. The kernel trick is applied to reduce the dimensions of feature space to solve it efficiently. Here we utilize radial basis function kernel-based support vector machine. Both stochastic gradient descent (SGD) and limited-memory Newton algorithms (LBFGS) are applied for computational efficiency. For the bandwidth and coefficient of regularization term, the optimal values 5 and 0.02 are selected respectively using cross-validation. We use 5-fold cross-validation to obtain the final prediction accuracy for different stories. To obtain the overall performance, we take the weighted average across multiple stories according to the corresponding numbers of samples.

Figure 2.7 shows the damage detection accuracy, damage quantification accuracy, and the damage quantification accuracy within ± 1 damage state with *DI-based features*, *Signal-based features*, and *AR coefficient-based features*. For our *DI-based features*, the damage detection accuracy and damage quantification accuracy are 71.49%, 65.96% and 90.59%, respectively. It outperforms by other features on all 3 types of tasks. Compared to the conventional features, our information-theoretic approach achieve upto 9.8% improvement in the damage detection and 6.27% in the 5-class damage quantification. The accuracy of damage detection is higher than damage quantification, showing that the difference between damaged state and undamaged state is easier to learn by the model than the difference among 5 types of damage severity.

Figure 2.8 presents the damage detection accuracy at different stories. Our DI-based feature achieves higher damage detection accuracy compared to the signal-based method and AR coefficient-based method except at story 15. Especially, at the 20th story, our method achieves

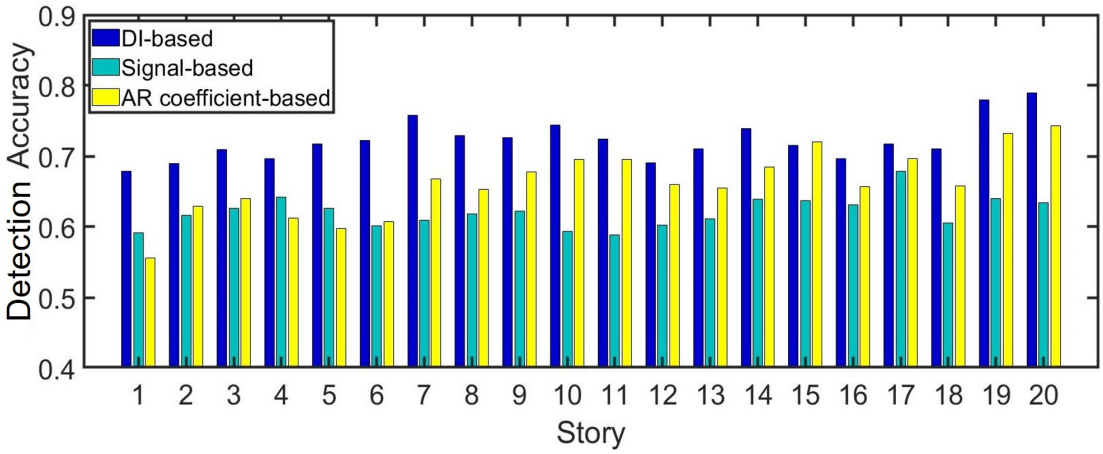


Figure 2.8: The story level damage detection accuracy using the DI-based features (blue), signal-based features (green), and AR coefficient based features (yellow).

15.48% improvement compared to the signal-based method. In lower story ($1 \sim 4$), the damage detection accuracy tends to be low. This may be because we combine the data from different buildings under different ground motions to train and test, which means that we assume these buildings' damage patterns are similar. However, the vibration signals collected from buildings with different heights under different ground motions may contain different noises introduced by the complex physical environment. This violation of assumption constrains the ability of the supervised learning model. For higher story $19 \sim 20$ story, the data is only available for the 20-story building, which avoids the assumption violation. Besides, it tends to be easier to predict the roof level damage combining the results in Section 2.6.2. Although the general accuracy tends to be low in lower stories, our method has the most significant improvements in accuracy in these levels, as shown in Figure 2.8.

2.6.2 Experimental data

In this section, our approach is evaluated using experimental data collected from shake-table tests of a 4 story moment-resisting frame. The data are used to analyze the response of directed information of each story to various earthquake intensities and compare performances (SDR estimation error) between our information-theoretic approach and benchmark methods.

Data collection

The data is collected from a 4-story steel moment-resisting frame under 1994 Northridge earthquake excitations. The experiment is conducted in State University of New York at Buffalo. The

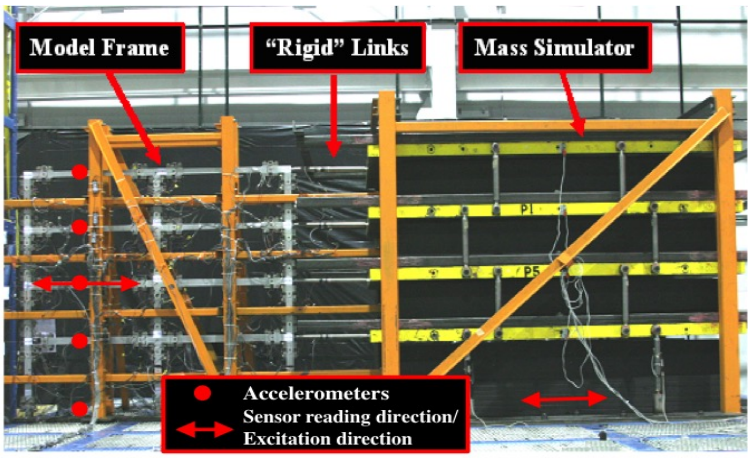


Figure 2.9: The frame of the four-story steel moment-resisting frame.

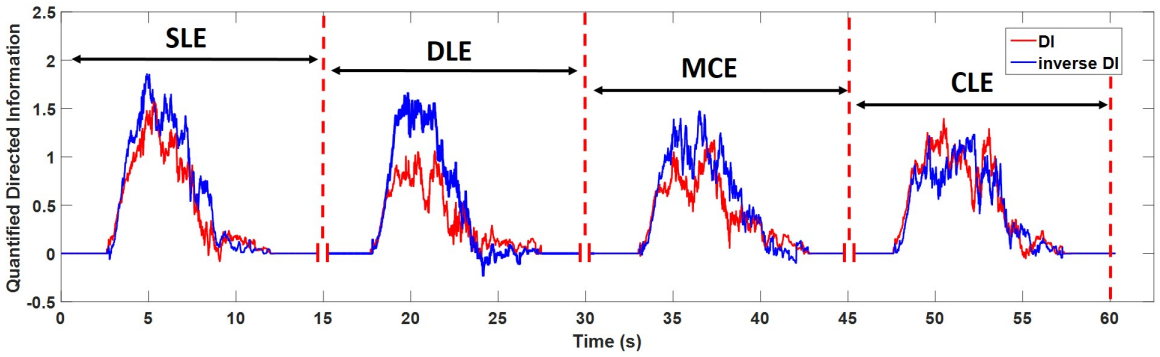
archetype of the frame is located at Los Angeles, CA and designed with reduced beam connections. The ground motions are recorded at Canoga Park Station from the the 1994 Northridge earthquake. Figure 2.9 shows the 1:8 scale model for the 4-story structure. At each floor, one accelerometer is installed to collect the vibration signals in horizontal direction.

The shake-table test sequence includes 4 intensities through collapse using incremental dynamic analysis. The test sequence includes a service level earthquake (SLE, 40% of the unscaled record), a design level earthquake (DLE, 100% of the unscaled record), a maximum considered earthquake (MCE, 150% of the unscaled record), and a collapse level earthquake (CLE, 190% of the unscaled record). The time history of accelerations at each floor under each earthquake excitations are recorded for data analysis.

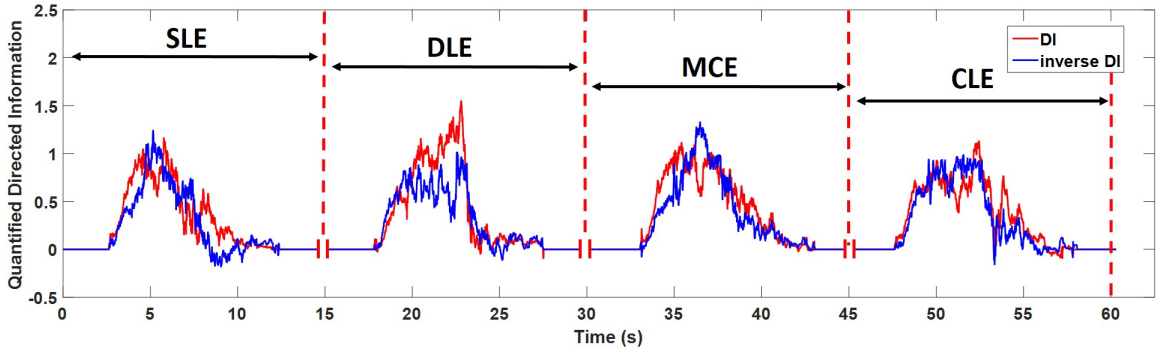
Feature characterization

Based on the collected structural vibration signals, the information exchanges in each story are extracted using directed information and inverse directed information of each story are extracted as features. Here quantization level is chosen as 10. The training set consists of the extracted directed information features for each measurement sample and story drift ratio (SDR) as the corresponding output.

The extracted directed information and inverse directed information are analyzed. Figures 2.6.2 and 2.6.2 show the time-series of directed information (red line) and inverse directed information (blue line) for the 1st story and the 2nd story under 4 increasing intensities of earthquake excitations (SLE, DLE, MCE, CLE), respectively. These figures show that 1) for different stories under the same earthquake excitation, the values of directed information and inverse directed information decrease with the increasing story number, which is due to the energy gradually dissipating as the



(a)



(b)

Figure 2.10: (a) Extracted directed information and inverse directed information at 1st story of the building under earthquake excitation with increasing intensities, (b) Extracted directed information and inverse directed information at 2nd story of the building under a series of earthquake excitations with increasing intensities.

wave propagates to the top story; 2) for the same story with different earthquake intensities, the relative values of directed information and inverse directed information change corresponding to various structural damage states shown in Figure 2.10. These characteristics of directed information and inverse directed information show that correlations exist between information exchange and structural damage in each story.

Results and Discussion

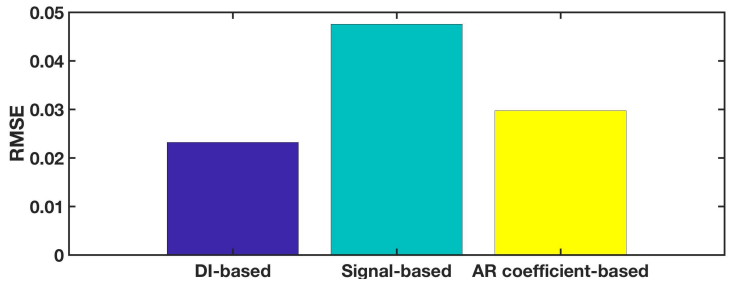
To explore the floor-level structural damages under earthquakes' effects, a regression method is applied to estimate the peak absolute SDR value during earthquake. When training the model, cross validation is used to evaluate the performance of the model. With cross validation, the mean square error is calculated to evaluate the accuracy of estimation. As shown in Figure 2.6.2, for

directed information based features, the mean square error between final estimation value and ground truth is 0.023. For the raw signal features, using the same training methods shown above, the performance of alternative models resulted in the mean square error of 0.048, which is 2.5X higher than the results of DI-based model. The mean square error for regression model based on AR coefficients is 1.3X higher than the mean square error of DI-based model. More detailed analysis results comparing the performance of the three approaches are shown in Figure 2.11. From Figure 2.6.2, we can find that our DI-based feature outperforms than other features if calculating the average mean squared error by the type of damage. From Figure 2.6.2, we can find that our method performs the best in service level earthquake excitation. Meanwhile, it can be found that DI-based model can estimate the SDR value with lowest mean square error in most stories (1st, 3rd and 4th) and its overall performance is the best among three models. In particular DI-based model performs best in the top (4th) story of experimental frame due to less sensor noise on the higher floors, as shown in Figure 2.6.2.

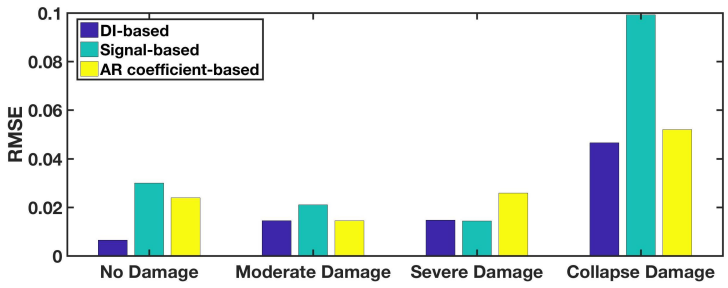
2.7 Conclusion

This chapter presents a new information-theoretic approach to diagnosing earthquake-induced structural damage. In our method, the process of wave propagation inside the building structure system is modeled as the process of information exchanges. We show both the physical insight and analytical proof of the physical relationship between structural dynamic characteristics and information exchanges. Extracted from structural vibration signals at each floor, the information exchanges are used as features for damage detection and quantification in story-level.

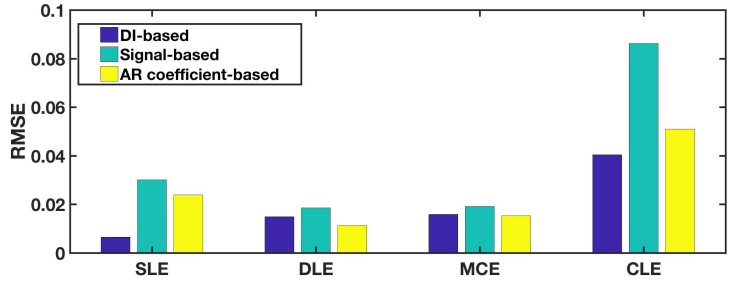
Our information-theoretic approach is evaluated in both simulated structural vibration data and experimental structural vibration data. As a result, our approach can achieve the accuracy of 71.49% in damage detection and 65.96% in damage quantification for the simulated data. Compared to the benchmark methods, our information-theoretic approach achieves upto 15.48% improvement in damage detection. For the experimental data, the accuracy of damage detection achieves 94%. In addition, estimation error for story drift ratio improved by factors of 1.3X and 2.5X when compared with conventional methods. This information-theoretic approach does not need to assume a particular structural model, or probability distribution of the vibration data. Furthermore, our approach uses only during-earthquake data from sparsely deployed sensors for detecting the existence of damage and estimating the actual story drift ratio at each story in a computationally efficient way.



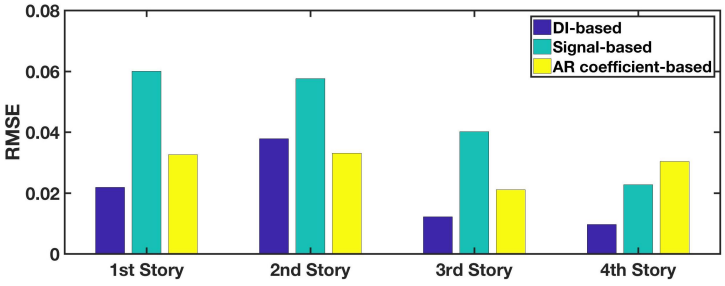
(a)



(b)



(c)



(d)

Figure 2.11: (a) The root mean squared error of estimating the story drift ratio in all stories under all earthquake excitation using our proposed DI-based features, signal-based features and AR coefficient-based features. (b) The root mean squared error of estimating the story drift ratio under multiple damage cases. (c) The root mean squared error of estimating the story drift ratio under different earthquake intensities. (d) The root mean squared error of estimating the story drift ratio from 1st to 4th story.

Chapter 3

Knowledge Transfer Across Different Infrastructures Without Labeled Data

3.1 Problem Overview

To accurately and timely diagnose structural damages, data-driven methods based on structural vibration sensing are developed and have received many attentions [164, 234]. By combining advanced machine learning techniques in a supervised fashion, these intelligent data-driven building damage diagnosis methods utilize historical vibration signals and corresponding true damage state (label) to train statistical models for building damage diagnosis [66, 82, 168, 224]. With the strength of modelling complex uncertainties and eliminating the need for explicit modeling of structural properties, these methods perform well when there is massive historical data collected from the same building of interest.

However, extensive collection of historical data, especially the true labels, is very difficult and expensive, if not impossible, in real-world practices. One of the most common damage labeling practices is to conduct visual inspections by trained human experts, which is labour-intensive, time-consuming and dangerous for human experts. In recent years, people also utilize different types of sensors to measure the structural damages as labels. But these sensors are either expensive to deploy or have strict requirements on the operation conditions. For example, story drift ratio (SDR), defined as the relative translational displacement between adjacent floors, is one of the commonly utilized engineering damage parameters to indicate the true structural damage states [61, 214, 234]. But accurately measuring the structural drifts after earthquake requires high-resolution expensive sensors (e.g. position sensitive sensors) [90, 144, 177]. It is costly to widely deploy these expensive sensors on buildings in earthquake prone areas. Besides, the frequency of similar types of earthquakes happening to the same building of interest is very low. Even though the costly sen-

sors are deployed on target buildings, there is very limited number of labels available due to rare earthquakes.

Meanwhile, it often results in low performance if we directly adopt the damage diagnosis model learned from other buildings to estimate the damage states of the building of interest. Many existing supervised learning methods are developed under the hidden assumption that the data used for constructing a model (training data) have the consistent input-label joint distribution as the data used for damage diagnosis of the current structure of interest (test data) [174]. As Figure 3.1 shows, in practices, different buildings often have distinct characteristics such as geometries, material properties, and foundation conditions, and thus their data distributions are characterized very differently. Directly adopting models learned from other buildings to diagnose new buildings would violate the aforementioned underlying assumptions. The violation of assumptions lowers the performance of developed supervised data-driven approaches, and significantly constrains the wide applications of data-driven building damage diagnosis in post-earthquake scenarios. Especially, directly combining the historical data from multiple other buildings to train one model and applying it to the building of interest may make the performance worse since the distributions across these buildings are already quite distinct to fit into a single model.

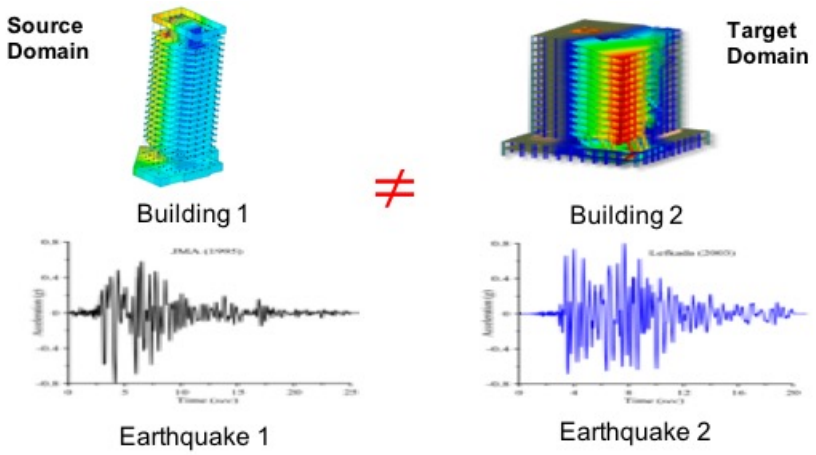


Figure 3.1: The data distribution (features, damages) of building 1 under Earthquake 1 does not equal to the distribution of building 2 under Earthquake 2 in real-world practices, which violates the underlying assumptions of most current supervised learning methods that data distribution is consistent across the training and test dataset.

To this end, the machine learning community has introduced domain adaptation techniques to address these knowledge transfer problems [60, 69, 174, 176, 240]. We denote the buildings which labelled data are collected with and the knowledge are learned from as "Source Domain," and the new building of interest without any labelled historical data as "Target Domain". This knowledge transfer problem is called multiple source domain adaptation when having the histor-

ical data from multiple different buildings (multiple source domains) to learn. Previous domain adaptation studies fall into several distinct categories such as instance-based [240] and feature representation-based [60, 69, 176]. The key idea of domain adaptation techniques is to extract feature representations such that the extracted features across different domains subject to a consistent distributions in the new space of representations. In this new space of representations, with consistent training and test data distributions, we can adopt the model learned from other source domains to diagnose damages of the target building. A successful domain adaptation should ensure the extracted feature representations of different buildings "domain-invariant" and "discriminative". The extracted features are called ***"domain-invariant"*** features if the distributions of feature representations across different domains are aligned. Meanwhile, the extracted feature representations need to be ***"discriminative"*** with respect to structural damages to ensure the damage diagnosis accuracy on the source buildings.

However, there are still challenges for knowledge transfer from source buildings to the target building for earthquake-induced structural damage diagnosis. *First*, different buildings may suffer from different earthquakes with various fast-changing dynamics and environmental noise, which is coupled in building responses and hard to distinguish from building structural response. To extract building-invariant features, we need to quantitatively eliminate the impact of different earthquake characteristics. Conventional single floor-based input features (e.g. raw floor vibration signals, or Fourier Transform of floor vibrations) does not contain sufficient information about the earthquake excitations due to the complex wave propagation process inside structures.

Second, the changes of distributions between source domains and the target domain depend on complicate structural and non-structural components. This makes it difficult to model the distribution changes and extract feature representations which retain domain-invariance and discriminativeness simultaneously. During the earthquake, the soil-structure interaction system is a fast-changing highly dynamic time-variant system. Therefore, the damage patterns, i.e. the functional mapping from the feature representations to the structural damages, vary from buildings to buildings. Simply forcing the domain-invariance of feature representations would easily ignore the domain-variant structural damage-related information contained in the original data, which makes the extracted features poor predictors for structural damage. On the other hand, if we focus on extracting discriminative features regardless of underlying distribution changes across different buildings, it would be difficult to ensure the domain-invariance of extracted features, which makes it difficult to transfer knowledge to the target building. Therefore, their trade-offs in the earthquake-induced building damage diagnosis problem need to be investigated and resolved.

Last but not the least, the data from different source buildings may induce different levels of biases to the learned model, which makes it difficult to best utilize the limited data from each

source building to integrate the knowledge from multiple source buildings. Due to the aforementioned problem of costly label collection, each source building usually has very limited number of data, and thus provides very limited information about earthquake-induced structural damage patterns. One possible solution is to integrate and transfer the knowledge learned from the data collected on multiple source buildings, which is called as "multiple source domain adaptation". In conventional multiple source domain adaptation methods, there exists an implicit assumption, that is, all of the source domains have the equal importance to provide equally sufficient information to learn domain-invariant features. However, in real-world practices, some source buildings data have significantly different distributions from the target building data. That is, different source buildings have different levels of biases compared to the target building's data distribution. This may be induced by distinct physical properties or too limited dataset constraining sufficient statistical estimations. Treating all source buildings the same important ignores this difference of bias levels, which may introduce high biases to the diagnosis model from those source buildings with distinct data distributions from the target building. This is because that the discriminativeness of extracted features on the target building would be sacrificed to ensure the features' domain-invariance on those distinct source buildings, which finally reduces the performance of knowledge transfer.

To address these challenges, we introduce a new physics-guided modeling framework which transfers the knowledge learned from multiple different source buildings to help diagnose the story-wise healthy conditions of the target building structure without any labels. To eliminate the influences of earthquake types on building-invariant feature extraction, we design a new multi-channel input features by combining the frequency information of ground motion, floor and ceiling responses as a tensor. This new input features help the statistical model better understand the local correlations between ground motions and floor vibrations at different frequency bands. To effectively extract domain-invariant features while retaining the discriminativeness of the extracted features, we present an adversarial modelling framework which integrates a feature extractor, a domain discriminator, and a label predictor with deep neural network architectures. The feature extractor is jointly optimized with the domain discriminator and label predictor in an adversarial way to find the optimal trade-off which ensure that the extracted domain-invariant features are still discriminative. The extracted domain-invariant features enables better understand the influences of physical relationships between various building properties, earthquake excitation and damage distributions. To eliminate the noise introduced by less similar/unrelated source buildings' distributions, we design a new loss function for the joint optimization based on simple and fuzzy prior physical knowledge, which is shown to significantly improve the performance in real-world practices.

This work has four main contributions:

1. To best of our knowledge, the framework we introduce is the first domain adaptation framework for earthquake-induced building damage diagnosis without any labels of the target building. This end-to-end framework integrates data augmentation, input feature extraction, and domain adaptation for damage diagnosis tasks including damage detection and damage quantification. Besides, this multiple source domain adaptation framework is flexible to combine the information from multiple source buildings to help diagnose the target building.
2. We introduce a new adversarial training framework to learn and transfer the knowledge from multiple heterogeneous source buildings. This framework ensures both the discriminativeness and domain-invariance of the extracted features, which makes it robust to the environmental noise and complicate distribution changes between different buildings and earthquakes.
3. We design a new physics-guided loss based on fuzzy physical knowledge about buildings, which eliminates the biases introduced by those source buildings with less physical similarities to the target building. We prove that the proposed new physics-guided loss provides a tighter upper bound for the damage prediction risk on the target domain compared to the general loss without combining physical knowledge.
4. We characterize the performance of proposed framework using both numerical simulation data and real-world experimental data, including 5 different buildings under more than 40 earthquakes for simulation and a real-world 4-story building.

This remainder of this chapter is organized as follows. Section 3.2 discussed related work about the domain adaptation and its applications on structural health monitoring. Section 3.3 analyzes the domain adaptation challenges in earthquake-induced building damage diagnosis scenarios. Section 3.4 describes our proposed knowledge transfer framework for building damage diagnosis, including the problem definition, model architectures, loss function design, and adversarial domain adaptation training scheme. Section 3.5 evaluates our knowledge transfer framework using both numerical simulation data and real-world experimental data. Section 3.6 concludes the work and gives further discussions.

3.2 Related Work

Earthquake-induced building damage diagnosis is a challenging problem. In recent years, wide deployment of vibration sensing systems on buildings provides rich building responses during earthquakes and enables the applications of data-driven approaches for earthquake-induced building damage diagnosis [66, 92, 124, 168]. Statistical models or machine learning techniques, such as multivariate linear regression [82], support vector machine [68], kernel regression [234], deep

autoencoder [179], and deep convolutional neural networks [2], are utilized to extract damage indicators from structural responses during earthquakes, either in frequency domain or time domain, and then estimate the building damages [80, 82, 164, 234]. However in practices, many buildings often have very limited or even no labels available for the collected structural responses data, which makes it difficult to train the damage prediction model for the target building.

In machine learning communities, people introduced domain adaptation for knowledge transfer between inconsistent training and testing data distributions. In domain adaptation, the input data are in the same space for both source and target domains, and the learning tasks, e.g. damage diagnosis, are the same across the source domain and target domain [44, 238, 244]. Especially, unsupervised domain adaptation focuses more on the unsupervised learning tasks in the target domain [43, 217], which means that the target domain data has no labels. In practice, most structural damage diagnosis tasks using other buildings' data are domain adaptation problem, specifically, unsupervised domain adaptation problem. This is because 1) the feature spaces are the same, although the distributions of features may be different. For example, we have the same sensing modality, e.g. floor vibration signals, for both source and target domains. 2) We have the same damage prediction tasks for different buildings since the definition for building damage states usually subjects to a consistent standard. 3) We have no label on the target building.

Traditional domain adaptation includes 2 types of approaches: instances-based and feature representation-based. Instances-based learning matches the joint distribution $P(X, Y)$ between the source and target domain by re-weighting the labeled instances (X, Y) from the source domains, where X is the input feature and Y is the label. Feature representation-based methods focus on learning a representation of the input X in a reproducing kernel Hilbert space (RKHS) in which the feature representation distributions of different domains are close to each other (**domain-invariant**). Our work falls into the feature representation-based class, which aims to first extract domain-invariant feature representations and then then build an optimal damage classifier based on the extracted feature representations. Especially, we need to learn domain-invariant feature representations across multiple source domains and the target domain.

In recent years, feature representation-based domain adaptation has been widely studied in different areas, such as image recognition [174, 176], natural language processing [104], sentiment analysis [18] and etc. Classic traditional methods include transfer kernel learning [132] (TKL), geodesic flow kernel [63] (GFK), joint distribution adaptation [131] (JDA), and transfer component analysis [175] (TCA). However, these methods mainly focus on learning shallow feature representations. Deep learning models are embedded to conduct more complex transformation to extract feature representations [49, 62, 233]. Deep neural networks have stronger expressive power to extract robust domain-invariant feature representations disentangling underlying exploratory factors

of variations and hierarchically combining features according to their correlation with invariant factors. However, these methods often work with fixed feature representations, which makes it difficult to ensure the extracted features both discriminative and domain-invariant.

To ensure the discriminativeness and domain-invariance of the extracted feature representations, researchers developed adversarial framework to embed domain adaptation into the process of feature representation extraction. In the adversarial frameworks, a domain discriminator block is added to distinguish between the samples from source domain and target domain, which provides strong regularization to encourage the domain-invariance of the extracted feature representations [56, 57, 212, 213]. Deep domain confusion loss is designed to encourage the data distributions for the source domain and target domain are as close as possible [212]. The gradient reversal algorithm is proposed to achieve adversarial training by reversing the gradient of domain discriminator during the back-propagation [56]. Adversarial discriminative domain adaptation is proposed to alternately learn the feature representations and train the domain discriminator [213]. However, these frameworks all focus on single-source-single-target problem. When there are multiple source domains, the distribution changes becomes more complicate. Naive applications of those single-source-single-target methods would result in sub-optimal solutions.

Some existing multiple source domain adaptation approaches are mostly based on fixed feature representation learning and can not utilize effective deep neural network models [55, 76]. Zhao et al. introduced a deep adversarial domain adaptation method for multiple source domain adaptation. This method combines multiple domain discriminators to extract domain-invariant features across multiple source domains and the target domain [243]. New loss function is designed to ensure extracted features' discriminativeness for damage prediction on multiple source domains. However, this method treats all domains with the same importance for domain-invariant feature learning. The main goal of domain adaptation is to ensure the prediction performance on the target domain without any labels. It is difficult to ensure each domain contains equally important information for the target tasks in real-world practices. In this work, we introduce a new loss function to focus on more informative source domains.

3.3 Domain Adaptation Challenges for Post-earthquake Building Damage Diagnosis

In this section, we first show the common challenges in domain adaptation for post-earthquake building damage diagnosis scenarios. Then we characterize the data distribution changes with different earthquakes and building types.

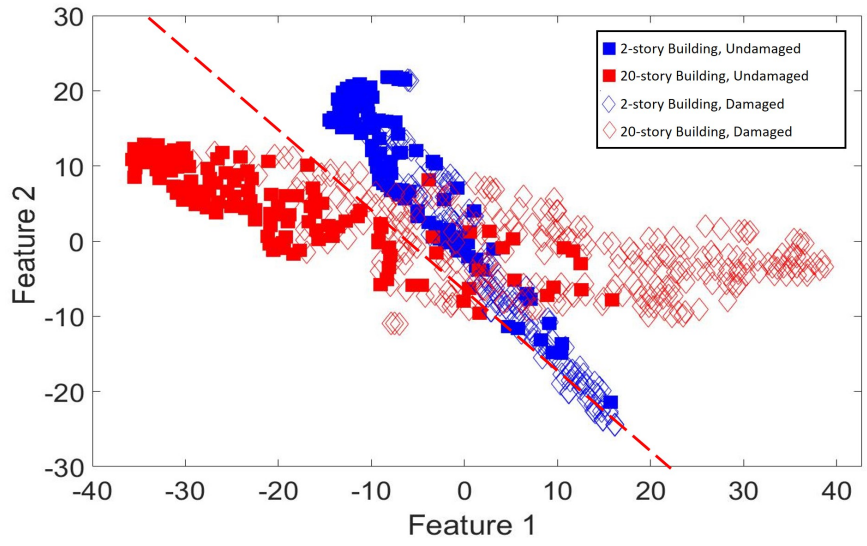
Domain adaptation is important for enabling wide applications of data-driven post-earthquake building damage diagnosis in data-constrained scenarios. In general framework of data-driven

approaches, structural response, e.g., the structural vibration signals, are collected as input x, the respective structural damage states are label y, and a relation between x and y are defined as a function $\mathcal{F} : x \rightarrow y$ in a discriminative way or the joint distribution $P(x, y)$ in a generative way. A common underlying assumption for general supervised learning methods is that the marginal distributions $P(x)$ and $P(y)$ and the joint distribution $P(x, y)$ are consistent between the training dataset and the test dataset.

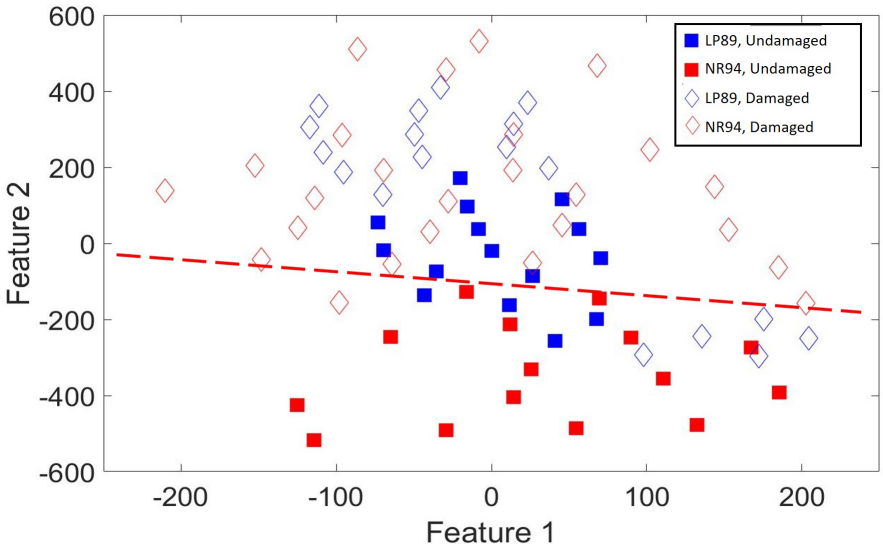
However, in our scenario, this underlying assumption is often violated when using labelled historical data from different buildings to predict the damages of the target unlabelled building. Denote the building with labelled data which we would like to transfer the knowledge from as source domain, i.e., s. Denote the building without any labelled data that we need to diagnose as target domain, i.e., t. The structural response and damage states collected from the source domain are X^s and y^s, respectively, and those from the target domain are X^t and y^t, respectively. The data distribution in the source domain does not equal to the data distribution in the target domain, i.e., $P(X^s, y^s) \neq P(X^t, y^t)$. Figure 4.6(a) presents the 2 dimensional t-Distributed Stochastic Neighbor Embedding (t-SNE) visualization [139] of structural vibration signals collected from a 2-story building (blue) and a 20-story building (red). The red dotted line refers to the true decision boundary for damage detection of the 20-story building (red points). The performance of damage detection for the 2-story building drops dramatically if the decision boundary learned from the 20-story building is used. Meanwhile, the data distributions of the same building under different earthquake excitations are also different. Figure 4.6(b) shows the visualization of structural vibration signals collected from the 1989 Loma Prieta (LP) earthquake (blue) and the 1994 Nothridge (NR) earthquake (red). The red dotted line represents the damage decision boundary for buildings under NR earthquake. Directly adopting the decision boundary learned from NR earthquake results in inaccurate damage decision for LP earthquake.

3.3.1 Characterization Of Distribution Changes With Earthquakes And Buildings

The data distributions change with different earthquakes and buildings, which is mainly caused by different of damage progression patterns inside the structures. The physical properties of earthquakes, including ground motion intensities, length, seismic waveform, significantly influence the structural damage patterns. Meanwhile, the physical properties of buildings (e.g. stiffness, damping, structured design, and non-structural components) are closely related to the damage occurrence. Here we utilize the incremental dynamic analysis data simulated using 5 numerical models of steel moment-resisting frames (MRFs) subjected to 40 ground motions to visualize and give a basic understanding about the complicate distribution changes across different buildings and



(a)



(b)

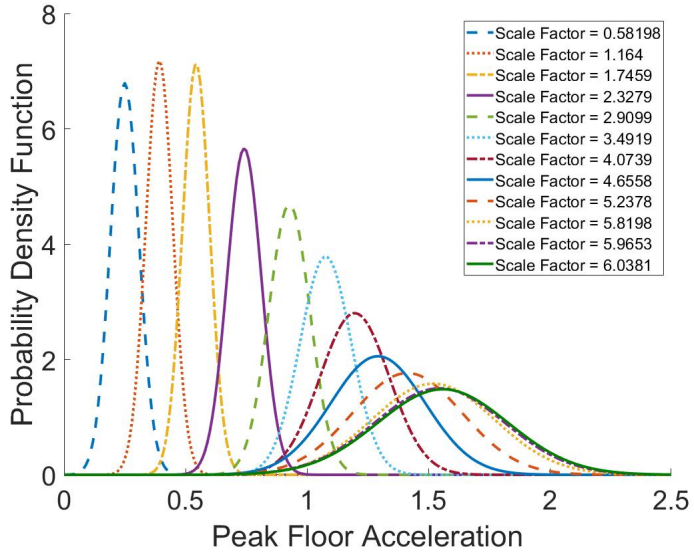
Figure 3.2: The 2D tSNE visualization of data distribution of different buildings under earthquakes changes significantly. The red points represent data sampled from the source domain, the blue points represent data sampled from the target domain. The red dotted line shows the classification decision boundary for source domain (red points). Diamond indicates damaged and square indicates undamaged. (a) shows the difference of data distribution between 20-story building and 2-story building. (b) shows the difference of data distribution between 1994 Northridge earthquake and 1989 Loma Prieta earthquake. Both figures show that directly applying model trained on source domain dataset (red) to diagnose structural damage on target domain (blue) will result in low performance.

earthquakes.

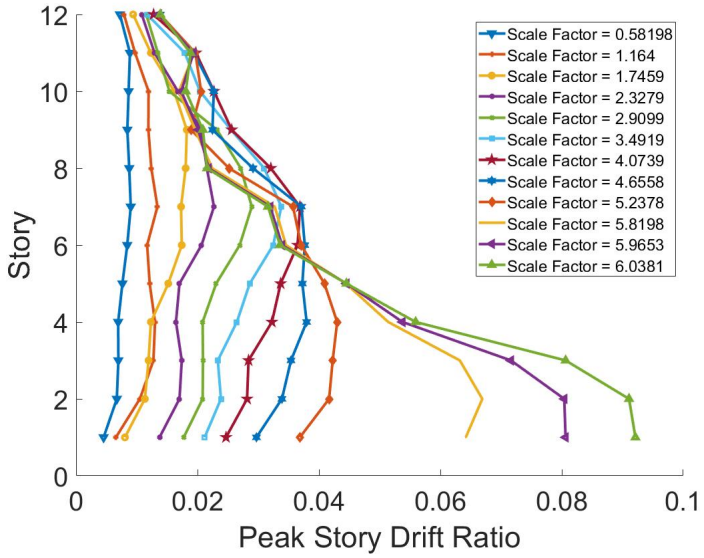
Impact Of Ground Motion Intensities

The previous research works show that the influence of earthquake excitations on structural damage patterns is significant and difficult to accurately model [207, 211]. The process of damage occurrence and progression is highly non-linear time-variant involving complicated wave transmission inside structures, which makes it difficult to quantitatively and accurately model the impact of earthquakes. It has been shown that the correlation between damages and ground motion characteristics are not consistent based on the observations in buildings in Mexico City [207]. There exist many studies exploring load-deformation model relating the engineering ground motion parameters to the structural damages. For example, the ground motion intensity is discovered to be related to hysteretic energy [178]. Destructiveness of seismic ground motions are also related the seismic duration, maximum absolute ground acceleration and frequency content of the respective strong ground motion [11]. Here we give examples to show that in a 12-story building, how data distributions, including the distributions of peak absolute floor acceleration (PFAs) and peak story drift ratio (SDRs) change with the ground motion intensities. Figure 3.3.1 shows the PFAs distribution changes with different intensities of a seismic ground motion (the ground motion observed in Station Gilroy Array #3 during 1989 Loma Prieta earthquake). Different scale factors for incremental dynamic analysis indicate different ground motion amplitudes used for simulations. We use normal distribution to fit the density estimation of PFAs. With intensity increasing, both mean and variance of PFAs increases. Statistically, with intensity increasing, the support of PFA distribution spreads and has more overlap with those distributions under low ground motion intensities. When there are limited data which mostly fall into the overlapping supporting range, it would become more difficult to distinguish the distribution changes. Figure 3.3.1 shows how the peak story drift ratio changes with increasing ground motion intensities on each story of the 12-story building. With ground motion intensity increases, the SDRs of middle stories first increase. When intensities become higher, the lower stories ($1 \sim 5$ story) are severely damaged and finally collapse. This transition trend indicates that under different ground motion intensities, the difficulty of knowledge transfer at different story levels varies a lot.

In summary, Figure 3.3 shows that even with the same type of ground motions, the building exhibit distinct response and damage patterns under different intensities. When different buildings subjected to various types and intensities of earthquakes, the changes of their structural responses and damage patterns become more complicated and hard to model. In conventional works, we often use single-floor-vibration-based features as input to the story-wise damage diagnosis models by assuming both training and test data are subjected to the same earthquake loading. But in



(a)



(b)

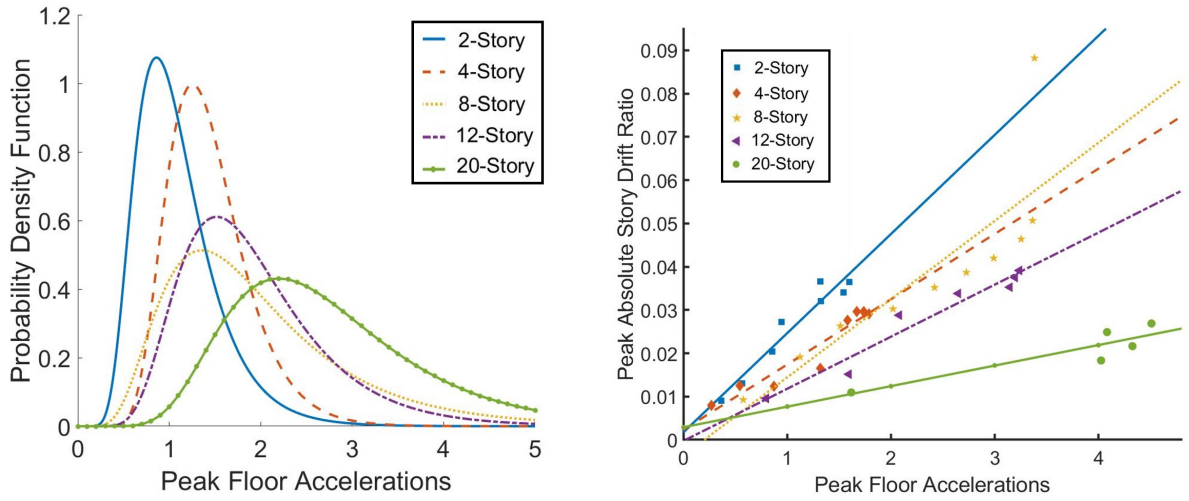
Figure 3.3: The distributions of (a) peak absolute floor accelerations (PFAs) and (b) peak story drift ratio (SDRs) changes with different ground motion intensities. The data is collected based on incremental dynamic analysis of a 12-story building based on the ground motion observed in Station Gilroy Array #3 during 1989 Loma Prieta earthquake. Note that the figures only show the data collected from a single building under a single type of earthquakes. The real-world distribution changes are much more complicated and intractable.

the domain adaptation scenario, ignoring the earthquake loading information would mislead the model in finding earthquake-invariant features. Therefore, seismic ground motions are important information to extract building-invariant features.

Impact Of Building Heights

The structural damage pattern is closely related to various structural and non-structural components, including the type of the structural frames, stiffness, damping, and height. These physical properties influence the structural system's elastic and elasto-plastic behaviour under earthquakes and thus are key elements to determine the structural response-damage distributions [39, 160, 198]. This influences on building damage distributions are often complicated, implicit and difficult to obtain during earthquake. For example, the input energy is not only related to the elastic period of the structure, but also the viscous damping and the characteristics of the plastic response [39]. More importantly, we often lack of the detail prior knowledge about the building structures (e.g. stiffness, damping) or quantify the physical knowledge (e.g. frame type) in earthquake scenario. The lack of sufficient physical knowledge makes it impossible to directly learn the influences of physical properties on structural damage patterns. Compared to aforementioned physical properties, building heights is a simple characteristics easier to be obtained. As an example, we show the influence of building heights on data distributions of different buildings. We show under similar ground motion intensities, how data distributions change with buildings with various heights in Figure 3.4. We present the PFAs and mean SDRs of the 2nd story of 2-story, 4-story, 8-story, 12-story and 20-story buildings with SMFs under the ground motions with similar intensities observed in Canoga Park-Topanga Can station during 1994 Northridge earthquake. Figure 3.3.1 shows the peak absolute floor acceleration distributions fitted with *Log-Normal* distributions. It is shown that the distributions of structural responses are distinct from each other. The general trend is that at the same story, high-rise building tends to have more spread structural responses. Figure 3.3.1 shows the correlations between PFAs and corresponding peak SDRs of different buildings. It is shown that low-rise buildings (e.g. 2-story and 4-story building) have larger SDRs, indicating more severe damages. This is because the low-rise buildings tend to experience the largest increase in ductility demands [9].

A general trend is that the buildings with similar heights tend to have similar damage patterns given the same type of structural frame and the same ground motion. The similar physical properties indicate similar responses and deformation patterns given consistent earthquake loading. If we had all prior physical knowledge about different buildings and ground motions, we can build transfer functions to learn the mapping from the structural responses to damage states. However in practices, the detailed physical knowledge are often difficult to obtain. We may have



(a) (b)

Figure 3.4: The distributions of (a) peak absolute floor accelerations (PFAs) and (b) correlations between mean peak story drift ratio (SDRs) and peak absolute floor accelerations (PFAs) change with different building heights. The data is collected based on incremental dynamic analysis of 2-story, 4-story, 8-story, 12-story and 20-story buildings based on the ground motion observed in Canoga Park-Topanga Can. station during 1994 Northridge earthquake.

some approximated and simplified physical knowledge, e.g. building heights, story/height ratio, strong-column/weak-beam (SCWB) ratios and etc. It is difficult to directly learn the distribution transitions based on these simplified physical knowledge. But it is still meaningful to quantitatively combine these fuzzy physical knowledge and let it guide and improve the training of powerful data-driven models. In later sections, we will show how we combine the building heights into the design of loss function for our data-driven model and the significant improvement.

3.4 Physics-guided Adversarial Domain Adaptation Framework for Infrastructure Damage Diagnosis

In this chapter, we introduce a new intelligent building damage diagnosis model which transfers the knowledge learned from multiple other buildings to predict and quantify the damage states of the target building without any historical data. As Figure 3.5 shows, the proposed framework includes several steps: data preprocessing, feature extraction, adversarial domain adaptation and damage diagnosis. Adversarial domain adaptation jointly train 3 neural networks, including domain-invariant feature extractor (E), damage predictor (M) and domain discriminator (D). In damage diagnosis, we directly input the extracted features for target domain data to well-trained damage predictor to

finally diagnose the damage states of the target building to obtain 1) domain-invariant and discriminative features, and 2) a well-trained damage predictor which maps from the extracted features to damage states. In this section, we first define the problem of multiple source domain adaptation for earthquake-induced building damage diagnosis. Then we introduce our new input features combining the earthquake information and structural responses. Then we present the architecture of the proposed framework, including the feature extractor, domain discriminator and damage predictor. To train the complicate neural network combination, we present our new physical loss function. Finally, we describe the adversarial training process for the proposed framework. With the adversarial training process, we would obtain the final feature extractor and damage predictor for final damage prediction and quantification.

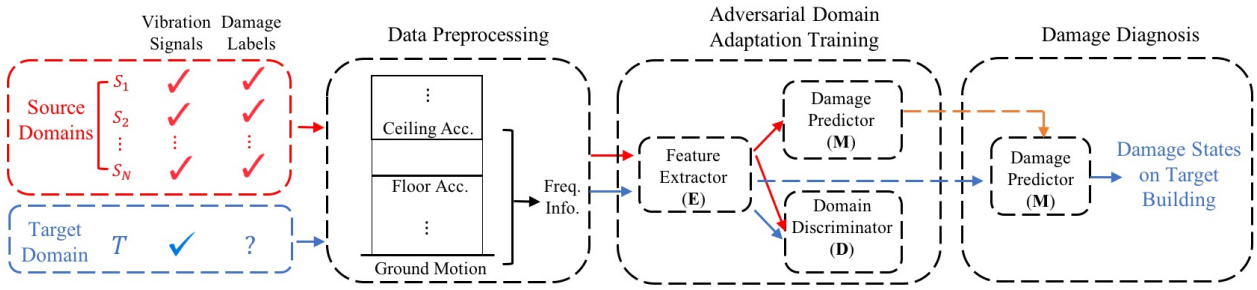


Figure 3.5: Proposed adversarial framework for multiple source domain adaptation for building damage diagnosis.

3.4.1 Problem Formulation

Our main objective is to learn and transfer the story-wise structural response-damages model extracted from multiple source buildings with labelled historical data to the target building without any labelled historical data. Here the input is story-wise structural vibration response, and label is the respective structural damage states. Denote the source domain as S and target domain as T. Assume that we have $n \geq 2$ buildings. Each of them has distinct data distribution. We denote each source domain as $S_i \in S, i \in \{1, \cdots, n\}$. The training dataset in the each source domain S_i is composed of $S_i = \{x_j^{S_i}, y_j^{S_i}\}_{j=1}^{m_{S_i}}$, where $x_j^{S_i}, y_j^{S_i}$ refers to the jth sample collected from the ith source building, and m_{S_i} refers to the number of samples from the ith source building. Assume the labeled source data S_i is drawn $i.i.d$ from the distribution $\mathcal{D}_{S_i}$. For the target building, we have a set of the collected earthquake-induced building floor vibration signal $\{x_j^T\}_{j=1}^{m_T}$, where m_T refers to the number of instances we need to diagnose for the target domain. Their corresponding true damage labels $\{y_j^T\}_{j=1}^{m_T}$ are unknown. Denote the true data distribution on the target domain as D_T. Assume the unlabeled target sample T is drawn $i.i.d$ from the marginal distribution of $\mathcal{D}_T$ over X,

i.e. $\mathcal{D}_T^X$. Denote the sample space for ith domain as $\mathcal{X}_{S_i}$ where $x_j^{S_i} \in \mathcal{X}_{S_i}$, and damage state space for for ith domain as $\mathcal{Y}_{S_i}$ where $y_j^{S_i} \in \mathcal{Y}_{S_i}$. Similarly, there is $x_j^T \in \mathcal{X}_T$ and $y_j^T \in \mathcal{Y}_T$ for the target domain T. In this work, we assume that $m_T \ll \sum_i^n m_{S_i}$.

Our final goal is to predict $\{y_j^T\}_{j=1}^{m_T}$ without any information about the labels on the target domain. That is, our goal is to build a classifier a classifier $\eta : \mathcal{X}_T \to \mathcal{Y}_T$ with a low target risk :

$$R_{\mathcal{D}_T}(\eta) = Pr_{(x,y)\sim\mathcal{D}_T}(\eta(x) \neq y)$$

without any information about the labels of $\mathcal{D}_T$. In this problem, the data distribution varies between different source domains and from source domains to the target domain, i.e. $P(\mathcal{X}_{S_i}, \mathcal{Y}_{S_i}) \neq P(\mathcal{X}_T, \mathcal{Y}_T) \forall i$. Let $\mathcal{H}$ denote the reproducing kernel Hilbert space (RKHS) for classifiers on $\mathcal{X}$. Define $\mathcal{H}$-divergence between the marginal distributions of the source and target domain on X as $d_{\mathcal{H}}(\mathcal{D}_S^X, \mathcal{D}_T^X)$. Define the *empirical $\mathcal{H}$-divergence* between two samples $S \sim (\mathcal{D}_S^X)^{m_S}$ and $T \sim (\mathcal{D}_T^X)^{m_T}$ as $d_{\mathcal{H}}(S, T)$. $\mathcal{H}$-divergence depicts the distance between two distributions on $\mathcal{H}$. Ben-David et.al proved that the final target classification risk is upper bounded by the combinations of *empirical $\mathcal{H}$-divergence* $\hat{d}_{\mathcal{H}}(S, T)$ and the empirical risk on source domain $R_S(\eta)$ [17].

Theorem 1. ***(Ben-David et al., 2006)*** *Let $\mathcal{H}$ be a hypothesis class of VC dimension d. Given the samples $S \sim (\mathcal{D}_S^X)^m$ and $T \sim (\mathcal{D}_T^X)^m$, with probability $1 - \delta$, for every function $\eta \sim \mathcal{H}$:*

$$R_{\mathcal{D}_T}(\eta) \leq R_S(\eta) + \hat{d}_{\mathcal{H}}(S, T) + \sqrt{\frac{4}{m}(d \log \frac{2em}{d} + \log \frac{4}{\delta})} + 4\sqrt{\frac{1}{m}(d \log \frac{2m}{d} + \log \frac{4}{\delta})} + \beta$$

with $\beta \geq \inf_{\eta^*}[R_S(\eta^\star) + R_{D_T}(\eta^\star)]$.

Extending this theorem to multiple source domains where $S_i \sim (\mathcal{D}_{S_i}^X)^m$ and $T \sim (\mathcal{D}_T^X)^m$, Zhao et.al [243] showed that $\forall \alpha \in \mathbb{R}_+, \sum_i \alpha_i = 1$, with probability $1 - \delta$ and for every function $\eta \sim \mathcal{H}$,

$$R_{\mathcal{D}_T}(\eta) \leq \sum_i \alpha_i \left(R_{S_i}(\eta) + \hat{d}_{\mathcal{H}}(S_i, T) \right) + \beta_\alpha + \mathcal{O}\left(\sqrt{\frac{1}{m}(d \log \frac{m}{d} + \log \frac{k}{\delta})} \right). \tag{3.1}$$

In feature representation-based methods, the empirical risk on source domain $R_{S_i}(\eta)$ depicts the discriminativeness of the extracted features on the source domain S_i, and $\hat{d}_{\mathcal{H}}(S_i, T)$ depicts the domain-invariance of the extracted features from the source data and the target data. To successfully ensure a successful prediction of η on the target domain, we need to minimize the target risk $R_{\mathcal{D}_T}(\eta)$. To minimize the target risk, we need to minimize the source risk $R_{S_i}(\eta)$ and the distribution distance between S_i and T on the input $\mathcal{X}$. That is, to ensure a successful knowledge transfer from multiple source domains to the target domain, we need to ensure the discriminativeness and the domain-invariance of the extracted features. Meanwhile, α_i here represents a convex

combination of the risk upper bound from different source domains.

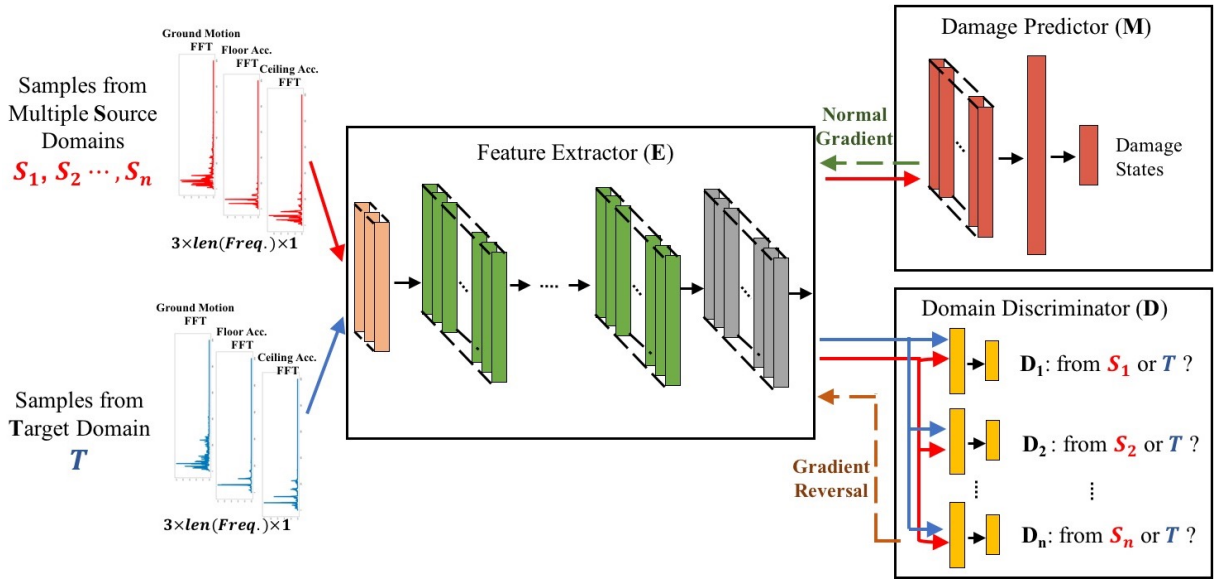


Figure 3.6: Proposed architecture

3.4.2 Data Preprocess

Given the labelled source data $\mathcal{D}_{S_i}$ and unlabelled target data $\{x_j^T\}_{j=1}^{m_T}$, we need to do a preliminary processing of the data and organize the data into a reasonable form for further learning and transferring tasks. The focus of the data preprocessing includes two key steps: 1) extract informative preliminary input features, and 2) data augmentation.

Based on previous characterization in Section 3.3, we also know that the distance between the data distributions of source domains and target domain is related to the earthquakes' and buildings' physical properties. Assume this underlying mapping function is $\mathcal{G}$, there is $d_{\mathcal{H}}(S_i, T) = \mathcal{G}(\mathcal{U}_{S_i}, \mathcal{U}_T, \mathcal{Q}_{S_i}, \mathcal{Q}_T)$, where $\mathcal{U}_{S_i}, \mathcal{U}_T$ represents the corresponding building physical properties and $\mathcal{Q}_{S_i}, \mathcal{Q}_T$ refers to the respective earthquakes' physical properties. It is obviously difficult to model $\mathcal{G}$ if all of these information are unknown. Fortunately, in our problem, though $\mathcal{U}_{S_i}, \mathcal{U}_T$ are mostly unknown, and $\mathcal{Q}_{S_i}, \mathcal{Q}_T$ can be partially depicted by the collected ground motions. By properly combine the ground motion information into the input, we can reduce the problem as to learn the feature representations to depict the correlation $d_{\mathcal{H}}(S_i, T) = \mathcal{G}_{\mathcal{Q}_{S_i}, \mathcal{Q}_T}(\mathcal{U}_{S_i}, \mathcal{U}_T)$, to eliminate the domain changes introduced by the earthquakes.

In conventional methods, people usually use raw floor vibration data or frequency spectrum of floor vibration as input features. This is because they often assume those floor vibrations are collected under the consistent ground motions. Here in our work, to diagnose the damage state of

each story, we combine the information from ground motion, the floor vibration of the story, and the ceiling vibration of the story. We first extract the frequency spectrum data for each vibration signal. Compared to directly using raw vibration data, using frequency domain information has less requirements on the consistent time length of each data sample (which is difficult to achieve in real-world practice), and is more flexible for a wider application. The length of spectrum and frequency resolution should be consistent across the three vibration signals. Then we concatenate the frequency spectrums of the ground motion, the floor vibration and the ceiling vibration according to the respective frequency band. Assume the length of each frequency spectrum is l, we would have each sample's input with size of $3 \times l \times 1$. By aligning the frequency information in this way, the convolutional layers in later adversarial domain adaptation model can better understand the local correlation between ground motion and floor acceleration to extract domain-invariant features. In our experiment, we take the the spectrum with frequency lower than 26Hz, which corresponds to a 1000×1 vector for each vibration signal. We choose lower frequency band since most fatal structural damages are induced by powerful low-frequency seismic wave.

In real-world practice, the size of each domain's dataset varies significantly, from tens to thousands of samples. While training the deep neural network often needs a proper amount of data to better utilize the expressive power of the deep architecture. To avoid over-fitting and better eliminate part of the influence of environmental noise, we conduct data augmentation on the raw datasets. We use varying-length sliding window on the raw vibration signals such that some information before/after earthquake is removed/added, smooth the signals and then conduct the aforementioned process to extract the input for domain adaptation.

3.4.3 Architectures of Feature Extractor, Domain Discriminator, and Damage Predictor

The proposed deep adversarial domain adaptation framework includes 3 main components: feature extractor (E), damage predictor (M), and domain discriminator (D), as Figure 3.6 shows. With the extracted frequency information of structural vibration responses, shown in Section 3.4.2, as input, feature extractor projects the input to a high-dimensional space to obtain the domain-invariant representations h of structural response. The extracted source features h_{S_i} and target features h^T are input to both domain discriminator and label predictor. Damage predictor is used to learn $P(h_{S_i}, \mathcal{Y}_{S_i})$ to predict structural damage states given extracted features. Domain discriminator include n sub-classifiers. The ith sub-classifier focus on distinguish the difference between the distribution of h_{S_i} and h_T. Our objective is 1) to find an optimal feature extractor such the extracted features are domain-invariant as well as discriminative with respect to the source domain labels, and 2) to find an optimal damage predictor with the extracted features as input. To achieve this, the

goal of this framework is to learn a best feature extractor such that even best domain discriminator cannot tell the difference between the extracted features across source and target domains, while ensuring that the damage predictor estimates the structural damage state accurately. In this way, we can find a best trade-off between the domain-invariance and discriminativeness of extracted features, and thus successfully adapt the knowledge learned from other buildings to help diagnose the target building of interest.

The architectures of feature extractor, domain discriminator and damage predictor are based on deep convolutional neural network (CNN). In recent years, deep learning techniques benefit many applications in engineering fields. By constructing neural networks with deep and special architectures, we can approximate a wide range of highly non-linear and complex mapping functions [113]. A variety of deep neural networks have been proved to be effective and powerful in many real-world tasks, including computer vision [109], natural language processing [46], medical imaging [202] and video game [152]. Some typical architectures of deep neural networks include deep convolutional neural network (CNN) [109] and recurrent neural network [148]. Among these architectures, deep convolutional neural network combines convolutional layers to understand the local structures of features in various resolutions and thus becomes very powerful to learn effective representations from complex data.

Table 3.1: Architecture for 5-class damage quantification

Networks	Operation	Kernel	Strides	Feature maps	Activation
Feature Extractor	Input			$3 \times 1000 \times 1$	
	Convolution	$81 \times 5 \times 1$	2×1	$81 \times 499 \times 1$	LeakyReLU
	Convolution	$81 \times 5 \times 1$	2×1	$81 \times 248 \times 1$	LeakyReLU
	Convolution	$81 \times 3 \times 1$	2×1	$81 \times 124 \times 1$	LeakyReLU
	Convolution	$81 \times 3 \times 1$	2×1	$81 \times 61 \times 1$	
Damage Predictor	Convolution	$243 \times 3 \times 1$	2×1	$243 \times 30 \times 1$	LeakyReLU
	Convolution	$81 \times 3 \times 1$	1×1	$81 \times 29 \times 1$	LeakyReLU
	Convolution	$27 \times 3 \times 1$	1×1	$27 \times 28 \times 1$	LeakyReLU
	Flatten			756	
	Full connection			5	Softmax
The ith Domain Discriminator	Flatten			4941	
	Full connection			2	Softmax

Feature Extractor: Feature extractor focus on extracting domain-invariant and discriminative features for all source and target domains. The feature extractor take the concatenated frequency information from the structural responses and respective ground motion as input. With 3 channels including ground motion, floor and ceiling vibration in the input, we first enlarge the number of

channel to discover the detailed local correlation between ground motion's and floor vibrations' frequencies. Then We use multiple convolutional layers with varying kernel size to extract the combinations of frequency energy with varying resolutions. The output of feature extractor is input to both the damage predictor and the domain discriminator.

The optimization of the feature extractor takes the gradient information from both domain discriminator and damage predictor. This optimization is a trade-off between domain-invariance and damage discriminativeness. On the one hand, the output of feature extractor should improve the performance of the damage predictor, on the other hand, the extracted features should be highly domain-invariant such that even optimal domain discriminator cannot distinguish which domain they come from. We use the new designed physics-guided loss function to train the feature extractor, which will be introduced in Section 3.4.4.

Damage Predictor: The damage predictor regularizes the discriminativeness of the extracted features. It is used to predict damage states based on the extracted domain-invariant features, which takes the extracted features as input and classify the samples into different damage states. The basic intuition for the damage predictor is to model the complicate mapping from extracted features to the damage states. Meanwhile, a well-trained damage predictor would backpragate as much sufficient information as possible to the feature extractor, such that the extracted features can have better discriminativeness. When designing the damage predictor, we need ensure the expressive power of the model. However, the damage predictor cannot be too deep either, otherwise the vanishing gradient backpropagated to the feature extractor could no longer provide any valid information about the damage patterns.

Domain Discriminator: Domain discriminator examines the domain-invaraince of the extracted features. Each domain discriminator D_i only takes the extracted features from the source domain S_i and the target domain as input, and classify the samples into 2 classes: the sample comes from source domain S_i or the sample comes from the target domain T. There are multiple source domains. Our goal is to reduce the distribution distance between each source domain and the target domain to extract domain-invariant features. Therefore, we have multiple sub-classifiers inside the domain discriminator. The domain discriminator is trained in an adversarial way to the feature extractor to encourage the domain-invariance of the extracted features. The basic intuition is that, even if the best discriminator cannot distinguish the domain-difference between the extracted features from the source domain and the target domain, the extracted features are mostly domain-invariant.

The architectures may change with different knowledge transfer tasks. For example, if focusing on binary damage detection task, the depth of feature extractor and damage predictor could be reduced since the damage detection task is relatively easier than the damage quantification tasks.

3.4.4 Physics-guided Loss Function for Adversarial Domain Adaptation

To best learn domain-invariant feature representations, we train the feature extractor and discriminator in an adversarial way. The adversarial training is shown to be powerful for training a generative model without labels [65, 213]. The key of the adversarial framework is the loss function design. In our architecture, we have different types of neural networks. Jointly optimizing these non-convex complex networks with unknown landscape is a difficult task. We use a more general notation $\mathcal{F}$ to represent the mapping function constructed by the deep neural networks. We denote $\mathcal{F}_E(\cdot;\theta_e) : \mathcal{X} \to h$ be the feature extractor neural network with parameters θ_e. Similarly, $\mathcal{F}_M(\cdot;\theta_m) : h \to \mathcal{Y}$ refers to damage predictor M with parameters θ_m which map from extracted features to damage states (we assume there are K damage states in total). And $\mathcal{F}_{D_i}(\cdot;\theta_{d_i}) : h \to \mathcal{C}$ refers to the ith domain discriminator D_i with parameters θ_{d_i} mapping from the extracted features to domain labels. D_i only takes extracted features from the source domain S_i and the target domain T. The domain label space $\mathcal{C}$ only have 2 labels: 1 represents the sample comes from the source domain, 0 represents the sample is from the target domain.

In the adversarial framework, the main goal is to regularize the learning of domain-invariant mapping to minimize the distribution distances between the source and target domains. The damage predictor is trained using standard supervised loss function such as cross-entropy function to ensure the model can learn the mapping from extracted feature representations to the damage states. That is, the damage predictor aims at minimizing $R_{S_i}(\eta)$. Each classifier in domain discriminator is a binary classifier. The training of domain discriminator aims to minimize the cross entropy to best classify whether a data point is from the source or target domain. That is, the domain discriminator aims at minimizing $\hat{d}_{\mathcal{H}}(S_i, T)$. The objective of training feature extractor is two-fold: 1) to ensure the discriminativeness, the extracted features should minimize the loss of damage predictor; 2) to ensure the domain-invaraince, the extracted features should maximize the loss of domain discriminator.

Meanwhile, from the Inequation 3.1, we can find that an optimal α_i combinations can approximate the tightest upper bound of the target risk. In previous work, people mostly set $\alpha_i = 1/n$ where n is the number of the source domain [243]. However, we show find that set $\alpha_i = \dfrac{1/\hat{d}_{\mathcal{H}}(S_i, T)}{\sum_i 1/\hat{d}_{\mathcal{H}}(S_i, T)}$ would result in a tighter bound.

Theorem 2. *Given* $R_{S_1}(\eta) = \cdots = R_{S_n}(\eta) \forall i$, $\alpha_i = \dfrac{1/\hat{d}_{\mathcal{H}}(S_i, T)}{\sum_i 1/\hat{d}_{\mathcal{H}}(S_i, T)}$ *is a tighter bound for the target risk* $R_{\mathcal{D}_T}(\eta)$ *than* $\alpha_i = \dfrac{1}{n}$.

Proof. Given that $R_{S_i}(\eta)$ depends the expressive power of the shared damage classifier $\eta =$

$\mathcal{F}_M(\mathcal{F}_E(\cdot;\theta_e);\theta_m)$, we assume that the classifier is optimally trained on the source data and the source risks are equal, i.e. $R_{S_1}(\eta) = \cdots = R_{S_n}(\eta) \forall i$. Therefore, for $\hat{d}_{\mathcal{H}}(S_i, T) > 0$,

$$\begin{aligned}
&\sum_i \frac{1/\hat{d}_{\mathcal{H}}(S_i,T)}{\sum_i 1/\hat{d}_{\mathcal{H}}(S_i,T)} \Big(R_{S_i}(\eta) + \hat{d}_{\mathcal{H}}(S_i,T)\Big) \\
&\qquad = \sum_i \frac{1/\hat{d}_{\mathcal{H}}(S_i,T)}{\sum_i 1/\hat{d}_{\mathcal{H}}(S_i,T)} R_{S_i}(\eta) + \frac{n}{\sum_i 1/\hat{d}_{\mathcal{H}}(S_i,T)} \\
&\qquad \leq R_{S_i}(\eta) + \sum_i \frac{1}{n} \hat{d}_{\mathcal{H}}(S_i,T) \forall i \text{ (based on Cauchy-Schwarz inequality)} \\
&\qquad = \sum_i \frac{1}{n} R_{S_i}(\eta) + \sum_i \frac{1}{n} \hat{d}_{\mathcal{H}}(S_i,T).
\end{aligned}$$

□

In Section 3.3 and 3.4.1, we show that the divergence between two buildings' sample data distribution $\hat{d}_{\mathcal{H}}(S_i, T)$ is related to the similarity between the two building's physical properties $d_{\mathcal{H}_u}(\mathcal{U}_{S_i}, \mathcal{U}_T)$, as discussed in Section 3.3. Therefore, when designing the loss function for the framework, it is necessary to weight the loss of different domain unequally, and the optimal weight for the ith domain,w_i, should have the property that for some non-decreasing function f,

$$w_i \propto \frac{f(1/d_{\mathcal{H}_u}(\mathcal{U}_{S_i}, \mathcal{U}_T))}{\sum_i f(1/d_{\mathcal{H}_u}(\mathcal{U}_{S_i}, \mathcal{U}_T))}. \tag{3.2}$$

Intuitively, the knowledge learned from the similar buildings should be 1) more sufficiently transferred to the target building, and 2) more informative and indicative for the damage prediction on the target building. Therefore, we assign higher weights to the source domains which have more similar physical properties in the loss design. We denote the weight assigned to the ith source domain as w_i where $1 \leq i \leq n$. Denote U_{S_i} and U_T as the known physical knowledge about the ith source building and the target building T. We calculate w_i by taking the reciprocal of the similarity between U_{S_i} and U_T, and then normalize w_i across all source domains using softmax function as

$$w_i = \frac{\exp\left[1/dist(U_{S_i}, U_T)\right]}{\sum_i \exp\left[1/dist(U_{S_i}, U_T)\right]}. \tag{3.3}$$

By using softmax function, we can smooth the influence of the differences between physical properties. As mentioned in Section 3.3, it is a complex process about how the building physical property influences the data distribution, especially when we have very limited physical knowledge. Therefore the physical knowledge should be carefully combined to regularize the domain-invariant

feature extraction. Note that the calculation of the similarities between different buildings' physical properties is quite open depending on how many known parameters that can be obtained (e.g. heights, the first three mode shapes, strong-column/weak-beam ratios). The key is the insight that using physical similarities guides the optimization. For example, in our experiment, we have very limited information about the building besides building height, combining the analysis in Section 3.3, we define the similarity as a function of building heights

$$dist(U_{S_i}, U_T) = (1 - H_{S_i}/H_T)^2 + \epsilon,$$

where H_{S_i} and H_T represents the heights of the source and target buildings, and ϵ is a smoothing factor to avoid $dist = 0$.

We use λ as the factor to tune the trade-off between domain-invariance and discriminativeness of the extracted features. With the weight defined as above, the trade-off the feature extractor, damage predictor and domain discriminator are trained with the below loss function:

$$\min_{\theta_e,\theta_m} \max_{\theta_{d_1},\cdots,\theta_{d_n}} \mathcal{L} = \sum_{i=1}^{n} w_i \mathcal{L}_M^i(\theta_e, \theta_m) - \sum_{i=1}^{n} w_i \mathcal{L}_{D_i}(\theta_e, \theta_{d_i}) \tag{3.4}$$

where

$$\mathcal{L}_M^i(\theta_e, \theta_m) = -\mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_{S_i}} \sum_{k=1}^{K} I(y = k) \log \mathcal{F}_M(\mathcal{F}_E(\mathbf{x})) \tag{3.5}$$

$$\mathcal{L}_{D_i}(\theta_e, \theta_{d_i}) = -\mathbb{E}_{\mathbf{x}_s\sim\mathcal{X}_{S_i}} \left[\log\left(\mathcal{F}_{D_i}(\mathcal{F}_E(\mathbf{x}_s))\right)\right] - \mathbb{E}_{\mathbf{x}_t\sim\mathcal{X}_t} \left[\log\left(1 - \mathcal{F}_{D_i}(\mathcal{F}_E(\mathbf{x}_t))\right)\right] \tag{3.6}$$

$\mathcal{L}_M^i$ represents the cross-entropy loss of using extracted features to predict the damage states for source domain S_i. $\mathcal{L}_{D_i}$ represents the cross-entropy loss of distinguishing the extracted features from S_i or T. This is a minimax problem. The goals between the feature extractor and domain discriminator is opposite, we need to train them in an adversarial manner to find the saddle point $\hat{\theta}_e, \hat{\theta}_m, \hat{\theta}_{d_i}$ such that

$$(\hat{\theta}_e, \hat{\theta}_m) = \arg\min \mathcal{L} \tag{3.7}$$

$$(\hat{\theta}_{d_1}, \cdots, \hat{\theta}_{d_n}) = \arg\max \mathcal{L}. \tag{3.8}$$

In practice, to find a stationary point, the adversarial training is achieved by a trivial gradient-reversal layer. As Figure 3.6 shows, the gradient-reversal layer connects between the feature extractor and the domain discriminator. In the forward-propagation process, the input is not changed. But in back-propagation, it reverses the gradient by multiplying it by a negative scalar during

back-propagation [57]. That is, during back-propagation, the neural networks follow the gradient updates as:

$$\theta_e \leftarrow \theta_e - \delta \sum_i \left(\frac{\partial w_i \mathcal{L}_M^i}{\partial \theta_e} - \frac{\partial w_i \mathcal{L}_{D_i}}{\partial \theta_e} \right) \tag{3.9}$$

$$\theta_m \leftarrow \theta_m - \delta \sum_i \left(\frac{\partial w_i \mathcal{L}_M^i}{\partial \theta_m} \right) \tag{3.10}$$

$$\theta_{d_i} \leftarrow \theta_{d_i} - \delta \left(\frac{\partial w_i \mathcal{L}_{D_i}}{\partial \theta_{d_i}} \right) \tag{3.11}$$

3.5 Evaluation

In this section, we evaluate the proposed framework on both simulation and real-world earthquake-induced building vibration datasets. We first give a brief description about the dataset in Section 3.5.1. Then we describe the baseline methods for comparison in Section 3.5.2 the knowledge transfer performance on simulation data (Section 3.5.3) and real-world dataset (Section 3.5.4). Finally, we characterize the training process and discuss the effect of hyperparameter λ.

3.5.1 Data Description

To evaluate the performance of our algorithm, we first transfer the knowledge between simulation data, and then transfer the knowledge learned from simualtion data to diagnose the real-world structural damages.

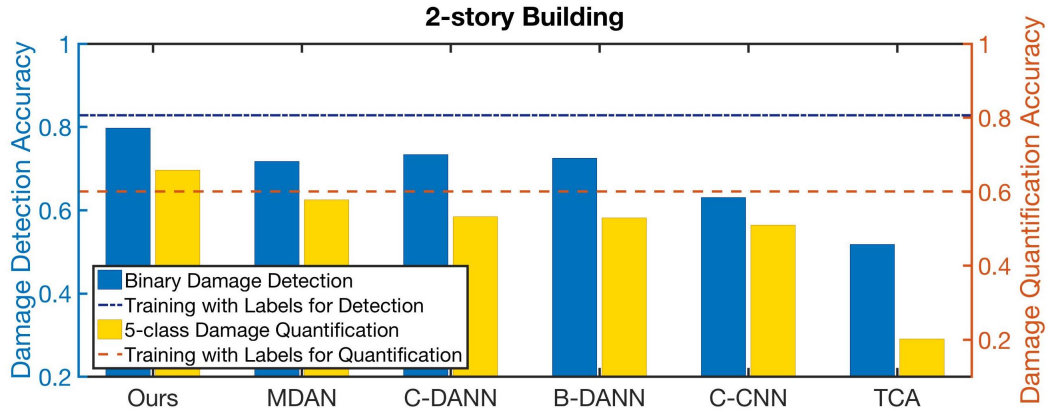
Simulation Data: To understand the building structural damage patterns under earthquakes, people build the building response database using a wide range of archetype steel frame buildings with MRFs [80, 82]. The archetypes we used include 2-story, 4-story, 8-story, 12-story and 20-story building with a first-story height of 4.6m and a typical story height of 4m. The steel MRFs of these archetypes are designed with three strong-column/weak-beam (SCWB) ratios of 1.0. The researchers utilize a suite of ground motions with large moment-magnitude ($6.5 \leq M_w \leq 7$) and short closest-to-fault-rupture distance ($13km < R_{rup} < 40km$). These ground motions are collected from 40 observation stations during 5 previous earthquake events. To simulate the building responses, people implement two-dimensional nonlinear model representations of all the archetype MRFs in the Open System for Earthquake Engineering Simulation (OpenSEES) Platform [82, 146]. Multiple incremental dynamic analysis (IDA) [214] are performed. The floor vibrations and story drift ratios are recorded with sampling frequency of $\sim$200Hz for each ground motion over a wide range of incremental factors.

Real-world data: the data is collected based on a series of shake table tests of a 1:8 scale model for a 4-story steel MRFs conducted at the State University of New York at Buffalo [124]. Researchers conduct a series of the scaled 1994 Northridge earthquake ground motions recorded at the Canoga Park, CA, station. The scale factor ranges from 0.4 to 1.9. Accelerometers and displacement meters, and strain gauges are instrumented on the structure to record the structural responses and story drift ratios $\sim$128Hz.

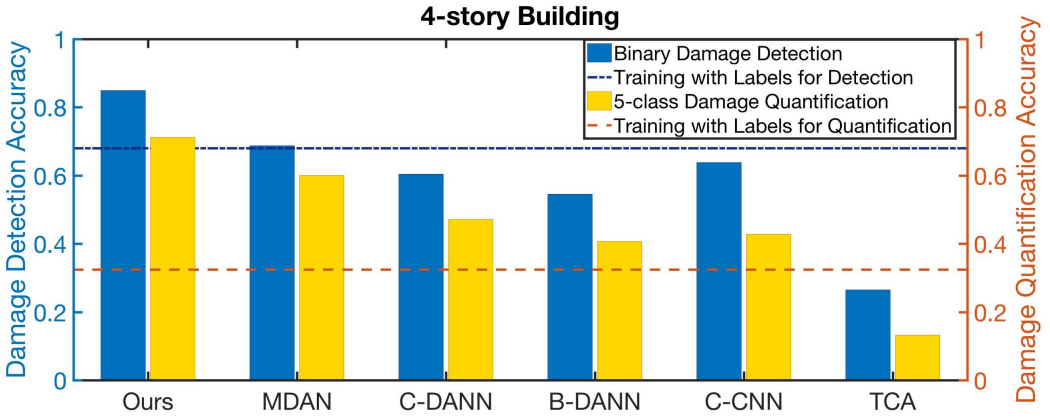
True damage label determination: According to the current practice standard (FEMA P695) [40, 103, 188, 194], structural damage states are defined based on the ground-truth peak story drift ratio at each story. For the damage detection task, we divide the damage state into no damage ($SDR \in [0, 0.01)$) and damaged($SDR \in [0.01, +\infty)$). For the task of damage quantification, there are 5 damage states: no damage ($SDR \in [0, 0.01)$), slight damage ($SDR \in [0.01, 0.02)$), moderate damage ($SDR \in [0.02, 0.03)$), severe damage ($SDR \in [0.03, 0.06)$), and collapse ($SDR \in [0.06, +\infty)$).

3.5.2 Benchmark Methods

To sufficiently evaluate the performance of our framework, we compare our model with 5 different other models, including MDAN [243], C-DANN, B-DANN, C-CNN, and TCA [175]. MDAN is a multiple source adversarial domain adaptation algorithm, which is used here to compare the improvement by our new designed loss. The architecture of MDAN in our experiment is designed as exactly same with the architecture for our proposed framework to ensure the fairness of comparison. C-DANN is to combine the data from all source domains as single source domain dataset, and use single source adversarial domain adaptation method DANN [57] to transfer knowledge from the source to the target domain. B-DANN is to transfer the knowledge from each source domain to the target domain using DANN [57], and select the one with best performance to report. C-DANN and B-DANN have exactly same feature extractor's and damage predictor's architectures with our framework, the only difference is that they only have one classifier in domain discriminator due to single source domain. C-CNN is to directly use deep convolutional neural network to train on the combined source data and predict on the target domain. The architecture for C-CNN is the combination of the architectures of the feature extractor and damage predictor in our framework mentioned in Section 3.4. TCA is a classic two-stage single-domain adaptation method. When implementing TCA on our dataset, we first combine the data from all source domains as single source domain dataset, then use TCA to extract the domain-invariant features for both the source and target domain, and finally use the support vector machine to train on the transformed source data and predict on the transformed target data. Besides, we also evaluate the performance of training on the target domain using deep convolutional neural network.

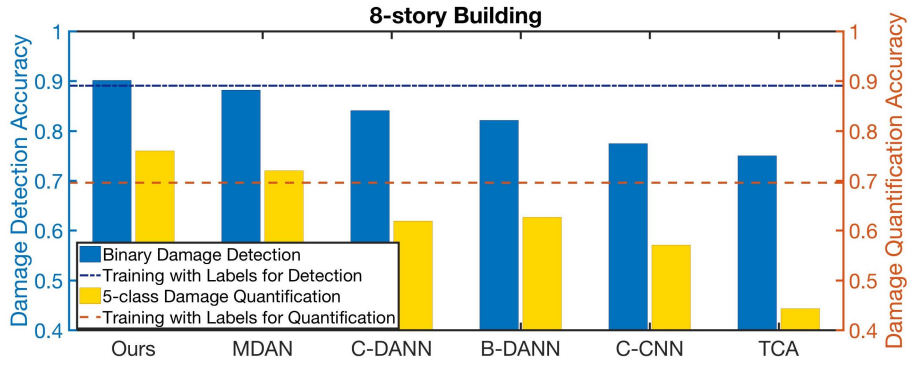


(a)

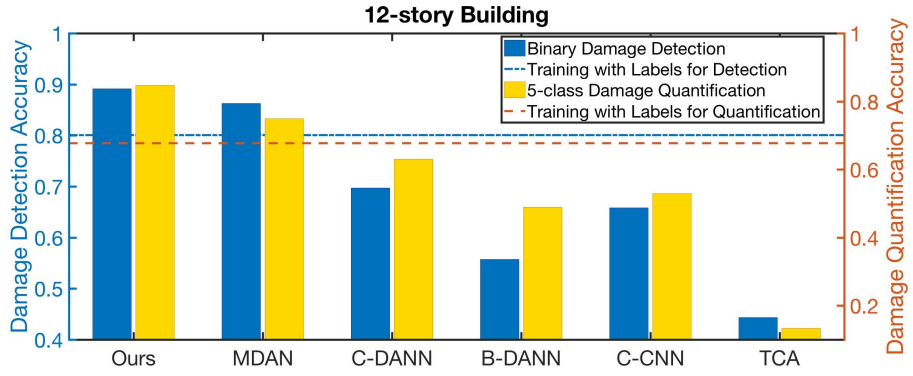


(b)

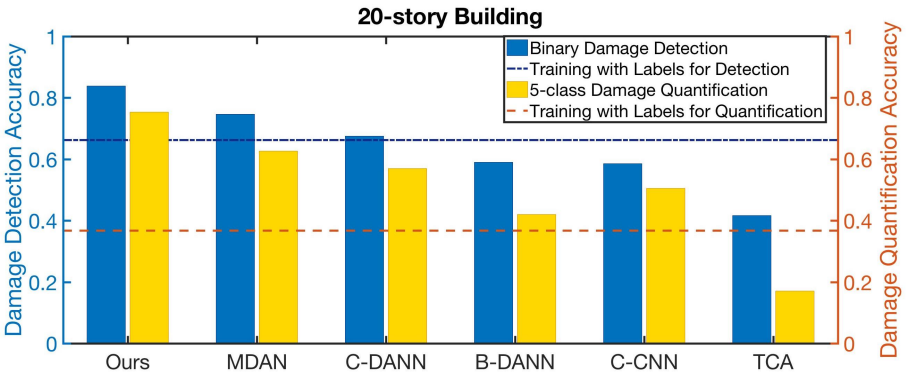
Figure 3.7: (a), (b)compares the performance to transfer knowledge from other buildings to the 2, 4-story buildings by using our method and other approaches. The results include the performance on binary damage detection (blue) and 5-class damage quantification tasks (yellow). We use the dotted line to represent the damage prediction accuracy of directly training on the target domain as reference.



(a)



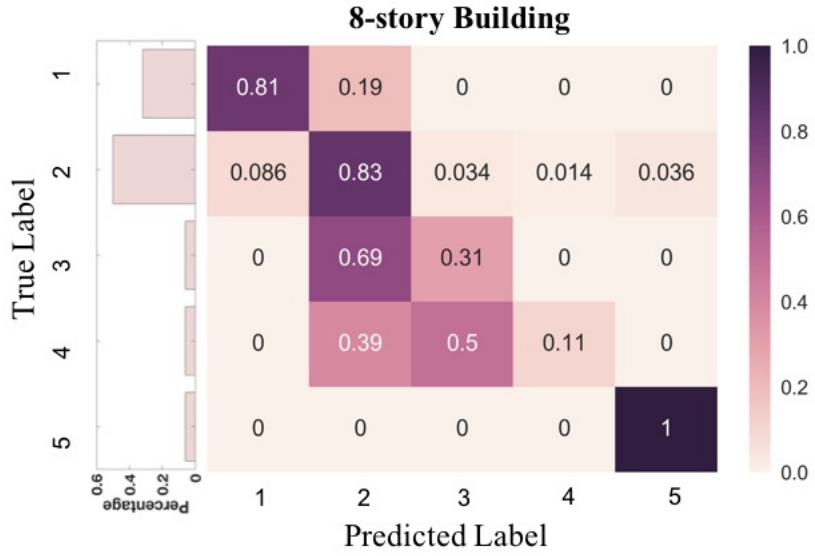
(b)



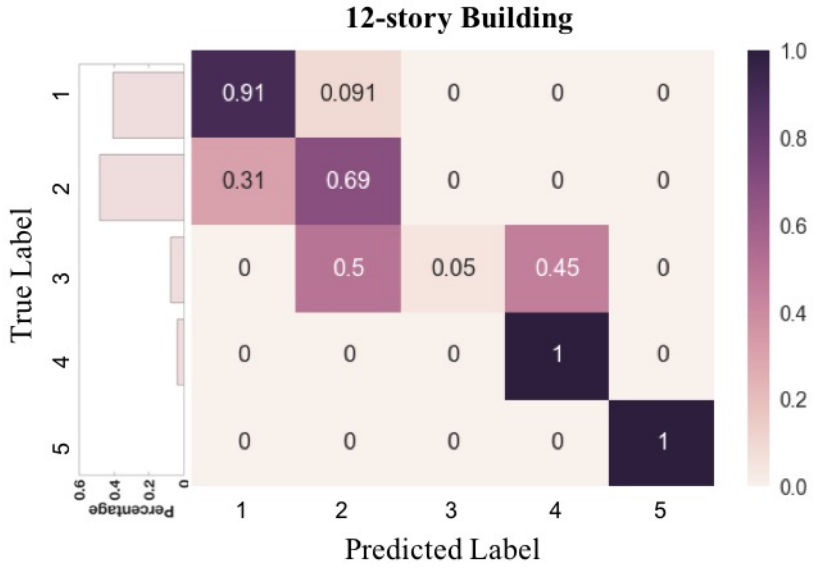
(c)

Figure 3.8: (a), (b), (c) compares the performance to transfer knowledge from other buildings to the 8, 12, 20-story buildings by using our method and other approaches. The results include the performance on binary damage detection (blue) and 5-class damage quantification tasks (yellow). We use the dotted line to represent the damage prediction accuracy of directly training on the target domain as reference.

3.5.3 Knowledge Transfer Across Different Buildings on Simulation Data



(a)



(b)

Figure 3.9: (a) Confusion matrix of the 5-class damage quantification result for knowledge transfer from other buildings to the 8-story building. (b) Confusion matrix of the 5-class damage quantification result for knowledge transfer from other buildings to the 12-story building. The left of each confusion map shows the density histogram of each damage class in the respective building's dataset.

We explore the performance of our method on knowledge transfer across simulation data. Two different tasks, binary damage detection and 5-class damage quantification are conducted to evaluate the performance of knowledge transfer.

To prepare the training and testing dataset for simulation data, we first conduct linear interpolation to align the timestamps between floor vibrations and impute the missing data for each building. We then select the data collected under ground motions ranging from 0.18 to 1.2, which covers most moderate earthquakes as Figure 3.3.1 shows. We also conducted data augmentation as some buildings have very limited structural response datasets. Based on the observations on the data, We take the sliding window length varying between $t-2.5$ seconds to t seconds where t is the time length of the raw vibration signal. We take the stride of sliding the window as 0.25 seconds. The vibration signals with too short ground motion duration are removed. Based on the selected data, we conduct Fast Fourier Transform (FFT) on the ground motion, the floor acceleration and the ceiling acceleration for each story-wise data sample. The peak story drift ratio is simultaneously quantified into damage class as ground-truth label. We organized the data for each building by story level.

We conduct different experiments which transfer knowledge from other buildings to the 2, 4, 8, 12, and 20-story building separately. The domain adaptation is implemented in story level, which is 2nd floor in our experiment. We choose to transfer knowledge across the 2nd floor of different buildings based on 2 reasons: 1) all the 5 buildings have the 2nd floor, and 2) as Figure 3.3.1 and Figure 3.3.1 shows, the data distribution change in lower floor is more complicate than higher level, which is a more difficult task. For the 2nd floor, the data size for 2, 4, 8, 12, 20-story building is 2250, 750, 2200, 1350, and 600, respectively.

We design different architectures for binary damage detection task and 5-class damage quantification task. The architecture for damage quantification is shown in Table 3.1 with $\sim$ 250K parameters. Since damage detection is an easier task, we reduce the number of convolutional layers in feature extractor and damage predictor to 3 and 2 respectively. When training the model, we use Adam as optimizer [102] with the momentum of 0.9 and the weight decay rate of $1e-4$. We take λ as $0.2 \sim 0.5$ for damage quantification and λ as $0.01 \sim 0.05$ for damage detection. The initial learning rate is set as 0.005 for damage quantification and 0.0002 for damage detection with learning decay rate of 0.1.

Figure 3.7 and 3.8 shows the performance of our proposed framework on knowledge transfer across 5 buildings' 2nd floors for different damage diagnosis tasks. Figure 3.5.1, 3.5.1, 3.5.1, 3.5.1 and 3.5.1 show the results of knowledge transfer from other buildings to the 2, 4, 8, 12, 20-story building, respectively. Blue bars refer to the results of binary damage detection and yellow ones represent the results of 5-class damage quantification tasks. We use the dotted line to represent

the damage prediction accuracy of directly training on the target domain as reference. Figure 3.9 shows the confusion matrix of damage quantification results for transferring knowledge to the 8-story and the 12-story buildings. Since we do not have class 5 (collapse) for 12-story building data, we mark it as 1 in the confusion matrix. It can be found that the damage quantification accuracy achieves at 76% and 84.67% for the buildings, respectively. Besides, the ± 1 damage quantification accuracy achieves at 97.41% and 100% respectively. Figure 3.10 show the performance of our framework to transfer knowledge from other buildings to the 12-story building across the 2nd story, 6th story and the roof story. The results on damage detection, damage quantification and ± 1 damage quantification are presented.

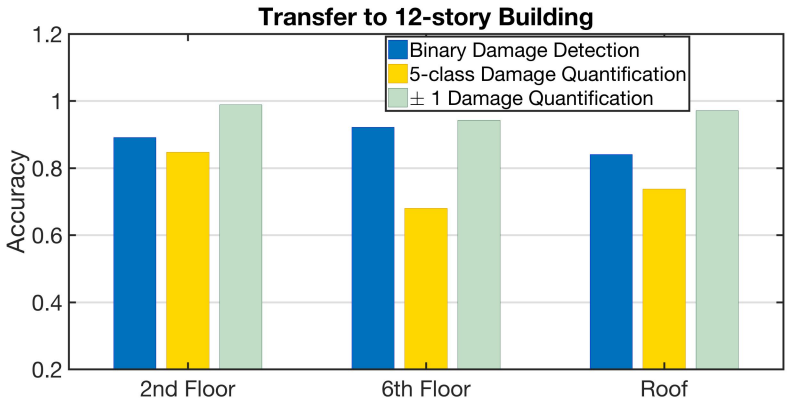


Figure 3.10: Performance of our framework to transfer knowledge from other buildings to the 12-story building across the 2nd story, 6th story and the roof story. The results on damage detection (blue), damage quantification (yellow) and ± 1 damage quantification (green) are presented.

Our framework achieve upto 68.76% improvement on damage detection and 86.37% improvement on damage quantification compared to the benchmark methods other than directly training on the target domain. We have several observations based on the results: 1) The ”no damage” and ”slight damage” data points, the ”moderate damage” and ”severe damage” are two groups difficult to classify. 2) Except transferring to the 2-story building, our method can achieve comparable performance or outperform by directly training on the target domain. This shows that our framework is good at combining the information from different source domains to improve the knowledge transfer. This is because that for some buildings with very few data, the information is too limited for directly training on the target domain. This makes the model easily overfit and reduce the prediction performance. For the 2-story building, the 2nd story is the roof story, which may result in a different damage pattern. 3) For most buildings, multiple source domain adaptation methods outperform than single domain adaptation methods. This is because multiple source domain adaptation methods can utilize information from multiple buildings compared to baseline like B-

DANN and TCA, and can better understand and extract these information compared to C-DANN and C-CNN. 4) Low size of target inputs would limit the performance of knowledge transfer. For example, the 8-story and the 12-story building can achieve relatively higher accuracy rather than the 4-story and the 20-story building. This is because there are too few samples even after data augmentation, few inputs make it difficult for the feature extractor to sufficiently learn the underlying marginal distributions and conduct domain-invariant transform.

3.5.4 Knowledge Transfer Across Different Buildings From Simulation to Real-world Diagnosis

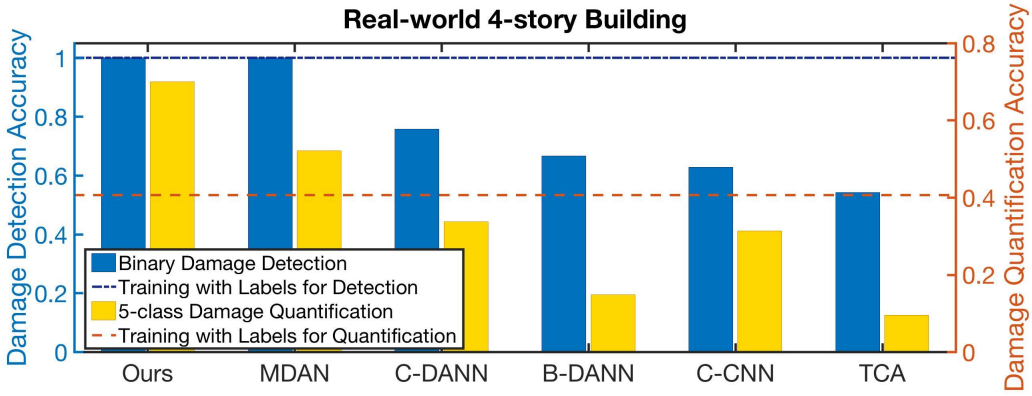


Figure 3.11: This figure compares the domain adaptation performance between our method and other approaches on binary damage detection (blue) and 5-class damage quantification tasks (yellow) on a real-world 4-story building. We use the dotted line to represent the damage prediction accuracy of directly training on the target domain as reference.

Compared to the simulated non-linear building model, real-world structures often have more complicate non-linear load-deformation model. However, real-world seismic structural responses data is often difficult to acquire. Here we would like to validate the potential application to transfer the knowledge from the simulation data to the real-world structure for earthquake-induced structural damage diagnosis.

For the data preprocessing, we follow the steps described for simulation data in Section 3.5.3. The difference is that we have more dense sliding window with small striding to augment the data, since the size of real-world building data is very limited. We finally obtain 1500 data points and respective damage states as real-world structure dataset. Note the damage class distribution is very imbalanced as Figure 3.12 shows, which is common in real-world data. We use the same architecture as Table 3.1 for damage quantification. While for damage detection, a lighter-weight model than the one used for simulation data's damage detection is employed and shows a surprisingly

good performance, which will be discussed later. We use Adam with the momentum of 0.9 and the weight decay rate of $1e-4$ as optimizer [102]. For damage quantification, we found the performance achieves the best when set $\lambda = 1$. We take λ as $0.01 \sim 0.05$ for damage detection. The initial learning rate is set as 0.005 for damage quantification and 0.0002 for damage detection with learning decay rate of 0.1.

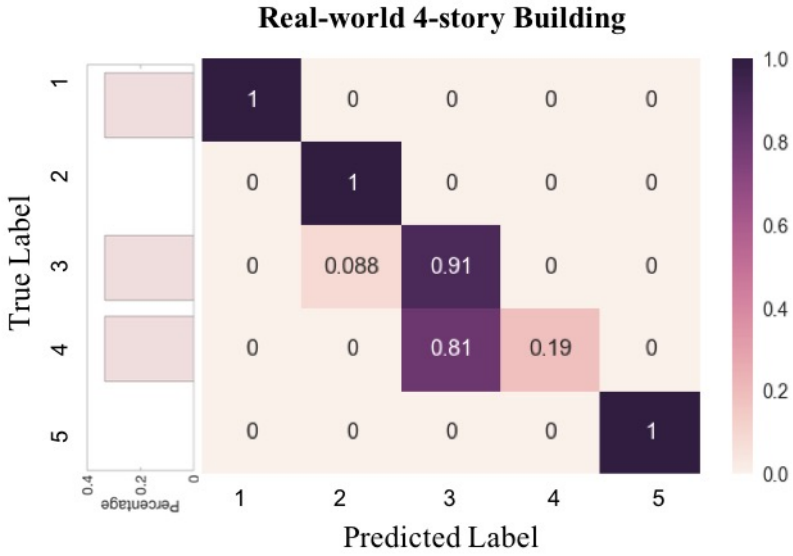


Figure 3.12: Confusion matrix of the 5-class damage quantification result for knowledge transfer from simulation data to a real-world 4-story building. The left of the confusion map shows the density histogram of each damage class.

As a result, our framework can achieve 100% damage detection accuracy and 69.93% 5-class damage quantification accuracy, as shown in Figure 3.11. In the task of damage detection, our framework has the same performance as MDAN, and outperforms by other method, which validates that multiple source domain adaptation methods have their unique advantage on fusing the information from different source domains. On the more difficult task of damage quantification, our method outperforms by other methods, which show the importance of discovering and combining the physical knowledge about the domains into the loss design. Figure 3.12 shows the confusion matrix for the damage quantification result, which indicates a $100\% \pm 1$ damage quantification accuracy. Since we do not have class 3 (moderate damage) and class 5 (collapse), we mark them as 1 in the confusion matrix. Interestingly, the source dataset contains data points indicating moderate damage and collapse, but the well-trained model only misclassifies a small group of data points belonging to moderate damage into slight damage, and avoids to classify any points into collapse.

3.5.5 Characterizing The Training Process and Effect of λ

In this section, we characterize the training process, giving some lessons and experiences about the optimization of the proposed framework. A key problem for adversarial framework is how to find a stationary saddle point in minimax/maximin optimization. For our proposed framework, this problem is more critical and difficult to resolve. Our framework combines three neural network architectures, which makes it more difficult to guarantee a stationary saddle point. Here we visualize the changes of the damage classification loss $\frac{1}{n}\sum_{i=1}^{n} w_i \mathcal{L}_M^i(\theta_e, \theta_m)$ and domain discrimination loss $\frac{1}{n}\sum_{i=1}^{n} w_i \mathcal{L}_{D_i}(\theta_e, \theta_{d_i})$, as Figure 3.13 shows. We can find that both loss keep fluctuating during the training epochs due to adversarial training scheme. The damage prediction accuracy on the target domain varies a lot at the early training stage, and finally converge a stable point. A basic insight to stabilize the training is that, in architecture design, the sub-classifiers in domain discriminator should be kept as light-weight nets but sufficiently powerful to conduct binary classification on the extracted features. However, sometimes it would be difficult to find a saddle point for the training framework. How to stabilize the training of adversarial frameworks needs to be resolved in the future work.

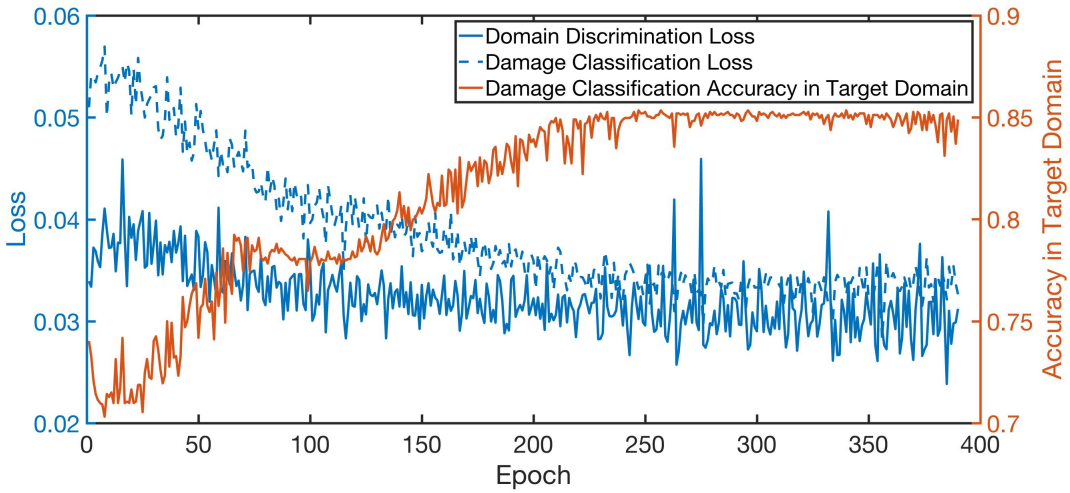


Figure 3.13: The loss of domain discrimination loss, the loss of damage classification on the source domains, and the accuracy of damage classification on the target domain change with training epochs.

The tuning of parameter λ is another key for the network optimization. λ controls the trade-off between the discriminativeness and domain-invariance of the extracted features. In our experiment, we found that different architectures have different optimal λ. If the architecture of damage predictor is more complex, or if we have limited data or more noisy data, we need higher λ to

ensure the domain-invariance of the extracted features.

We also visualized the kernel in the first convolutional layer of learned feature extractor. As Figure 3.14 shows, there are 3 groups of 81 kernels with size of 5×1. The 3 groups focus on extracting information from floor response frequency, ceiling response frequency, and ground motion frequency. We can find that most kernels are active, which means our network parameters are not redundant. It can be found that in some groups, the kernels focus on extracting information in the same frequency band, while in some other groups, the kernels for floor and ceil responses tend to focus on opposite frequency band to the kernel for ground motion information.

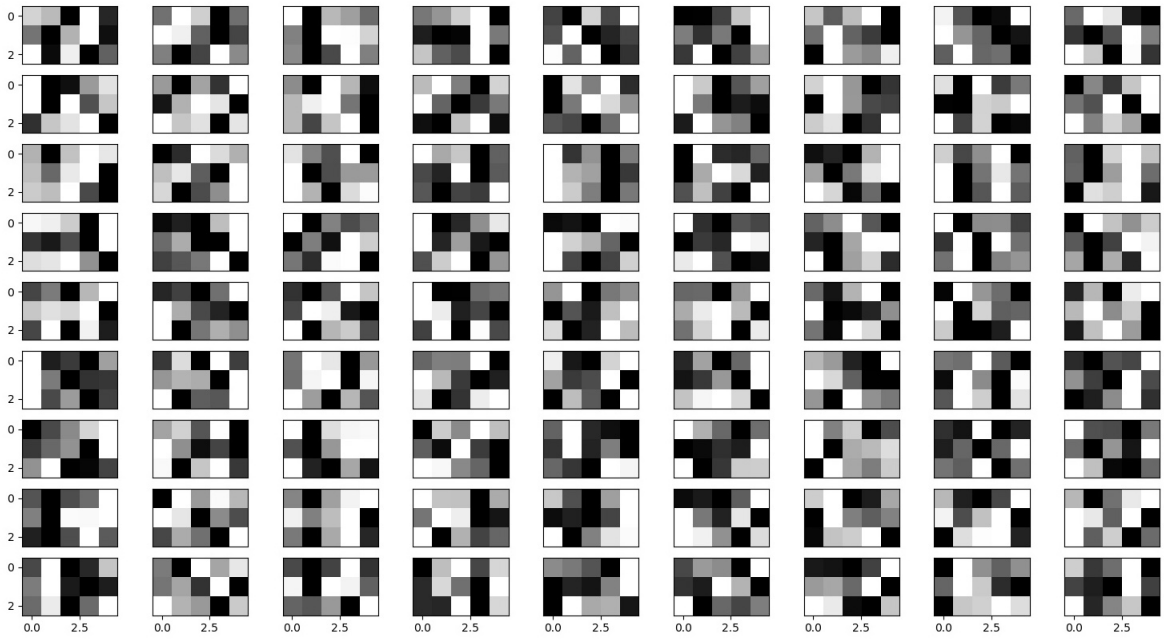


Figure 3.14: Visualized kernel for the first convolutional layer of the learned feature extractor described in the Table 3.1. There are 3 groups of 81 kernels with size of 5×1. The 3 groups focus on extracting information from floor response frequency, ceiling response frequency, and ground motion frequency, which is ordered from top to bottom in the figure.

3.6 Conclusion

In this work, we introduce a new modeling framework to adapt and transfer the knowledge learned from different buildings to diagnose the earthquake-induced structural damages of another building without any labelled data. The proposed adversarial domain adaptation approach extracts domain-invariant and damage-discriminative feature representations of data from different buildings. To

best of our knowledge, this is the first framework to address the multiple source domain adaptation challenge in post-earthquake building damage diagnosis without any labels of the target building. This end-to-end framework integrates data augmentation, input feature extraction, and domain adaptation for damage detection and damage quantification. Besides, this framework is flexible to combine as much available information in historical datasets from different other buildings as possible to help diagnose the current building, which ensures its application practicalities. In this framework, we propose a new input feature embedded with the ground motion information. We design a new physics-guided loss based on fuzzy physical knowledge about buildings to eliminate the uncertainties introduced by those source buildings with less physical similarities to the target building. We prove that this new loss provides a tighter upper bound the damage prediction risk on the target building.

As for evaluation, we evaluate the proposed framework using both simulation data and real-world data, including 5 different buildings under more than 40 earthquakes for simulation and a real-world 4-story building under incremental dynamic analysis. Our method achieves upto 90.13% damage detection accuracy and 84.66% damage quantification accuracy on simulation data. We also successfully transfer the knowledge learned from the simulation data to the real-world building with 100% damage detection accuracy and 69.93% damage quantification accuracy, which outperforms by the state-of-the-arts frameworks.

Chapter 4

An Efficient Sensor Deployment Mechanism in Urban Mobile Crowd Sensing Systems

4.1 Problem Overview

To monitor urban infrastructures, an efficient sensor deployment is important to achieve an optimal quality of sensing coverage and ensure the most informative data is collected from infrastructure for further data analysis with limited budget. In urban infrastructure systems, the sensor nodes are deployed in the static or mobile manner, from city-scale deployment to deployment inside a single infrastructure such as a bridge or a building. One of the most challenging problem is how to achieve efficient sensor deployment in city-scale mobile sensing systems for urban infrastructure monitoring. Compared to static sensor deployment, mobile sensing systems need to account for the mobility of sensor carriers. This makes the problem of optimal sensor deployment more complicated and challenging, especially for a large city scale mobile sensor deployment. As discussed in [136], the mobility of sensor carriers is a double-edged sword. On the one hand, sensor carriers' mobility prevents the cost of deploying many fixed sensors to collect data around a large city. On the other hand, the sensor carriers' stochastic and heterogeneous mobility makes it difficult to guarantee a reliable quality of sensing coverage over time.

In this chapter, we introduce an efficient sensor deployment framework which optimizes the quality of spatio-temporal sensing coverage by incentivizing sensor carriers in urban mobile crowd sensing system. In recent years, mobile crowd sensing system, as a new type of mobile sensing system, has attracted many attentions for urban infrastructure monitoring. Mobile crowd sensing systems can reduce cost and energy consumption by utilizing the low-cost mobile sensors mounted on non-dedicated individual mobile devices [50, 58, 74, 216, 227]. Vehicular crowd sensing systems are a typical example. Mobile sensors are pre-installed on individual vehicles to sense the tar-

get data at different time and locations, which reduces the cost to deploy, manage and maintain the mobile sensor system, and becomes more flexible to various short-term tasks [6, 15, 127, 162, 183]. However, since these vehicles are non-dedicated platform and change their locations dynamically, it is difficult to ensure the sensors carried by vehicles always distribute in an optimal way. This inefficient sensor deployment is a common challenge for these mobile crowd sensing systems. In the following context, we first give a brief introduction of mobile crowd sensing systems, and then present the common challenge of inefficient sensor deployment.

Typically, a vehicular crowd sensing system includes three components: a **data request end**, a **crowdsourcer**, and **vehicle agents**. The **data request end** requests city-wide sensing data from crowdsourcer for future data analysis. When requesting, the data request end also provides the budget and a desired distribution of the collected data, which we call the target (sensing) distribution. Note we name the distribution/density of the collected data in spatio-temporal domain as the "**sensing distribution**" in the rest of the chapter. The target distribution generally depends on the sensing objective of the data request end and consists of the desired information precision in different regions. For example, if the sensing data is collected for general air quality monitoring, the data request end usually expects the collected air pollution data to be uniformly distributed over the city to obtain enough information in different regions for real-time monitoring and forecasting [15, 31, 185]; while when monitoring mobs in a large city, forest fires, factory pollution, or special atmosphere activities during special dates or seasons, the data request end expects to spend the most budget on collecting information in the crowded areas or specific neighborhoods instead of uniformly across the city [7, 100, 119, 180, 239]. With rapidly increasing smart-city applications of vehicular crowdsensing system, the demand of the data request end becomes more diverse, which requires our crowdsourcer to be highly flexible to data requests with various target distributions.

The **vehicle agent** refers to vehicles, e.g. taxis, drones, buses and etc., that have pre-mounted sensors to collect specific types of data at a given sampling frequency while moving through a city. The primary goal of each individual vehicle agent is to finish its original task, e.g. transporting passengers or transporting goods, to make money. Mounting sensors on these non-dedicated vehicles provides a more flexible and cheaper way to collect city-wide data for different application scenarios. For example, taxis with air pollution sensors can monitor city-wide air quality while serving passengers; and delivery drones with cameras can be used for cartography on the way to deliver packages.

The **crowdsourcer** plans sensing data assignments for vehicle agents and organizes the collected data for the data request end. By integrating all data points collected by all pre-mounted vehicle agents during their movement, the crowdsourcer obtains the final sensing dataset. To satisfy

the demand of the data request end, the crowdsourcer needs to 1) optimize the sensing distribution, which ensures the sensing distribution and the target distribution as similar as possible; and 2) be flexible to various desired target distributions of the data request end.

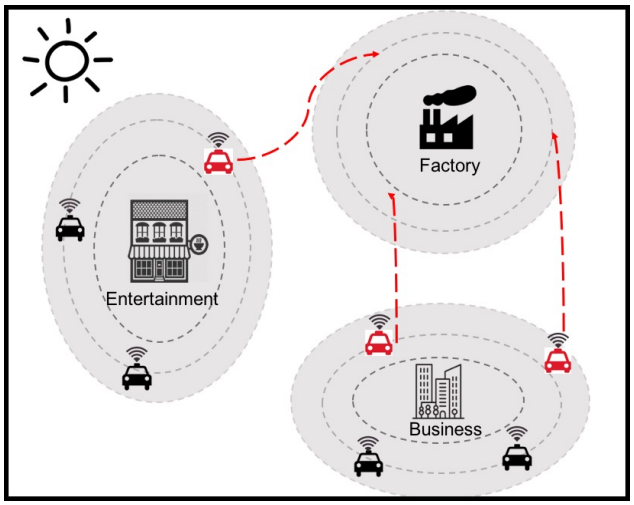


Figure 4.1: Incentivizing vehicle agents to achieve uniform distribution over spatial domain.

However, a vehicle agent such as a taxi has a different goal from that of the crowdsourcer. Such inconsistency of goals often results in a sub-optimal sensor deployment and thus low quality of sensing coverage. The goal of the vehicle agents is to look for more task requests to make money, rather than to sense data, while the crowdsourcer prefers that the taxis distribute themselves according to a target distribution required by the data request end to optimize the sensing quality. For example, to monitor the urban air quality, the data request end needs air pollution data throughout the entire city and across different time intervals to ensure sufficient information for every area [29, 247]. To satisfy the demand of the data request end, the crowdsourcer needs to distribute the air quality measurements uniformly across the city. But taxi drivers spend most time staying in the crowded areas of a city since those areas usually have more ride requests [42, 130]. As a result, few taxis appear in the large non-crowded areas, and the collected air pollution data in these areas is limited. In this case, the sensing system is not able to provide sufficient information about these areas for air pollution monitoring compared to the scenario where taxis distribute uniformly across the city. With the increasing applications of mobile crowdsensing systems, the demand of the data request end may become more and more diverse and not limited to a uniform target distribution [7, 28, 30, 100, 119, 180, 239]. The inconsistency may become more severe when the data request end requests a specifically designed non-uniform distribution. For example, the data request end may request a Gaussian distribution concentrated in factory areas where few taxis pass by. In this case, the difference between the collected data distribution and target distri-

bution is larger than in the case of a uniform target distribution, which will significantly affect the quality of sensing coverage.

A common approach to resolve this problem is to incentivize part of available vehicle agents to new trajectories by offering money, or other forms of non-monetary incentive, e.g. higher probabilities of getting a passenger at the destination, such that the overall distribution of all vehicle agents approximates the target distribution. However, the budget provided by the data request end is often too limited to incentivize all vehicle agents when there is a huge number of vehicle agents.

In this chapter, we aim to design a vehicle agent incentivizing algorithm for a crowdsourcer to optimize the sensing distribution and make it close to the target distribution with a limited budget. To optimize the sensing distribution, the key for the crowdsourcer is to figure out 1) which vehicle agents to incentivize, 2) where the vehicle agents should be incentivized to go, and 3) how much to pay for incentivizing each vehicle agent.

However, there are three **challenges** for this objective. 1) For generic target distributions, the difficulty of selecting vehicle agents and their appropriate incentives increases with the complexity of the target distributions. For example, with a uniform target distribution, one can intuitively attempt to ensure equal numbers of vehicle agents in each location. Most previous studies focus on a uniform distribution. When the target distribution dynamically changes over time and space, however, it becomes difficult to decide how to incentivize these vehicle agents. 2) It is difficult to design an incentive that mitigates the inconsistency of goals between the vehicle agents and the crowdsourcer. On the one hand, the crowdsourcer needs to reduce the monetary cost for each vehicle agent to better utilize the budget. On the other hand, the vehicle agents need enough incentives to ensure at least the same profit from following the specified trajectories. 3) There is a large number of vehicle agents, and the number of their candidate trajectories increases exponentially with the length of time, which makes it impossible to use an exhaustive search to obtain the optimal incentive solution.

To address these challenges, our work introduces a multi-incentive vehicle agent dispatching algorithm. Our algorithm has three major contributions:

- *A novel modeling of the incentivizing problem:* To our best knowledge, we are the first to model the quality of sensing coverage as the *KL-divergence* between the target and sensed data distributions and formulate the sensing coverage optimization problem. We further prove that this formulation is a non-linear multiple-choice knapsack problem, which is NP-complete and impossible to solve in polynomial time.
- *A novel hybrid incentive design to reduce the incentivizing cost*: We design a hybrid incentive for the vehicle agents, which combines the non-monetary incentive of potential task requests at the vehicle agent destination (we call this a "hidden incentive") and the monetary

incentive. This combination of incentives allows us to better utilize the budget by decreasing the average cost of incentivizing one vehicle agent.

- *A novel and efficient algorithm to compute optimal the incentivizing strategy*: We introduce the algorithm ***iLOCuS***, which **i**ncentivizes vehic**L**es to **O**ptimize the sensing distribution in a **C**rowd **S**ensing system. The algorithm finds the solution to reduce the dissimilarity in a more efficient way than exhaustive search by a two-stage optimization method.

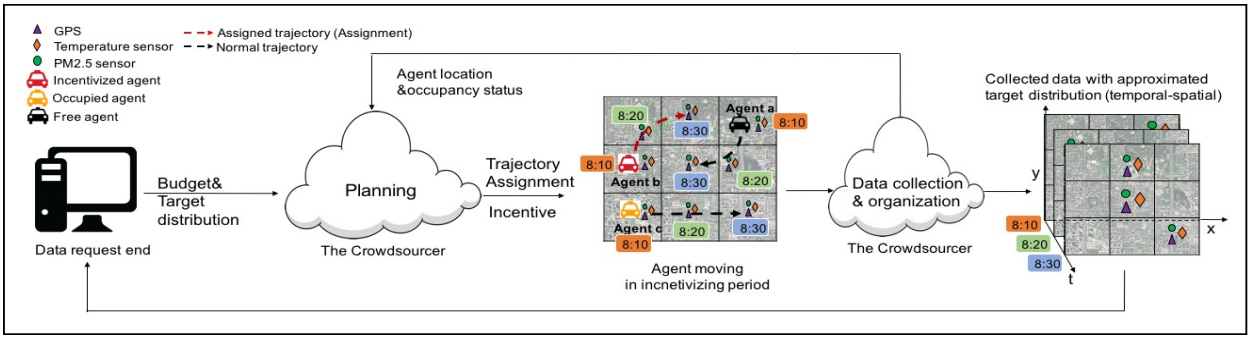


Figure 4.2: The diagram of the considered vehicular mobile crowd sensing system.

The rest of the chapter is organized as follows: Section 4.2 introduces related work in optimizing the sensing distribution in mobile sensing networks. Section 4.3 formulates the problem. Section 4.4 proposes an optimization algorithm to solve the formulated problem. Section 4.5 evaluates the proposed problem formulation and algorithm in simulation data. Section 4.6 summarizes the results and concludes. The Appendix provides the proof of the theorems proposed in the chapter.

4.2 Related Work

We outline the related work on spatio-temporal quality of sensing coverage optimization in mobile sensing networks.

In a mobile sensing system, the quality of sensing coverage refers to a combination of the data coverage, i.e., how many spatial grids and time spots the collected data covers [32, 86, 136, 235], and the data balance level, i.e. whether the collected data has a similar distribution to a given target distribution in the spatio-temporal domain [91, 206]. To achieve a good quality of sensing coverage with a limited budget, most previous works in mobile crowd sensing select a subset of vehicle agents to collect long-term sensory data in their current locations [86, 129, 235], or select dispatching destinations for vehicle agents [206, 219, 235]. These methods mainly focus on static

distributions of sensors at some time point, but ignore the influences of the vehicle agents' dynamic mobility on the quality of sensing coverage.

Recently, other works have begun to use the predictable vehicle mobility to improve the vehicle selection in mobile crowd sensing scenario [1, 74, 91, 99, 216]. There are various settings, assumptions, and objectives for these works. For example, [74] discussed how to incentivize vehicles to maximize the total number of covered regions in all time slots or the covered time length in all regions by selecting several vehicles with predicted mobility. [216] minimizes the incentivizing cost considering probabilistic and deterministic mobility models. [100] aims at identifying the important vehicles for the whole network based on their historical mobility patterns, but without considering the quality of spatio-temporal sensing coverage. [91] proposed a framework to optimize the sensing quality in the spatial domain, which assumes the mobility of each user is known and deterministic, and that all users volunteer to sense data without any incentive reward. [1] proposed a reputation-aware framework considering the vehicle availability to select vehicles that achieve target spatial coverage with budget constraints.

Our setup and objective are, in some aspects, different from previous work. In our work, sensors are already pre-mounted on vehicle agents [29, 45, 138] to make it more convenient to collect data, especially for driverless vehicle agents. For the objective, instead of only focusing on spatial coverage or temporal coverage, we aim at the joint spatio-temporal sensing distribution. With this objective, the crowdsourcer better controls the precision of sensing distribution in both time and spatial domains to fulfill the data request end's demand. Meanwhile, instead of directly optimizing the coverage, we make the collected sensing data distribution as similar to the target distribution as possible. In this way, our system is more efficient and flexible to the various demands of data request end on the target distribution. During incentivizing, we not only select part of vehicles from all equipped vehicles, but also decide trajectories for these selected vehicles. This is quite different from previous works which only have vehicle selection but no trajectory selection. Since we jointly optimize the spatio-temporal sensing distribution, every location that the vehicle agents pass by matters in our objective function. On the one hand, being able to select trajectories for some vehicle agents makes it more flexible to incentivize vehicle agents to different locations and achieve better sensing coverage. On the other hand, the trajectory selection makes the problem more computationally complex. This is because we need to select the best trajectory for each vehicle from a huge number of candidate trajectories.

As for the incentive design, many incentivizing mechanisms are proposed based on auction and game-theoretical models, such as reverse auction [118], Stackelberg game [226] and other budget-feasible mechanisms [8, 87, 225, 250]. [167] summarized and compared different types of incentivizing mechanisms for mobile crowdsensing systems. Generally the incentive could be

monetary or non-monetary reward. A rule of the thumb of incentive designing is to ensure the value of the incentive is not less than the cost of vehicle agent implementing the sensing assignments.

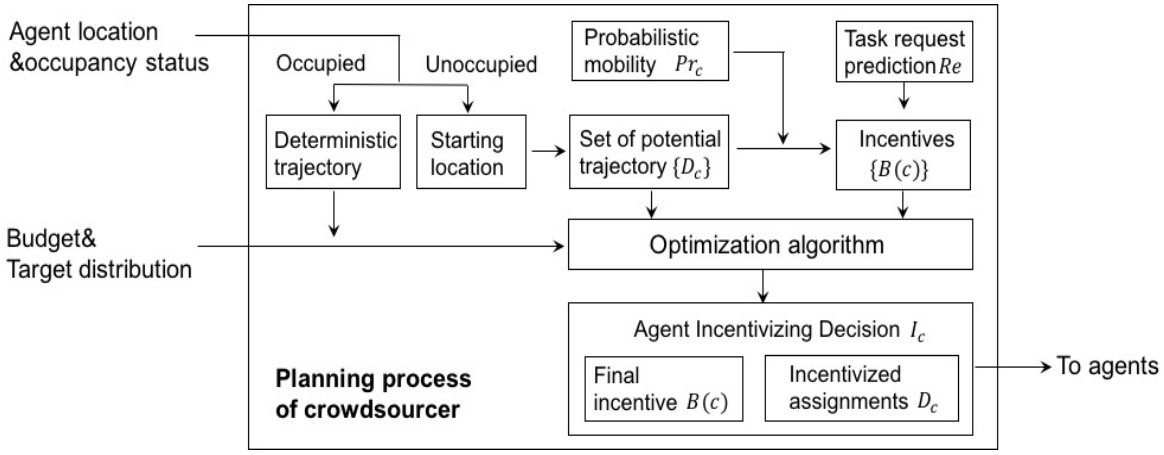


Figure 4.3: The flowchart of crowdsourcer's planning process, which is part of the system described in Figure 4.2.

4.3 Problem Formulation

Our goal is to incentivize taxi mobility so as to match the collected data distribution to the target distribution with a limited budget and a limited number of vehicle agents. We first define key components of this mobile crowd sensing problem in Section 4.3.1. Section 4.3.2 introduces the objective function in detail, which is applicable to various target distributions. Then we describe the design of customized incentives by combining non-monetary rewards to reduce the monetary cost of incentivizing vehicle agents in Section 4.3.3. Finally, we formulate the problem with physical mobility and budget constraints in Section 4.3.4.

4.3.1 Background and Definitions

In this section, we define key components of our mobile crowd sensing system, as shown in Figure 4.2. To simplify the problem, the map of a city is discretized into $a \times b$ grids according to the size of the target area and the desired spatial granularity specified by data request end. Time is also discretized into T time slots, where T is the incentivizing period as defined in Section 4.3.1. We denote the grid locations as (i, j), where $1 \leq i \leq a$, $1 \leq j \leq b$, and the current time point is $1 \leq t \leq T$. All vehicle agents are installed with sensors and assumed to run inside the $a \times b$ map region. The data request end provides the budget, target distribution, and other requirements

to the crowdsourcer. According to location and occupancy information, the crowdsourcer selects vehicle agents to incentivize and plans trajectories for them. During a specific time period, vehicle agents move inside the target region, either with their usual mobility patterns or following the crowdsourcer's specified trajectory. Data is automatically collected along with the traces of the vehicle agents. The crowdsourcer collects and organizes the uploaded data from all vehicle agents and sends the data to the data request end for further analysis.

Data request end

The data request end requests and analyzes the data. The crowd sensing system serves the needs of the data request end. The data request end provides its requirements to the crowdsourcer: the incentivizing period T, the budget B for incentivizing vehicle agents during T, and the target distribution $O(i, j, t)$. Finally, the crowdsourcer returns the crowdsensed data back to the data request end. The crowdsensed data is collected during T without exceeding budget B, and its distribution over time and space is compared to $O(i, j, t)$. If the data request end needs multiple incentivizing periods, it should specify the respective budget and target distributions for each period.

Incentivizing period: denoted as T. At the beginning of each incentivizing period, the crowdsourcer plans and assigns the incentivizing strategies for the next T time points. The length of the incentivizing period indicates how frequently we choose to incentivize a set of vehicles and should be chosen appropriately. If it is too long, the accumulative error of mobility prediction will increase with time and affect our algorithm's performance. If it is too short, it will consume intensive computational resources. If the data request end would like data collected for a longer time span, we can directly incentivize multiple Ts.

Budget: denoted as B, refers to the total amount of money provided by the data request end to incentivize vehicle agents during one incentivizing period.

Target distribution: refers to the desired/expected distribution of data collected over time and space. The target distribution, denoted as O, is a distribution over time and space. $O(i, j, t)$ refers to the percentage of sensing data collected in the location (i, j) at the time point t. Thus, we must have $\sum_{i=1}^{a} \sum_{j=1}^{b} \sum_{t=1}^{T} O(i, j, t) = 1$. The target distribution varies according to the goal of the data analysis. For example, monitoring city-wide air quality requires air pollution data from all regions of the city, and thus requires that the collected data be distributed uniformly over space and time.

Crowdsourcer

The crowdsourcer incentivizes vehicle agents based on the provided information from the data request end and current status of each vehicle agent within the incentivizing period. Figure 4.3 shows the details of the crowdsourcer's planning process. The crowdsourcer takes as input location

and occupancy information from the vehicle agents as well as the budget and target distribution from the data request end. The planning process of the crowdsourcer includes three steps: 1) selecting incentivized vehicle agents, where I_c denotes a binary decision to incentivize vehicle agent c, 2) specifying the incentivized trajectory D_c, which will be introduced in Section 4.3.1, for each incentivized vehicle agent c, and 3) designing the customized incentive $B(c)$ to give each incentivized vehicle agent c according to its assignment. After the planning process, if the selected vehicle agents accept the incentive, they move according to the assignment. Meanwhile, no matter whether the vehicle is selected or not, the pre-mounted sensors on the vehicle will automatically collect and upload the sensing data. Finally, the crowdsourcer will organize the collected sensing data from all vehicle agents within an incentivizing period, and send it to the data request end.

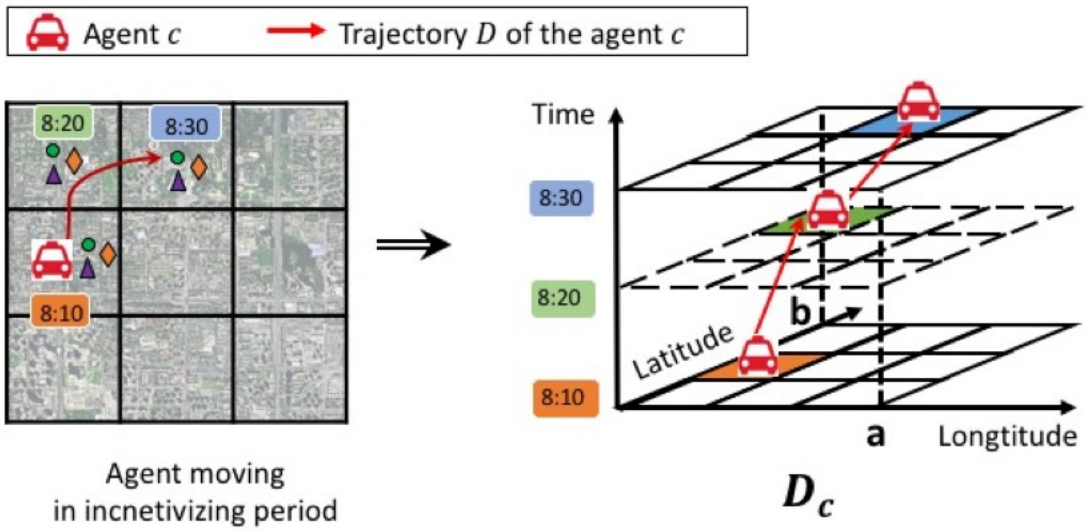


Figure 4.4: The 3-D matrix D_c, which represents the trajectory D of vehicle agent c.

Vehicle agent

Vehicle agent refers to an individual vehicle with sensors pre-mounted. The sensors generally include a GPS sensor, an occupancy sensor, and sensors collecting requested information (e.g., air pollutant sensors). The GPS sensor collects location information about the vehicle agent at each time point. The occupancy sensor shows whether the vehicle agent is unoccupied so that crowdsourcer can incentivize it. Since the sensors are pre-mounted on the vehicle agents, it is reasonable to assume that there is no new vehicle entering the system in the incentivizing period.

Status of the vehicle agent: At the beginning of each incentivizing period, all vehicle agents have two possible statuses: unoccupied or occupied. If a vehicle agent is completing an original task at a given time, we mark its status as "occupied" and not available for further incentivizing

in its occupied duration. Otherwise, we mark the status of the vehicle agent as "unoccupied". After the incentivizing assignments are distributed, vehicle agents have three possible statuses: free, incentivized or occupied. "Incentivized" vehicle agents follow the incentivized trajectories distributed by the crowdsourcer. "Free" vehicle agents are neither occupied nor incentivized, and cruise according to their usual mobility patterns. Note that since sensors are already installed on the vehicle agents, information is still being collected when the vehicle agent is "occupied" or "free"; however, for an "occupied" or "free" vehicle agent, where data is collected as determined by its original task instead of the crowdsourcer.

Trajectory: denoted as D_c, refers to the mobility of a vehicle agent c during the current incentivizing period. As Figure 4.4 shows, D_c is a $a \times b \times T$ matrix. Each element of D_c, $0 \leq D_c(i,j,t) \leq 1$, represents the probability that the vehicle agent c appears in (i,j) at time t, where $\sum_{i,j} D_c(i,j,t) = 1$. If $D_c(i,j,t) = 1$, the agent c has deterministic mobility at time t. At the beginning of the incentivizing period, when the crowdsourcer plans the incentivizing strategies, there are two different types of trajectory from the perspective of the crowdsourcer: original trajectory for non-incentivized vehicle agents including occupied and free vehicle agent, incentivized trajectory for incentivized vehicle agents.

- *Original trajectory:* represents the original trajectory of the non-incentivized vehicle agents. Non-incentivized vehicle agents include "occupied" vehicle agents and "free" vehicle agents. The original trajectory of an "occupied" vehicle agent is known to the crowdsourcer, since it is determined by the vehicle agent's original task and reported to crowdsourcer when the task begins. For the "free" vehicle agent which is neither occupied nor incentivized, its original trajectory is probabilistic for the crowdsourcer at the beginning of T. In a general 2-D map grid, there are 9 potential directions that the free vehicle agent can move to in the next time point, which is the adjacent 8 grids and current grid itself (staying in the current location). So the mobility of "free" vehicle agents are stochastic from the perspective of the crowdsourcer. The mobility can be learnt from historical mobility data [137, 184]. So a probabilistic mobility prediction model is employed to predict $Pr_c(i,j,t)$, which refers to the probability that a **free** vehicle agent c appears in the location of (i,j) at time of t. Some empirical mobility prediction models include Markov Chain[27, 120, 156].
- *Incentivized trajectory:* denoted as D_c, refers to the trajectory that crowdsourcer assigns to the incentivized vehicle agent c during T. The assignment is designed by the crowdsourcer so as to achieve the target distribution of the collected data. If a vehicle agent accepts the assignment and respective incentive, it becomes "incentivized" and is not allowed to take tasks during the incentivizing period. The incentivized trajectory of each incentivized vehicle agent c is usually selected from the set of candidate trajectories $\{D_c^{k_c} : k_c \in \{1, \cdots, K_c\}\}$

given K_c deterministic candidate trajectories.

Task request distribution: denoted as $Re(i, j, t)$, refers to the probability that one vehicle agent could obtain at least one task request at a given spatial location (i, j) and time t. It is approximated by the ratio of task request number over the number of unoccupied vehicle agents inside the grid (i, j) at t. The task request probability is marked as 1 if the ratio is higher than 1. Re can be learned and predicted from historical task request data by applying models discussed in [110].

4.3.2 Objective Function: Dissimilarity between Collected Data Distribution and Target Distribution

The objective of our crowd sensing system is to optimize the sensing distribution such that sufficient information is collected at each time and location. To achieve the objective, in this study, we aim to incentivize the vehicle agents to make the sensing distribution achieve a given target distribution. However, the target distribution may differ according to the demand of data request end. To make the model generalized for heterogeneous target distributions, we define the quality of sensing distribution as how similar the collected data distribution is to the provided target distribution. Therefore, to optimize the quality, we need to reduce the dissimilarity between these two distributions. We measure the dissimilarity using *Kullback-Leibler divergence* from the the collected data distribution to target distribution [110].

In the context of Bayesian inference, the *Kullback-Leibler divergence* from a distribution O to a distribution P, $KL(P||O)$, is a measure of the change of information when one revises beliefs from the prior probability distribution O to the posterior probability distribution P [110]. Optimizing the sensing distribution means minimizing the information changes from the target distribution O to sensing distribution P, which is minimizing the Kullback-Leibler divergence.

In our problem, the target distribution $O(i, j, t)$ over time t and space (i, j) is provided by the data request end. The collected data distribution $P(i, j, t)$ is obtained by integrating all vehicle agents' trajectories in the spatio-temporal domain. Without the loss of generality, we assume all vehicle agents have the same sampling frequency, which is 1 data point per time point per vehicle agent. With all vehicle agents' trajectories $\{D_c : c \in \{1, \cdots, C\}\}$, the amount of collected sensing data at location (i, j) at t is $\sum_{c=1}^{C} D_c(i, j, t)$. The total amount of collected sensing data by all vehicle agents during the whole incentivizing period T is CT. Therefore, given C vehicle agents, the collected sensing data distribution $P(i, j, t)$, which is also the the density of vehicle agents at grid (i, j) at time point t, is calculated as

$$P(i, j, t) = \frac{\sum_{c=1}^{C} D_c(i, j, t)}{CT}.$$

Table 4.1: Mathematical Notation

Symbol	Descriptions of Notations
$t \in \{1, \cdots, T\}$	tth time unit to collect the data, T time units in one incentivizing period
(i, j)	spatial location where $i \in \{1, \cdots, a\}, j \in \{1, \cdots, b\}$
$c \in \{1, \cdots, C\}$	the cth car in all C cars
P	sensing data distribution over the map grid during the incentivizing period collected by all vehicle agents, with dimension of $a \times b \times T$.
O	target data distribution over the map grid during the incentivizing period, with of $a \times b \times T$.
I_c	a binary indicator showing the vehicle is incentivized or not.
K_c	the number of all deterministic candidate trajectories for vehicle agent c.
k_c	the k_cth trajectory of c, $k_c \in \{0, \cdots, K_c\}$, $k_c = 0$ is the probabilistic trajectory when c cruises without incentivizing or passengers.
D_c	the trajectory of the vehicle agent c, a $a \times b \times T$ matrix, also noted as $D_c^{k_c}$ to distinguish the k_cth candidate trajectories of c.
$B(c)$	expected budget for the cth car to be incentivized to its trajectory D_c.
Re	forecasted task request distribution over time and space in incentivizing period, with dimension of $a \times b \times T$.
Pr_c	mobility prediction of vehicle agent c over time and space in incentivizing period, with the dimension of $a \times b \times T$.
$V(c, D_c^{k_1}, D_c^{k_2})$	the decreasing of *KL-divergence* after switching c from the k_1th to the k_2th trajectory.

With the target distribution and collected sensing data distribution, the quality of sensing distribution is defined as the negative of the Kullback-Leibler divergence of P from O:

$$-KL(P||O) = \sum_{i,j,t} P(i,j,t) \log \frac{O(i,j,t)}{P(i,j,t)},$$

Therefore, optimizing the sensing distribution is equivalent to minimizing $KL(P||O)$.

For example, when the target distribution is a uniform distribution, which has probability mass function $O(i,j,t) = const$, the quality of the sensing distribution can be simplified as

$$\begin{aligned} -KL(P||O) &= \sum_{i,j,t} P(i,j,t) \log \frac{const}{P(i,j,t)} \\ &= \log const - \sum_{i,j,t} P(i,j,t) \log P(i,j,t) \end{aligned}$$

The second term $-\sum_{i,j,t} P(i,j,t) \log P(i,j,t)$ is the entropy of the collected data distribution. Previous work [91] utilizes the entropy to evaluate whether the collected data matches the target uniform distribution. Thus, our definition of the sensing distribution quality matches this previous work when the target distribution is uniform.

Furthermore, as opposed to entropy, our objective function, *KL-divergence*, directly measures the dissimilarity between the target distribution and collected sensing data distribution, and thus is more generally applicable to more complex target distributions such as Gaussian mixture distribution, etc.

4.3.3 Customized Incentives

Our incentivizing system assigns incentives to each vehicle agent to ensure that the vehicle agent is willing to execute the assignments while the total amount of incentives stays within the budget limit. We define the vehicle agent utility as the expected future revenue. Given the incentivizing assignment D_c, the incentive $B(c)$ should cover the possible loss of utility induced by switching from rejecting D_c to accepting D_c. Since the vehicle agents always tend to maximize their utilities, this ensures that they accept the incentives.

Our incentive design's key idea is to incorporate the probability of getting a new task request in the destination of the vehicle agent. Since the primary objective of vehicle agents is to search for potential tasks, if the assigned trajectory brings the vehicle agent to a destination with more tasks compared to the vehicle agent's original trajectory, this improvement is an additional hidden incentive to motivate the vehicle agents accepting the assignment.

We define $r_{\max}$ as the utility from finishing the original task within the incentivizing period

of T. The $r_{\max}$ may change over time, since in practice the price of finishing a task may change according to the weather conditions or time of the day. We denote $r_u = r_{\max}/T$ as the utility per time point. We assume r_u and $r_{\max}$ are constant during one incentivizing period T given a short T (e.g. 10 min). Meanwhile, for all vehicle agents, there exists a lower bound $r_{\min}$ for the incentive such that the incentive is not too small to be negligible for vehicle agents. Here we design the incentive $B(c)$ to incentivize the vehicle agent c accepting the assignment D_c as

$$B(c) = \max(r_{\min}, \min(r_{\max}, r_{\max} - r_u(R^c_{ctrl} - R^c_{rand}))). \tag{4.1}$$

$$R^c_{rand} = \sum_{i,j}^{a,b} Re(i,j,T) Pr_c(i,j,T)$$
$$R^c_{ctrl} = \sum_{i,j}^{a,b} Re(i,j,T) D_c(i,j,T)$$

The task request distribution Re and probabilistic mobility distribution Pr_c are both distributions over the spatio-temporal domain. R^c_{rand} is the expected task request that vehicle agent c could obtain at T by following her/his original trajectory. R^c_{ctrl} is the expected task request that c can obtain in the destination of incentivized trajectory D_c at T.

$B(c)$ is in the range of $[r_{\min}, r_{\max}]$. If the incentivizing assignment D_c helps the vehicle agent c find more task requests, we will take this improvement of task request probability as the hidden incentive and pay less than $r_{\max}$. Otherwise we pay the vehicle agent as much as the utility obtained from original task.

Theorem 3. *$B(c)$ always ensures that utility-maximizing vehicle agents are willing to accept the incentivizing assignment.*

We proved that when incorporating the hidden incentive, the overall utility of the vehicle agent c accepting the incentivizing assignment is larger than the utility of the vehicle agent c rejecting the assignment and running by herself/himself. The proof of Theorem 3 is shown in Appendix A. We also note that $B(c)$ is the minimum incentive that satisfies this property; thus, it is the minimum incentive we can offer rational vehicle agents while ensuring that they will accept.

4.3.4 Putting It Together: Formulation of The Vehicle Incentivizing Problem In Crowd Sensing Systems

We formalize our incentivizing policy as the solution to a minimization problem by considering each vehicle agent's physical mobility constraints and the overall budget constraint. For

$c \in \{1, \cdots, C\}, t \in \{1, \cdots, T\}, i \in \{1, \cdots, a\}, j \in \{1, \cdots, b\}$, given the target distribution O and budget limit B, our problem is to decide I_c: whether to incentivize vehicles from C, and k_c: the k_cth candidate trajectory selected to be assigned to incentivized vehicle agent c for sensing data collection, such that

$$\min_{\substack{I_1,\cdots,I_C \\ k_1,\cdots,k_C}} KL(P||O) \tag{4.2}$$

$$\text{subject to } P(i,j,t) = \frac{\sum_{c=1}^{C} D_c(i,j,t)}{CT} \tag{4.3}$$

$$\sum_{c=1}^{C} B(c) \cdot I_c \leq B \tag{4.4}$$

$$D_c(i,j,t) \cdot I_c \in \{0,1\}\ . \tag{4.5}$$

$$D_c = D_c^{k_c} \text{ where } k_c \in \{0, 1, \cdots, K_c\} \tag{4.6}$$

Note that $B(c)$ is determined by I_c and k_c and thus is not included as the optimization variable. As Table 4.1 shows, given the whole map of a target area with longitude of a, latitude of b, and assignment time length T, I_c is a binary indicator of whether the vehicle c is incentivized, and k_c specifies the trajectory assigned to vehicle c, $k_c = 0$ is the probabilistic trajectory when c cruises without incentivizing or passengers, and $k_c > 0$ represents a deterministic incentivized trajectory.

In Section 4.3.2, we established that optimizing the sensing data distribution is equivalent to minimizing the *KL-divergence*. The target distribution is provided as O. The resulting collected sensing data distribution P should have minimal divergence from O, which is the objective function as Equation 4.2.

The constraints of this problem include budget constraints (Equation 4.4) and physical mobility constraints (Equations 4.5, 4.6). The budget constraint ensures that the total incentive assigned to all vehicle agents should not exceed the specified budget limit B. Section 4.3.3 shows how to calculate the incentive $B(c)$ to each vehicle c in Equation 4.1. To calculate the incentive, the predicted task request distribution Re and predicted vehicle agent's mobility Pr_c depend on the specific application scenario. For example, in a taxi-based sensing platform in which vehicle agents wish to obtain more ride requests, we utilize the model proposed by [88] to forecast the distribution of ride requests. The mobility prediction model of the taxis can also be learned from historical trajectories of vehicles, as in [27]. We discuss the impact of errors in the task request and mobility prediction models in Section 4.5.

The physical constraints are generated from vehicle agents' mobility. Due to limits on vehicle velocity, each vehicle can either move to a neighboring grid or stay in the original grid within one

time unit; it cannot move further. If the vehicle agent c is selected to be incentivized, that is, $I_c = 1$, the incentivized trajectory D_c is specified by the solution, which requires that each element in the matrix of D_c be either 0 or 1. $D_c(i, j, t) \cdot I_c = 1$ means that in the solution, the vehicle agent c is incentivized to pass through the location (i, j) at the time point t. $D_c(i, j, t) \cdot I_c = 0$ means that the incentivized vehicle agent should not pass through the grid (i, j) at t if $I_c = 1$, or the vehicle agent c is not incentivized if $I_c = 0$. Meanwhile, whether vehicle agents are incentivized or not, there is always $\sum_{i,j}^{a,b} D_c(i, j, t) = 1$ for all t. Since each vehicle agent c can only collect one sensing point at any given time point t, the summation of probability that a vehicle agent c appears in different locations at t should be one.

In the next section, an optimization algorithm is introduced to solve the above problem.

4.4 Algorithm

In this section, we propose a new optimization algorithm, *iLOCuS*, to efficiently solve the problem stated in Section 4.3.4. Given budget constraints, and vehicle mobility constraints, the algorithm selects a set of vehicle agents and incentivizing trajectories to minimize the objective function. However, the optimization problem is NP-complete, and thus cannot be solved in polynomial time. In this section, we first characterize the hardness of the formulated problem in Section 4.4.1. Then we propose an optimization algorithm to solve the formulated problem in Section 4.4.2. In Section 4.4.3, we discuss the mathematical insights and complexity of the algorithm.

4.4.1 Problem Characterization

Lemma 2. *This problem is a non-linear multiple-choice knapsack problem, with a convex non-separable objective function and non-continuous variables.*

We first characterize the problem in Lemma 2. The proof is shown in Appendix B. The problem fits the basic form of a non-linear multiple-choice knapsack problem [21, 98]. The optimization version of classic knapsack problem and quadratic knapsack problem are well-known to be NP-hard. In our nonlinear multiple-choice knapsack problem, the objective function becomes the *KL-divergence* between integrated data distribution and target distribution, which makes the problem even harder than the classic linear knapsack problem.

To show that the optimization version of our problem is NP-hard, we first show the decision version of the problem is NP-complete. The decision version of our problem is: *Does there exist a vehicle incentivizing solution such that the* KL-divergence *between collected sensing data distribution and target distribution is smaller than a specific value* h *while the constraints are satisfied?*

Theorem 4. *The decision version of our problem is NP-complete.*

We showed the proof of Theorem 4 in Appendix C. In Appendix C, we firstly show the decision version of our problem can be verified in polynomial time, hence the problem is NP. Then to prove the problem is NP-hard, we show that one special case of our decision-version problem is equivalent to the decision version of the classic linear multiple-choice knapsack problem, which is widely-known and already proved to be NP-hard [83, 84, 98]. Since the special case is already NP-hard, the decision version of our problem is NP-hard. Therefore, the decision version of our problem is NP-complete. From Theorem 4, since the decision version of our problem has been proven to be NP-complete, it is reasonable to claim that the optimization version of our problem is NP-hard, which is Corollary 1.

Corollary 1. *The optimization version of our problem is NP-hard.*

Based on Theorem 4 and Corollary 1, we show that the formulated problem is NP-hard, which means it is impossible to find an exact optimal solution in a reasonably short time as the scale of the problem increases. The brute-force algorithm has a complexity of $\mathcal{O}(9^T)$, which is not applicable in real-world scenarios. Greedy algorithms can be employed to obtain a sub-optimal, approximated solution. However, the objective function is non-separable with non-continuous variables, which is said to be "much more difficult to solve than the separable problem" [21]. Therefore, it is important to deal with the non-separable function with non-continuous variables where most of the existing greedy algorithms [19, 20, 21, 64] do not apply.

4.4.2 Proposed Algorithm: *iLOCuS*

To solve the formulated non-linear knapsack problem, we proposed *iLOCuS*. The basic idea of *iLOCuS* is that, instead of directly estimate the gradient of *KL-divergence* with respect to each vehicle agent and its trajectories, we can decompose the non-separable objective function in two stages, firstly by spatio-temporal grid level and then by vehicle agent level. In detail, *iLOCuS* does the following steps in an iterative way: 1) find the time-location pair with the highest ratio between the number of vehicle agents in current solution and the desired vehicle agents at the respective time and location, and 2) dispatch part of passing vehicle agents in the found time-location pair to different trajectories to decrease the *KL-divergence*.

Algorithm 1 describes the steps to keep improving the quality of sensing distribution. We firstly initialize a feasible solution S under the constraints, which is implemented by randomly selecting the vehicle agents until the budget is spent. The solution S includes the binary indicator I_c and feasible assignment $D_c^{k^*}$ for incentivized vehicle agent c. Then we iteratively update S such that the objective function is minimized. During each iteration, we select the maximum value of $P(i,j,t)/O(i,j,t)$, which increases monotonously with the gradient of *KL-divergence* at $P(i,j,t)$.

Algorithm 1: *iLOCuS*

Input : Location and occupancy information for all vehicle agents
Output: An improved feasible solution $\mathcal{S}^{\star}$

Initialize a feasible solution $S = \{I_c, D_c^{k^*}\}$ and respective incentive $\{B(c)\}$ for all $c \in \{1, \cdots, C\}$;
set $\mathcal{S}^{\star} = \mathcal{S}$;
for $i++ \leq$ *MaxIter* **do**
    $S = S^{*}$;
    Select $(i^*, j^*, t^*) = arg\max_{i,j,t} P(i,j,t)/O(i,j,t)$;
    Select vehicle agents where $D_c^{k^*}(i^*, j^*, t^*) > 0$ and get the set tmp_car;
    $c = tmp\_car \rightarrow head$;
    while $c! =$ *null* **do**
        Get potential trace set $\{D_c^k\}$ of vehicle agent c, where $k \in \{1, \cdots, K\}$;
        if *Total cost is less than* B **then**
            Select $k = arg\max_k V(c, D_c^{k^*}, D_c^k)$
        end
        $c = c \rightarrow next$
    end
    Select $(c', k') = arg\max_{c,k} V(c, D_c^{k^*}, D_c^k)$;
    if $k' > 0$ **then**
        Update $\mathcal{S}^{*}$ as $I_c' = 1$ and $D_{c'}^{k^*} = D_{c'}^{k'}$;
        Update respective incentive $B(c')$
    else
        $I_{c'} = 0, B(c') = 0$;
    end
end
Output $\mathcal{S}^{\star}$ including I_c, $D_c = D_c^{k^{\star}}$, and respective incentive $B(c)$.

We can obtain the respective time and location pair

$$(i^*, j^*, t^*) = arg \max_{(i,j,t)} P(i,j,t)/O(i,j,t).$$

We need to adapt the number of vehicle agents in (i^*, j^*, t^*) to best decrease the ***KL-divergence***. We also defined $V(c, D_c^{k^*}, D_c^{k'})$ to measure how much the ***KL-divergence*** will decrease when switching the vehicle agent c from current k^*th trajectory to the new k'th trajectory, where

$$\begin{aligned} &V(c, D_c^{k^*}, D_c^{k'}) \\ &= \sum_{i,j,t} (\log \frac{P(i,j,t)}{O(i,j,t)} + 1)(D_c^{k^*}(i,j,t) - D_c^{k'}(i,j,t)) \end{aligned} \tag{4.7}$$

The basic idea of designing the $V(c, D_c^{k^*}, D_c^{k'})$ is to use the first-order gradient to approximate the difference induced by switching the trajectory. Take (c, k') as $arg \max_{c,k'} V(c, D^{k^*}, D^{k'})$ from all positive V belonging to vehicle agents in (i^*, j^*, t^*). We can decrease the ***KL-divergence*** approximately most by updating the trajectory of the vehicle agent c to $D^{k'}$ in the current solution S. To distinguish different candidate trajectory for each vehicle agent, we use D_c^k to express the kth possible trajectory of the vehicle agent c. When an unoccupied vehicle agent is not incentivized, she/he follows the usual trajectory, which is a probabilistic trajectory learned from Pr_c, and denoted as D_c^0.

The time complexity of the algorithm is upper bounded by $\mathcal{O}(CT^4)$, where C is the total number of vehicle agents and T is the length of each incentivizing period. For each vehicle agent, it has $\mathcal{O}(T^2)$ potential destinations. Therefore each vehicle agent can only travel in the graph constructed by the $\mathcal{O}(T^2)$ vertices and the $\mathcal{O}(T^2)$ edges connecting those vertices. Using the ***Bellman-Ford*** algorithm to find the trajectory with maximum value from all the candidate trajectories will cost $\mathcal{O}(T^2 \cdot T^2) = \mathcal{O}(T^4)$. Therefore in the worst case, the overall time complexity for our algorithm is upper bounded by $\mathcal{O}(CT^4)$.

4.4.3 Insights Behind *iLOCuS*

Our objective function is non-separable with respect to the variables I_c and D_c, since it cannot be converted to the form $\sum_q f_q(I_c, D_c)$, i.e., a linear combination of a group of functions f_q. Meanwhile, both I_c and D_c are not continuous variables. Therefore, it is difficult to use the conventional gradient descent method with respect to the optimization variables I_c and D_c. However, the objective function is convex with respect to $P(i,j,t)$. To minimize the objective function, we can do gradient descent on each $P(i,j,t)$ until the gradient is near 0, which is similar to the idea of coordinate descent with respect to the variables $\{P(i,j,t) : \forall i,j,t\}$. Given a convex function

$f(x_1, \cdots, x_n)$ for which it is difficult to simultaneously perform gradient descent with respect to $X = (x_1, \cdots, x_n)$, coordinate descent performs gradient descent at $x_1, \cdots, x_n$ separately to achieve the optimal solution with a much faster convergence rate [26, 161, 220]. The gradient of the objective function with respect to $P(i, j, t)$ is then

$$\frac{\partial KL(P||O)}{\partial P(i,j,t)} = \log \frac{P(i,j,t)}{O(i,j,t)} + 1$$

To accelerate the gradient descent at $P(i, j, t)$, we need to select the steepest direction, which is $(i^*, j^*, t^*) = arg\max_{i,j,t} \log P(i, j, t)/O(i, j, t)$. Then at each iteration, we keep doing gradient descent on each $P(i^*, j^*, t^*)$ until the objective function converges. To decrease the gradient, we need to remove some of the vehicle agents in (i^*, j^*, t^*). Whichever vehicle agent is removed, the gradient with respect to current (i^*, j^*, t^*) always decreases by the same amount as $\log 1/(CT \cdot O(i^*, j^*, t^*)) + 1$.

However, the change in the objective function induced by switching the trajectory varies with different vehicle agents. When removing one vehicle agent at (i^*, j^*, t^*), there must be some other (i', j', t') in which the number of vehicle agents increases. Different $P(i, j, t)$ are no longer independent with each other when changing I_c and D_c^k, since $P(i, j, t)$ is respect to the overall vehicle agents' distribution, while I_c and D_c^k are respect to each vehicle agent's mobility. Therefore, directly using the conventional coordinate descent method cannot solve our problem. It is important to consider how to change I_c, D_c^k to realize the largest decrease of the objective function while performing gradient descent with respect to $P(i, j, t)$.

It is computationally expensive to directly calculate how much the objective function decreases with switching a trajectory. We approximate this change using the product of the gradient and changes of $P(i, j, t)$. When switching the trajectory of one vehicle agent c, only the numbers of vehicle agents at times and locations related to c's old and new trajectories will change. Therefore, only a few $P(i, j, t)$ changes, and the computational efficiency is largely improved. We denote the decreasing of the objective function due to switching vehicle agent c's trajectory from D^{k^*} to $D^{k'}$ as $V(c, D^{k^*}, D^{k'})$. If $V > 0$, the objective decreases. The larger V is, the more the objective function decreases. We have

$$\begin{aligned} &V(c, D^{k^*}, D^{k'}) \\ &\propto \sum_{i,j,t} (\log \frac{P(i,j,t)}{O(i,j,t)} + 1)(D_c^{k^*}(i,j,t) - D_c^{k'}(i,j,t)), \end{aligned}$$

It is easy to compute $V(c, D^{k^*}, D^{k'})$, since the deterministic trajectory D_c^k is a sparse matrix. Also, when computing the gradient at the beginning of each iteration, we have stored the matrix of

$\log P(i,j,t)/O(i,j,t)+1$. Denote $C^\star$ as the set of vehicle agent c where $D_c^{k^*}(i^\star,j^\star,t^\star)>0$. For $c \in C^\star$, we firstly check whether a potential new trajectory $D_c^{k'}$ satisfies the budget constraints and store the respective incentive for late usage. The number of budget-feasible trajectories changes with the amount of leftover budget in the current solution. It is generally much smaller than the total number of candidate trajectories. For a budget-feasible trajectory D_c^k, we can use the dynamic programming method like Bellman-Ford algorithm to obtain the best trajectory $D^{k'}$ for c to incentivize to. If $k'=0$, it means the best solution is to let the vehicle agent c' run as usual without incentivizing. Otherwise, we will incentivize the vehicle agent c to the trajectory $D_{c'}^{k'}$ instead of its original trajectory. The algorithm repeats the above steps until the *KL-divergence* converges or the maximum number of iterations is achieved. We will output a final solution $\mathcal{S}^\star$, which contains 1) I_c: whether one vehicle agent is incentivized, 2) D_c, the trajectory the vehicle agent c is assigned, and 3) respective incentives $B(c)$ for all incentivized vehicle agents.

4.5 Evaluation

To evaluate our algorithm, we used a real mobile crowd sensing system based on taxis to collect real-world historical data of taxis' mobility and conduct experiments to show the performance. In Section 4.5.2 and 4.5.3, we evaluate the impact of the number of vehicles, budget, target distribution, incentive mechanism, and mobility prediction model accuracy on the *KL-divergence*. We compare the performance of *iLOCuS* with no incentivizing, random incentivizing, random incentivizing with the proposed incentives, and Greedy method to optimize the spatial coverage.

4.5.1 Experiment Setup

Experimental Dataset

This work uses real-world taxis' trajectories to evaluate our algorithm design. The dataset includes trips of $20,067$ taxis in one month in the city of Beijing, one of the biggest cities in China. Each record in the dataset contains taxi id, time, location and occupancy status. The location is expressed as longitude and latitude while the occupancy status represents whether the taxi is occupied by one or more customers. Each taxi collects one record every minute whenever it is operating. We also have the respective ride request information. The evaluation area occupies a size of $15km$ by $15km$ and is discretized into a 15×15 map grid, in which each grid has the size of 1×1.

We predict the mobility of each vehicle agent using the method developed by [27] where the spatio-temporal dynamics of taxi mobility are probabilistically encoded using a Markov model to improve the prediction accuracy. As for the task request prediction, we utilize the time-space

graphical model proposed by [88] and combine historical ride requests of Beijing taxis to forecast the ride request distribution in Beijing city. By learning the temporal evolution and spatial property of the constructed ride request topological graph, this model has been shown to be accurate on the real-world dataset in [88].

We use the Beijing taxi trajectory dataset during November 2015. We discretize the map with 1km grids. We set the incentivizing period as $T = 10$min, which depends on the average distance of a riding, 5km, and the average velocity of vehicle agents, 30km/h. We set the time resolution as 2min, which is the average time needed to run 1 grid. Due to the 2 USD flag-down fare of Beijing taxi, we adopt $r_u = 2$USD/min, $r_{min} = 2$USD and $r_{max} = 20$USD. The first 3 weeks' data is used for training mobility prediction and ride request prediction models, while the rest of the month is used for testing our method. To simplify the problem, we consider the sampling frequency of each vehicle agent as 1 data point per 2 minutes, which means in each grid, a taxi can collect 1 data point. This can be expanded to a higher sampling frequency, but the sensing distribution will not change. We assume that all vehicle agents stay inside the map where the crowdsourcer aims to sense. In order to consider the temporal variations of taxi density, taxi mobility pattern, and ride request density, we take data from multiple times of a day 0:00 am, 6:00 am, 9:00 am 12:00 pm and 6:00 pm.

Benchmark Methods

We adopt three benchmark methods to compare our algorithm with and to validate our algorithm's ability to improve the sensing distribution quality.

- *No Incentivizing (NA)*: This method does not incentivize taxis nor match ride requests. All the taxis just follow their original trajectories. We can check the performance improvement of our method by comparing with NA.
- *Random Incentivizing (RND)*: This method randomly selects taxis and incentivizes them to the sparsest areas with random trajectories within the given budget. *RND* always offers the maximum monetary incentive $r_{\max}$. By comparing this method with our method, we can check the performance improvement brought by our customized incentive and optimization scheme.
- *Random Incentivizing with Ride Request Prediction (RND_RQ)*: This method randomly incentivizes taxis and trajectories within the budget while offering the incentive $B(c)$ as described in Equation 4.1, which includes the ride request in the destination as a hidden incentive. By taking the same incentive mechanism, we compare this method with our method to show the performance improvement of optimizing the incentivizing decision I_c and assignment trajectory D_c.

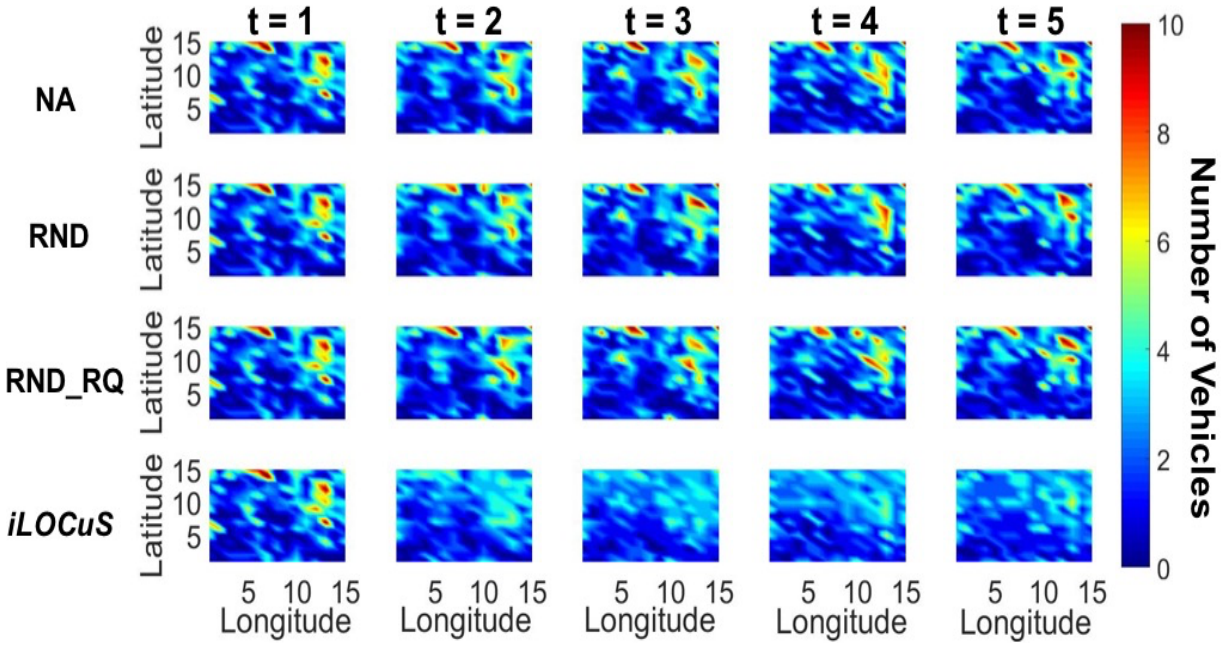


Figure 4.5: This figure visualizes the incentivizing results with **Ob_dist_1** as target distribution. The first row is the collected data distribution under no incentivizing, the second row is the distribution under random incentivizing, the third row is the collected data distribution using random incentivizing with proposed incentive, and the fourth row is the incentivizing results under our algorithm *iLOCuS*. The brighter the area is, the denser the vehicles/sensing data points are.

- *Greedy-SC*: This method is from [74] discussed in Section 4.2. *Greedy-SC* is developed to greedily choose vehicle agents that can maximize the *Cost Effectiveness* value to maximize the number of covered region in all periods of time using predictable mobility of vehicle agents.

Performance Metrics

We take the objective function, *KL-divergence*, as one of the quantitative evaluation metric. As discussed in Section 4.3.2, small *KL-divergence* means high quality of sensing distribution. Therefore, the algorithm minimizing *KL-divergence* performs the best.

To show how much the *KL-divergence* is decreased by algorithm $algo^\star$ compared to benchmark method $algo$, we define the *divergence reducing percentage (DRP)* as follows:

$$DRP(algo^\star, algo) = \frac{KL_{algo} - KL_{algo^\star}}{KL_{algo^\star}}.$$

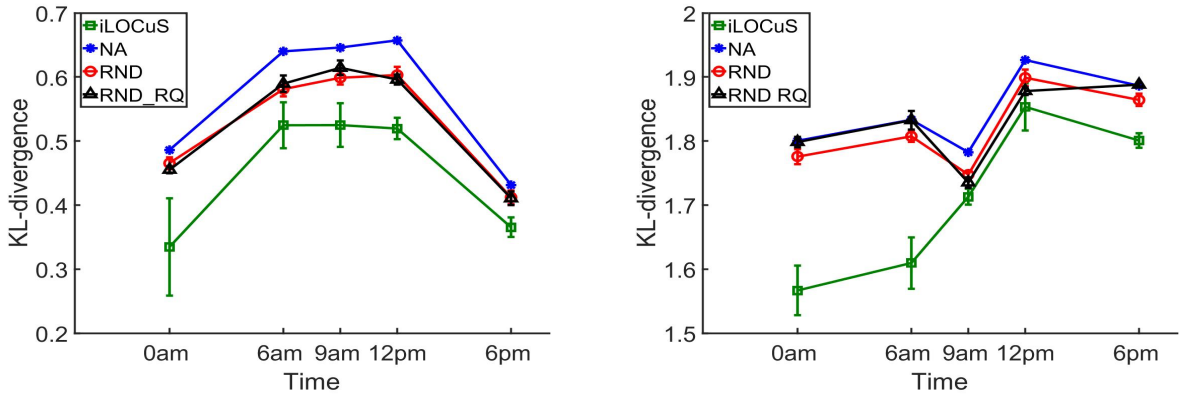
DRP measures how much $algo^\star$ decreases the *KL-divergence* compared to $algo$, which represents the improvement of the quality of sensing distribution by using the incentivizing policy from $algo^\star$. The higher the DRP is, the better the algorithm performs.

4.5.2 Performance under Multiple Target Distributions

To show the performance under different target distributions, we investigated 4 target distributions under the same budget (1000 USD), vehicle number (500):

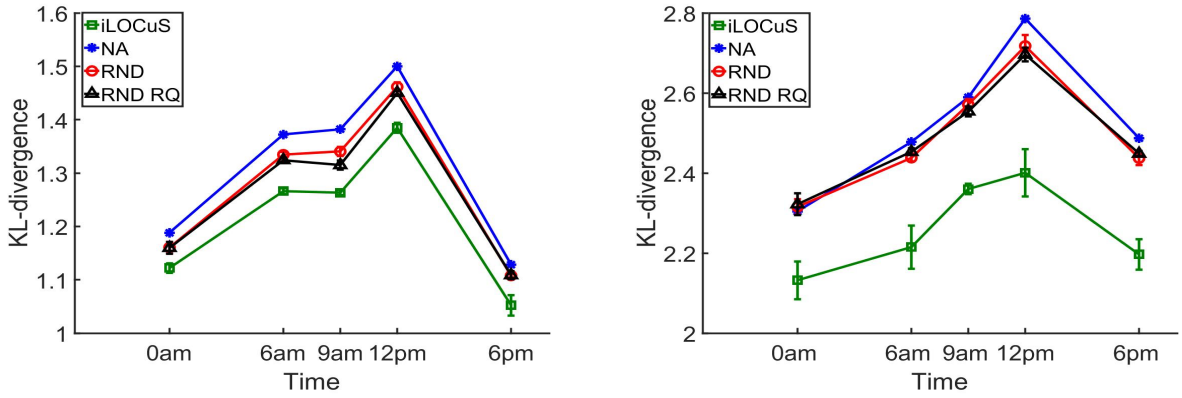
- **Ob_dist_1** Uniform distribution over the temporal and spatial domain. For example, when data request end needs the air pollution data over all regions [45, 91].
- **Ob_dist_2** Gaussian distribution over the spatial domain, where the probability mass function achieves the maximum at the grid $(10, 10)$ with fixed variance at each time point. For example, when data request end would like to focus on monitoring the factory neighborhood all day [7, 180, 239].
- **Ob_dist_3** Gaussian mixture distribution over the spatial domain, where the probability mass function is centered at the grid $(5, 10)$ and $(10, 5)$ with fixed variance at each time point. For example, when data request end plans to focus on monitoring both the factory neighborhood and center areas all day [180, 239].
- **Ob_dist_4** Gaussian distribution in spatial domain but the probability mass function achieves the maximum at different grids at different time. For example, when data request end would

like to monitor the factory neighborhood during daytime and center area during night [180, 239].



(a) **Ob_dist_1**: Uniform distribution over the temporal and spatial domains

(b) **Ob_dist_2**: Gaussian distribution over the spatial domain



(c) **Ob_dist_3**: Gaussian mixture distribution over the spatial domain

(d) **Ob_dist_4**: Gaussian distribution over space and peak changes with time

Figure 4.6: This figure shows the *KL-divergence* under 4 different target distributions at the different time of a day. The figures compare the performance of *iLOCuS* (Square green line) with multiple benchmark methods, including no incentivizing NA (Star blue line), random incentivizing RND (Circle red line) and random incentivizing with the proposed incentive RND_RQ (Triangle black line).

Table 4.2: Divergence reduction percentage of the *KL-divergence* by different algorithms compared to no incentivizing, $DRP(algo^*$, NA), under different target distributions and time of the day. The $algo^*$ includes *iLOCuS*, and benchmark methods (RND and RND_RQ).

Time	**Ob_dist_1** Uniform			**Ob_dist_2** Gaussian over Space			**Ob_dist_3** Gaussian Mixture			**Ob_dist_4** Gaussian over Space and Time		
	iLOCuS	RND	RND_RQ	*iLOCuS*	RND	RND_RQ	*iLOCuS*	RND	RND_RQ	*iLOCuS*	RND	RND_RQ
0:00 am	**45.26%**	4.39%	6.79%	14.92%	1.40%	0.11%	4.66%	0.61%	2.64%	13.12%	0.47%	-0.89%
6:00 am	**22.00%**	10.16%	8.57%	13.89%	1.43%	0.04%	7.26%	1.70 %	3.52%	14.98%	1.89%	0.69%
9:00 am	**23.08%**	7.89%	5.16%	4.06%	2.00%	2.73%	10.48%	4.55%	5.87%	9.12%	-0.60%	0.79%
12:00 pm	**26.52%**	8.99%	10.27%	3.95%	1.46%	2.59%	10.42%	3.42%	3.93%	18.90%	2.27%	4.01%
6:00 pm	**18.09%**	4.72%	5.04%	4.75%	1.19%	-0.09%	5.87%	1.69%	2.07%	18.57%	4.17%	1.81%
Average	**26.99%**	7.23%	7.17%	**8.31%**	1.50%	1.07%	**7.74%**	2.40%	3.61%	**14.94%**	1.64%	1.28%

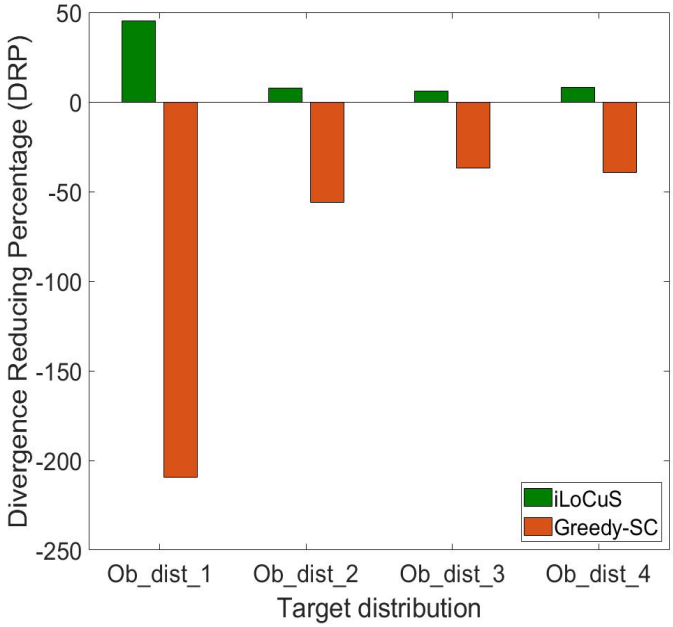
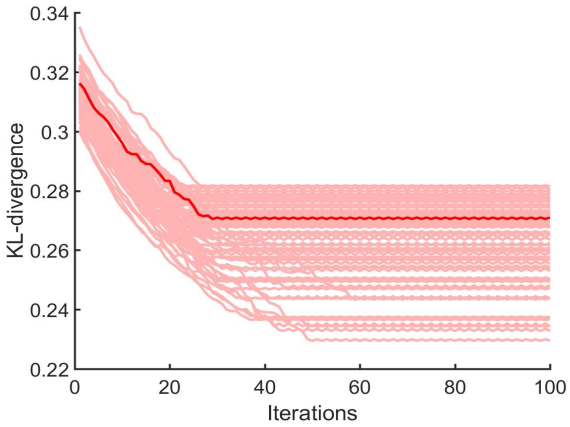
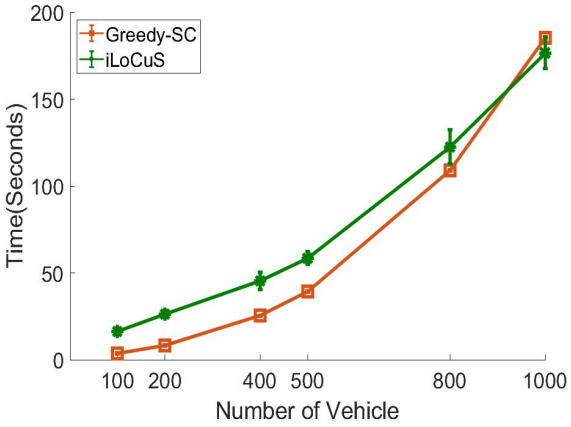


Figure 4.7: This figure shows the DRP of *iLOCuS* and baseline method Greedy-SC. It can be found that since Greedy-SC is designed for a different objective function, using Greedy-SC to incentivize vehicle agents will increase the dissimilarity between the collected data distribution and target distribution compared to not incentivizing any vehicle agents.

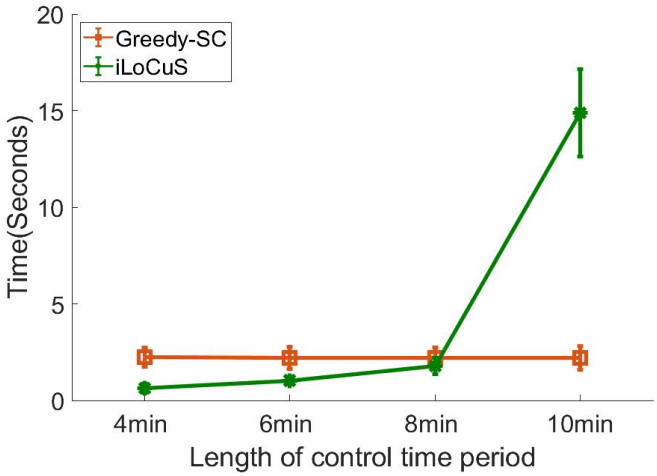
Uniform distribution is one of the most common target distribution for sensing systems [91, 248, 249], which ensures the information is collected uniformly over the temporal and spatial domain. Gaussian distribution and Gaussian mixture distribution over spatial domain consider two common distributions. In these two distributions, the importance level of sensing data changes with their spatial location, while keeping consistent over time. Using these two distributions aims to test the performance of *iLOCuS* in the type of scenario where the data request end may focus more on some specific area(s) and expect to obtain more information near the target spatial area(s). A Gaussian distribution over temporal and spatial domains targets a more complicated distribution, where the importance level of sensing data changes with both time and their spatial location. We investigate the 4 different distributions to show the generalization of our algorithm. Figure 4.5 visualizes an example of the incentivizing results compared with a target uniform distribution. The sensing distribution, which is also the taxi distribution, at the beginning of one incentivizing period is shown in the column of $t = 1$. While from the next time point $t = 2$ to the end of the incentivizing period, the sensing distribution of *iLOCuS* is much more uniform than the benchmark methods. Figure 4.5 shows that the dark red and the dark blue areas in the results of no incentivizing (NA), which indicate too much or too few taxis, disappear in the results of *iLOCuS*.



(a) *KL-divergence* vs Iteration



(b) Time vs Number of Vehicle



(c) Time vs Length of incentivizing period

Figure 4.8: This figure shows the convergence and time complexity with uniform target distribution. (a) shows the *iLOCuS*'s iteration number of convergence with different initialization. (b) shows the computation time of both *Greedy-SC* and *iLOCuS* increases with the number of vehicles and compares the time between *Greedy-SC* and *iLOCuS*. (c) shows the computation time *iLOCuS* increases much faster with the length of the incentivizing period T than *Greedy-SC*. The fast increasing of *iLOCuS*'s time computation is mainly induced by selecting trajectories from exponentially increasing candidate trajectories, while *Greedy-SC* does not allow trajectory selection.

Observations

Figure 4.6 shows that *iLOCuS* always outperforms the benchmark methods, which proves the robustness of *iLOCuS* under different target distributions. Figure 4.6(a), 4.6(b), 4.6(c) and 4.6(d) compare the performance of *iLOCuS* with benchmark methods under the **Ob_dist_1**, **Ob_dist_2**, **Ob_dist_3**, and **Ob_dist_4** at different time of a day. The results show that the *KL-divergence* of *iLOCuS* is always smaller than the other benchmark methods under the 4 different target distributions. Table 4.2 presents the divergence reduction percentage of each method compared to no incentivizing. It is shown that under **Ob_dist_1**, *iLOCuS* reduced the *KL-divergence* of 26.99% on average compared to no incentivizing, which improves 19.76% and 19.82% compared to random incentivizing and random incentivizing with proposed incentive, respectively.

Combining Figure 4.6 and Table 4.2, we make several observations that we elaborate below: 1) The *KL-divergence* under no incentivizing (blue line) in Figure 4.6, mostly achieves the maximum at 12:00 pm and the minimum at 0:00 am or 6:00 pm except **Ob_dist_2**. 2) The *KL-divergence* obtained by our algorithm *iLOCuS* has the similar trend with the *KL-divergence* under no incentivizing. 3) Our algorithm performs best on both *KL-divergence* and DRP under **Ob_dist_1**, which is uniform target distribution. 4) At most time, random incentivizing (random incentivizing and random incentivizing with proposed incentive) performs better than no incentivizing.

For observation (1), at around 12:00 pm, people mostly work and travel in the central area. Thus most taxis also gather in the central area to locate passengers and/or ride requests. This results in a highly concentrated distribution, which is quite different from the target distributions. At 0:00 am, there are few human activities in the central area, and the traffic is less congested. The drivers are able to cruise in a larger area for finding passengers. At 6:00 pm, people head back to the residential areas, which are mostly located on the border of the map. Therefore, at 0:00 am and 6:00 pm, the taxis mostly head to the area which has few taxis at daytime (12:00 pm), which is similar with common commuting pattern in large cities [245]. This makes the sensing distribution less different from our target distributions compared to 12:00 pm. For **Ob_dist_2**, the *KL-divergence* under NA at 9:00 am is similar with 0:00 am. It may be because the taxis gather in the center area closer to the grid of $(10, 10)$ at 9:00 am. The difference between **Ob_dist_2** and **Ob_dist_4** may be because **Ob_dist_4** has changing variance in time domain, while the **Ob_dist_2** requires uniform distribution over the time domain.

For observation (2), our incentivizing algorithm is based on the mobility of all the taxis. There are around $100 - 250$ occupied taxis from the 500 total taxis. These occupied taxis are not able to be incentivized, which means that the quality of sensing distribution will be significantly influenced if these taxis gather in the center area. Meanwhile, if the traffic conditions and weather conditions influence the sensing distribution under no incentivizing, our incentivizing results will

be changed in a similar way. Therefore, the performance of *iLOCuS* is closely correlated to the no incentivizing sensing distribution.

For observation (3), compared to the uniform distribution, the three Gaussian distributions, **Ob_dist_2**, **Ob_dist_3** and **Ob_dist_4**, are more concentrated. From Figure 4.6, we found the *KL-divergence* under NA in uniform distribution ranges in $[0.4, 0.7]$, which is much smaller than the *KL-divergence* under NA in those Gaussian distributions which ranges in $[1.1, 2.8]$. This shows that these Gaussian distributions diverge much more from the real vehicle distribution than uniform distribution. Incentivizing vehicles to the three Gaussian distributions are more difficult than to uniform distribution under the same conditions.

As for observation (4), in general, random incentivizing can make the distribution more uniform than no incentivizing since random incentivizing dispatches the taxis to the destinations where there are fewer number of taxis. But the random incentivizing methods perform much worse than *iLOCuS*, as shown in Table 4.2. This may be because those random incentivizing methods do not consider the influence of trajectory on our objective function.

We compared *iLOCuS* with *Greedy-SC* as well. Figure 4.7 compares the DRP between *Greedy-SC* and *iLOCuS* under 4 different target distributions in the same condition. It can be found that directly using *Greedy-SC* will only increase the dissimilarity between the collected data distribution and target distribution. This may be because 1) the objective of *Greedy-SC* only cares about the coverage in the spatial domain, therefore not jointly optimizing the sensing distribution on both temporal and spatial domains; 2) the objective of *Greedy-SC* only cares about the number of covered grids instead of the exact density of collected data points at each grid. So the collected data distribution by *Greedy-SC* is far from the target distribution considering both temporal and spatial domains. The absolute DRP of *Greedy-SC* achieves the maximum for uniform distribution, which may be because the *KL-divergence* is sensitive to the changes of density in all grids in uniform distribution but only center grids in Gaussian distributions.

Time Complexity

We also show the convergence of *iLOCuS* with **Ob_dist_1** as the target distribution at different times of the day. Figure 4.8(a) shows that *iLOCuS* converges after around $20 - 60$ iterations, which shows the convergence. The variance of the converged value of *KL-divergence* is smaller than the variance of *KL-divergence* based on the finally collected data in Figure 4.6(a). This is because the crowdsourcer optimizes the policy based on the probabilistic mobility prediction of each unoccupied vehicle agents but the vehicle agent will finally only choose one deterministic trajectory during running in real scenarios. By improving the accuracy of mobility prediction, this variation can be mitigated. Figures 4.8(b) and 4.8(c) compare the time complexity under different

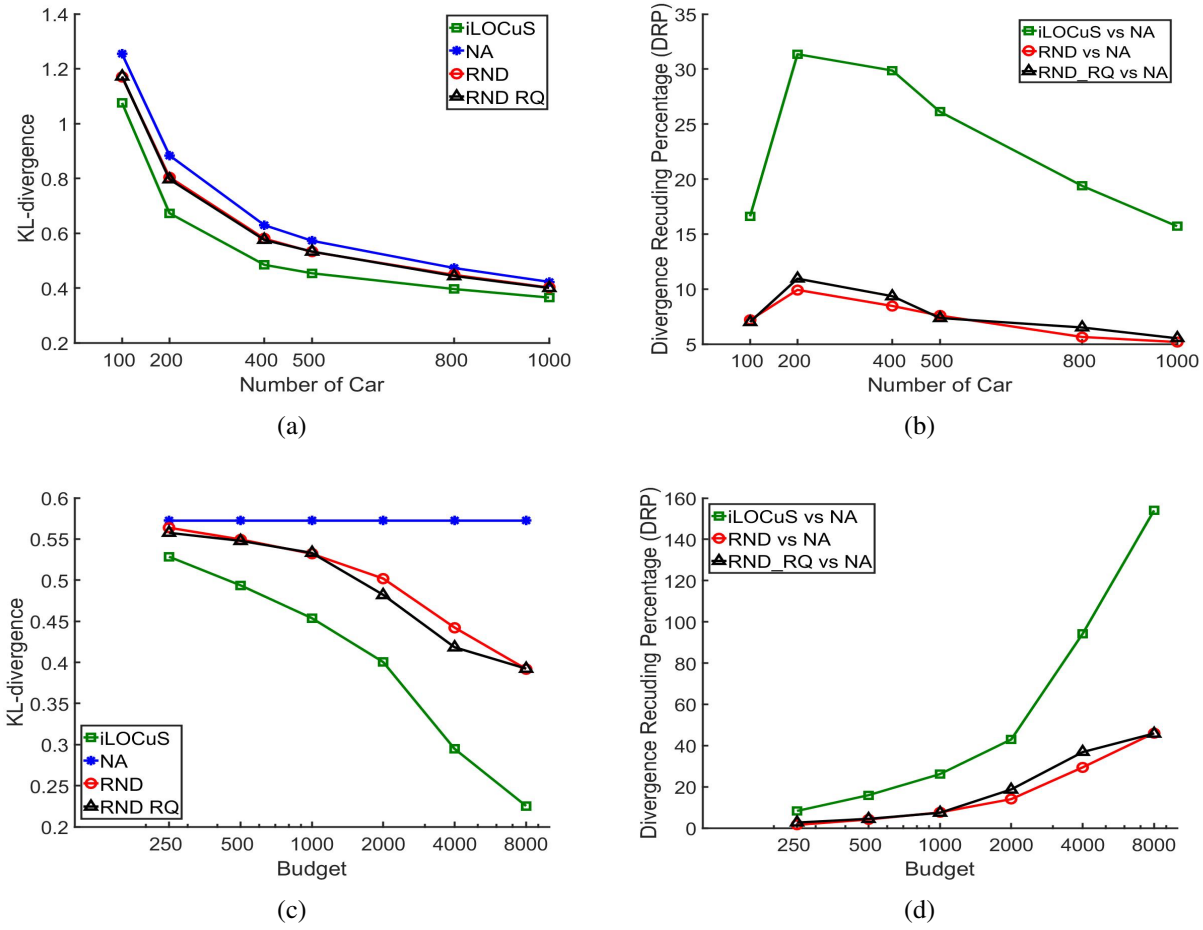


Figure 4.9: This figure shows the performance of *iLOCuS* at **Ob_dist_1** target distribution under the different number of vehicle agents and budget. (a) presents how *KL-divergence* changes with increasing number of vehicle agents; (b) shows the DRP comparing the average *KL-divergence* of *iLOCuS*, RND, and RND_RQ over time with the average of NA over time under the different number of vehicle agents. (c) presents how the average *KL-divergence* over different time of the day changes with the amount of budget; (d) shows the DRP comparing the average *KL-divergence* of *iLOCuS*, RND, and RND_RQ over time with the average of NA over time under different budgets.

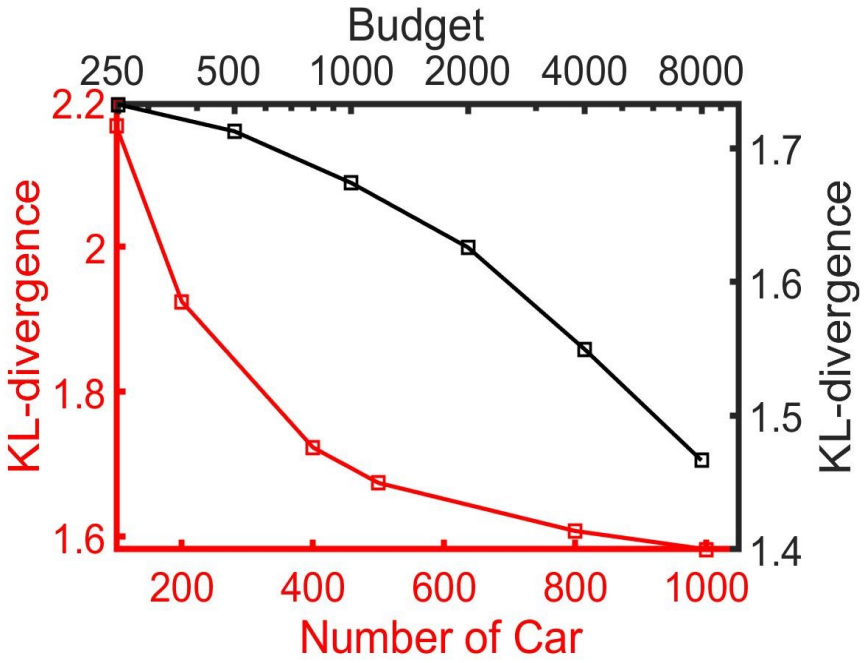


Figure 4.10: This figure shows that under **Ob_dist_2**, how the *KL-divergence* of *iLOCuS* changes with number of cars and budget, respectively. The red line and axes represent the changing of *KL-divergence* with number of cars. The black line and axes show the changing of *KL-divergence* with budget.

numbers of vehicles and lengths of incentivizing period. The time of *Greedy-SC* increases faster with the number of vehicles than *iLOCuS*, but slower with length of incentivizing period than *iLOCuS*. The complexity of *Greedy-SC* is $\mathcal{O}(C^2T)$ while *iLOCuS* is $\mathcal{O}(CT^4)$. Our *iLOCuS* depends more on the length of incentivizing period is because that *Greedy-SC* only selects vehicle agents but no trajectories while *iLOCuS* selects both vehicle agents and the trajectories for incentivized vehicle agents.

Since *iLOCuS* and *Greedy-SC* have different optimization objective function, here we mainly focusing on comparing their overall performance. In the following exploring influence factors on our *iLOCuS*, we will no longer compare to *Greedy-SC*.

4.5.3 Influence Factors

In this section, we investigate the performance of *iLOCuS* under different influence factors, including the number of vehicles, budget, incentives, the accuracy of mobility prediction, and vehicle's acceptance rate. We use **Ob_dist_1** as the target distribution for this analysis. The results show that *iLOCuS* is robust to different setups and always outperforms the benchmark methods.

Number of vehicles

With uniform target distribution, under different number of vehicles, *iLOCuS* always achieves the lowest *KL-divergence* compared to benchmark methods as Figure 4.9 shows. With increasing number of vehicles, there are more free vehicles available and thus the *KL-divergence* decreases in all methods. Figure 4.10 shows a similar trend that the *KL-divergence* decreases with the number of cars with the target distribution of **Ob_dist_2**. For Figure 4.9(a), 4.9(b) and red line in Figure 4.10, we fix the other influence factors as budget and take the average at different times of the day.

We also compare the averaged *KL-divergence* across different times of the day obtained by different algorithms under uniform distribution. We get $DRP(algo, NA)$. $DRP(algo, NA)$ measures how much the *KL-divergence* is reduced by the algorithms (*iLOCuS*, *RND*, and *RND_RQ*) compared to no incentivizing (*NA*). It is shown that the DRP of *iLOCuS* outperforms other benchmark methods and achieves upto 31.35% improvement. All algorithms reach their maximums when the number of vehicles is 200. When the number of vehicles is less than 200, more vehicles improve the flexibility for the algorithm to incentivize and incentivize vehicles. So the DRP increases with the number of vehicles increasing to 200, even though the reference *KL-divergence* in no incentivizing keeps decreasing as Figure 4.9(a) shows. When vehicle number keeps increasing from 200 with the fixed budget, it is possible that when the number of vehicles becomes larger, more vehicles need to be incentivized to realize the same sensing distribution with that under a small number of vehicles. Since all algorithms need to incentivize the vehicles, the budget constraint limits the ability to incentivize more vehicles to more sparse places to improve the sensing quality, which makes the increasing more difficult.

Budget

Figure 4.9(c) shows that the *KL-divergence* obtained by all methods except no incentivizing decreases with increasing budget with uniform target distribution. Under different available budgets, *iLOCuS* outperforms the benchmark methods. With more budget available, all methods can incentivize more vehicles except no incentivizing. Meanwhile, the *KL-divergence* obtained by *iLOCuS* decreases more quickly than the benchmark methods with budget increasing. This shows that our algorithm utilizes the budget much more efficiently than other methods. Figure 4.10 shows similar trend that the *KL-divergence* decreases with the budget with the target distribution of **Ob_dist_2**. For Figure 4.9(c), 4.9(d) and black line in Figure 4.10, we fix the number of vehicles as 500 and take an average at the different times of the day.

Meanwhile, as Figure 4.9(d) shows, DRP of *iLOCuS* increases as budget increases and achieves around 154% when the budget is 8000. The difference between *iLOCuS* and benchmark methods also increases. Compared to no incentivizing and random incentivizing, *iLOCuS* can reduce the

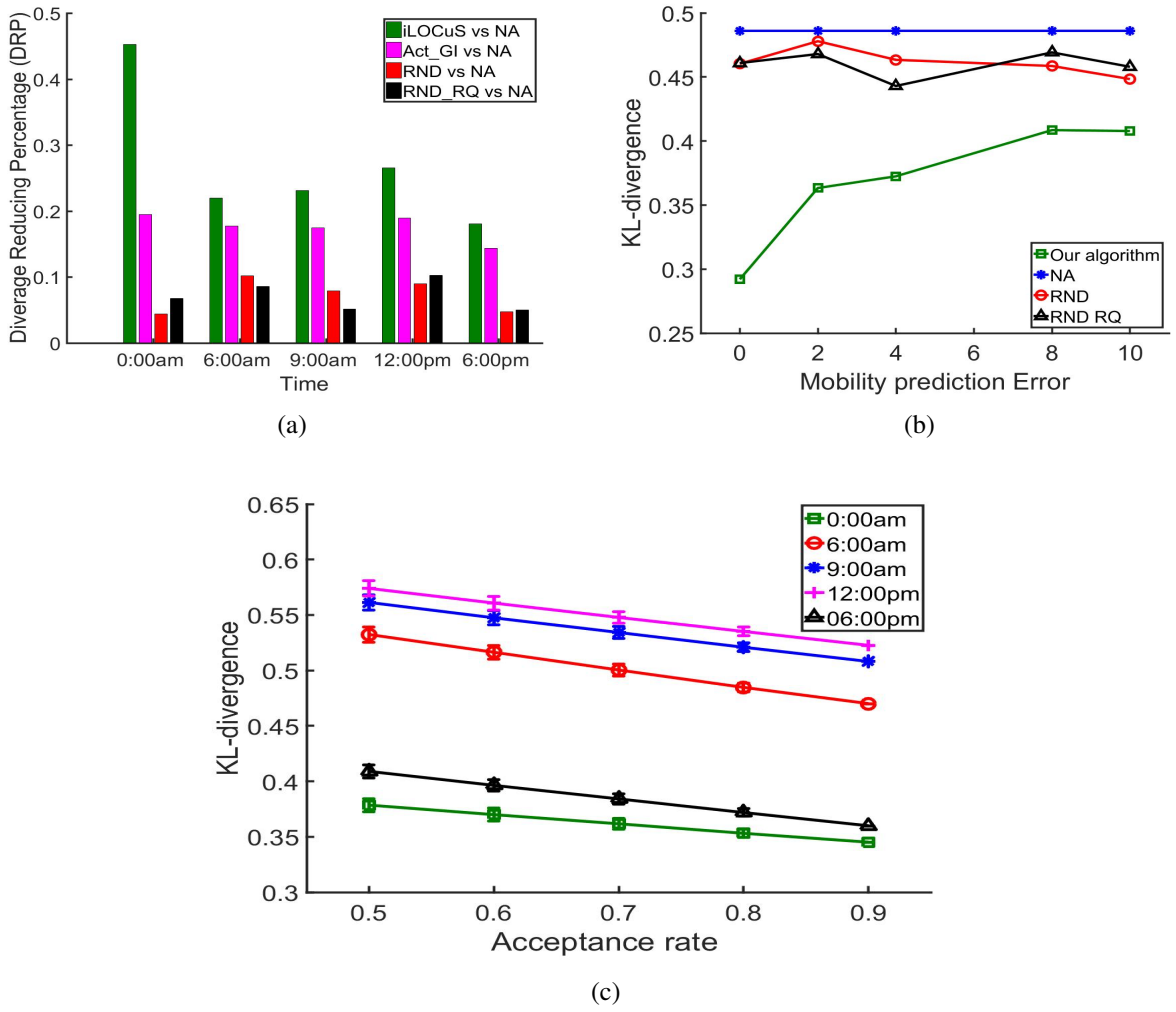


Figure 4.11: This figure shows the influence of the incentive design, mobility prediction accuracy, and acceptance rates on the performance of the algorithms. (a) shows the DRP of the *KL-divergence* by using different incentives and different incentivizing algorithms under **Ob_dist_1** at the different time. (b) shows the average *KL-divergence* changes with different mobility prediction bias at 0:00 am. (c) shows the *KL-divergence* under different acceptance rates at the different time in one of our experiments.

cost of incentivizing vehicle agents using ride request as hidden incentives, and realized the same performance (*KL-divergence*) with less budget. Compared to random incentivizing with proposed incentive, *iLOCuS* selects and incentivizes the vehicles which can help reduce the objective value the most, and thus better utilizes the budget.

Incentives

We then explore the application of our algorithm under different incentives. We test the performance of our optimization method under fixed incentives $r_{\max}$ instead of $B(c)$ defined in Equation 4.1. We denote Act_GI as the policy obtained by using our optimization algorithm with all the incentives of $r_{\max}$. We get the divergence reducing percentage of each method compared to no incentivizing. As Figure 4.11(a) shows, the DRP of all methods rank as *iLOCuS* $> Act\_GI > RND\_RQ \approx RND$, while all the methods outperform than no incentivizing.

The results show 2 points: 1) our optimization algorithm can help decreasing the *KL-divergence* sufficiently with different incentive methods. 2) with our optimization method, the new customized incentive can efficiently improve the performance of our crowd sensing system when comparing *iLOCuS* and Act_GI; We compared the performance of using our optimization method and general constant incentive Act_GI (magenta), with random incentivizing (red) and random incentivizing with proposed incentive (black) in Figure 4.11(a). Even with constant incentives, our optimization algorithm still reduces the divergence significantly.

Mobility prediction accuracy

We showed the robustness of our algorithm to the different accuracy levels of mobility prediction. Here we need to take mobility prediction results for taxis with different accuracy. When taking mobility distribution with varying bias, we first compute the Euclidean distance between a candidate trajectory to the trajectory finally chosen by the taxi [78]. Taking the distance d as the variable, we assign probability $p(d)$ obtained from Gaussian distribution $p(d) \sim N(\sigma, \mu^2)$. If the mobility prediction model is accurate, there should be $\sigma = 0$. Here, we set the bias of mobility prediction model ranging in $[0, 10]$. With different configured Gaussian distribution bias, we can obtain multiple mobility prediction models with varying accuracy level. As Figure 4.11(b) shows, the smaller the error is, the more accurate the mobility prediction model is.

As Figure 4.11(b) shows, *iLOCuS* is robust to the different accuracy level of mobility prediction model. Although *KL-divergence* increases when the prediction error is larger, *iLOCuS* always outperforms the benchmark methods. The *KL-divergence* gradually increases with increasing prediction error, and converges at around 8. The *KL-divergence* obtained at error of 0 is 71.57% of the *KL-divergence* obtained at error of 10. The increasing pattern of *KL-divergence* is because

the mobility prediction influences our estimation on each vehicle's positions in future duration, on which our incentivizing method is based. Thus, when the mobility prediction model has an error, *iLOCuS* may also incentivize vehicles in an inefficient manner.

Vehicle's acceptance rate

Theorem 3 proves that a driver can maximize the monetary utility by accepting the incentivizing assignment. However, in practice, it is possible that some vehicle agents may reject the incentivizing due to emergencies, lack of knowledge, or other preferences. We evaluate the effects of the driver's acceptance rate on the performance of *iLOCuS*. Assuming the acceptance rate r, after getting the incentivizing policy, each vehicle agent has a probability of r to accept the assignments. We conduct the accepting experiment 1000 times given each acceptance rate. Figure 4.11(c) shows the performance of *iLOCuS* under acceptance rate in the range of $[0.6, 1.0]$ at different times of the day.

As Figure 4.11(c) shows, the *KL-divergence* obtained at the acceptance rate of 100% ranges between $[88.01\%, 91.17\%]$ of that obtained at the acceptance rate of 60%, which shows the robustness of *iLOCuS* to different acceptance rate. DRP(*iLOCuS*$_1$, *iLOCuS*$_{0.6}$) ranges in $[9.68\%, 13.62\%]$, where *iLOCuS*$_i$ means the algorithm with acceptance rate of i. The *KL-divergence* decreases linearly with the increasing acceptance rate. All the p-values of the linear regression are less than 0.001, which shows that the linear trends are significant. This figure also shows that if the acceptance rate ranges around $[80\%, 100\%]$, we can ensure at least 93.69% of the performance compared to the 100% acceptance rate using our algorithm *iLOCuS*.

Besides the problem of acceptance, it is also possible that some agent accepts the assignment but does not adhere to the incentivized trajectory. There are two possible solutions to address this adherence problem: 1) if a vehicle agent is subjective malicious, we could exclude it from our vehicle pool. 2) for each candidate trajectory of the malicious vehicle agents, we can adjust the trajectory matrix into a probabilistic format. Based on the adherence history, we can assign a probability to the potential grids that the malicious vehicle agent may deviate from current trajectory to. Our algorithm will automatically evaluate the influence of this deviation to decide whether to incentivize the malicious agents.

4.6 Conclusion

In this chapter, we introduce *iLOCuS*, a vehicle incentivizing method to realize a target sensing distribution in mobile crowd sensing system using non-dedicated vehicle platforms. In mobile crowd sensing systems, the inconsistency between the goal of the data request end and the vehicle

agents results in the low quality of sensing distribution or requires a large budget. To this end, we formulate the problem into a multiple-choice knapsack problem with a non-separable convex objective function, by considering the budget constraints and physical mobility constraints. We use the task requests at the destination of the incentivizing assignment as a "hidden incentive" to reduce the cost of incentivizing vehicles. Since the formulated problem is NP-complete, we proposed an optimization algorithm to find a solution based on the insight of coordinate descent.

To evaluate the algorithm, we use real Beijing taxi data to show that our system always outperforms benchmark methods under different objective distributions, numbers of vehicles, budgets, lengths of the incentivizing period, incentive mechanisms, mobility prediction accuracy, and incentive acceptance rates. The results show that our algorithm can achieve up to 26.99% improvement in the quality of the sensing distribution compared to not incentivizing the vehicle agents.

In our future work, we plan to better characterize the specific mobility and acceptance rate patterns of each vehicle agent. To better learn the mobility and acceptance rate of taxis, we will use geographical functional zones to discretize the spatial locations instead of directly dividing the map into grids and then use history of acceptance behaviors to predict their future acceptance rate.

Chapter 5

Indirect Traffic Monitoring Using Ambient Building Vibration Sensing Systems

5.1 Problem Overview

Although sensing technologies become ubiquitous, it is often difficult to find proper sensors to collect the desired infrastructure information or can only prepare a limited number of sensors for the target task because of the deployment constraints and cost. For example, sensors like laser scanner, which directly measure the geometric distortion of railway tracks, are expensive and difficult to maintain for city-scale railway track health monitoring. In fact, for those city scale urban infrastructure systems such as transportation systems, finding a cost-effective sensor to deploy is especially difficult. This is because these transportation systems (e.g. railway systems, road system, bridge systems and etc.) often spread widely inside a city, which makes it difficult and costly to densely deploy sensors and maintain the sensing systems. For example, current practices of traffic monitoring has relied on directly instrumented systems [29, 166, 189, 190, 191, 208, 241]. In particular, dedicated traffic sensors are developed and deployed in many cities (e.g. vehicle detection loops, and traffic camera monitoring system are often used to monitor traffic events) [67, 105, 158]. However, these sensor-based methods require dense deployment of dedicated sensors that are costly both for deployment and maintenance. The demand of these dedicated sensors will increase with the increasing need to understand and optimize transportation systems. Furthermore, these systems are designed as "silo" systems, requiring one system to monitor each aspect of traffic, and thus further increasing costs.

One way to address this challenge is to conduct indirect sensing. The indirect sensing first utilizes the sensors installed on other infrastructure systems to collect the data during the interactions between the target infrastructure without sensors and other instrumented infrastructures. Then in-

direct sensing infers the states (e.g. health, events) of the target infrastructures by extracting the information related to the target infrastructures from the collected data. As an example, to address the challenge of lack of proper sensors for traffic monitoring, we can utilize the traffic-induced building vibrations collected from ambient building systems. With the increasing need of structural health monitoring, many of these buildings are being instrumented with vibration sensors to monitor their health [48, 51, 59, 121, 133, 204]. Ambient traffic events will induce the vibration of these buildings [93, 150, 151, 169, 170, 173]. Most information about the traffic events is carried by vibrations on the roadway and thus coupled with the buildings through the ground.

In this chapter, we present an algorithm using the building vibrations to monitor the train traffic around the buildings. In general, existing structural health monitoring systems collect building vibrations with sampling rate around 100Hz to 1000Hz [133, 204]. Our algorithm is developed to focus on the frequency range around 0-100 Hz, which falls within the typical structural vibration monitoring range. Our algorithm and evaluation results show that it is feasible to monitor the train traffic using ambient building vibration. By utilizing the existing building vibration monitoring systems, the need (and thus costs) of dedicated traffic monitoring systems will be significantly reduced. In general, this research provides an example showing the feasibility to expand the utilization of building sensor systems for multiple purposes. With the development of smart cities, more sensor systems will be deployed. This kind of multiple utilization will reduce the overall cost of the smart city sensing systems and/or improve robustness through redundancy.

To monitor traffic events using building vibrations, we need to understand how buildings respond to traffic events. We focus on two research challenges in this approach. First, the complexity of propagation media makes it difficult to apply physics-based methods, e.g. modal analysis, which requires detailed prior knowledge about the building. Second, the deployment of vibration sensor systems on different buildings do not obey the same criterion. Finally, the data is very noisy because of effects from human activities, machinery inside the building, etc.

To address these challenges, our approach uses causality between the vibrations in various locations of a building to represent the building vibration pattern, then detects and infers the train events around the building by looking at the response of the building vibration pattern to the train events. Causal analysis characterizes causality between pairs of signals. Recent advances in causal analysis show a new perspective to analyze the causality between signals in an information-theoretic approach, and has been applied in data compression, economic analysis and neuroscience [141, 181, 182, 193, 242]. The information-theoretic approach extracts the information exchanging between two locations of sensors as features to depict the vibration patterns in the corresponding physical interval. As a data-driven method, our approach eliminates the requirement for detailed prior knowledge about the structure. Meanwhile, causal analysis can extract the direction and

amount of information flow between pairs of vibration signals. Thus, we can detect the changes in information flow between multiple sensor pairs to extract building's response to ambient events. This method also provides physical insights about the effects of the excitation events on building vibration.

Our approach includes three steps: data collection, event detection and event inference. First, we collect building vibration data from multiple sensors. Second, the time of traffic events, which is the time interval when train pass by the building in our case, can be detected based on features extracted by wavelet analysis, and is effective for decomposing non-stationary signals. Third, we infer the types of traffic events, which is direction of train moving in our study, by causal analysis using information theory and machine learning techniques.

We evaluate our results through the vibration sensing system deployed in the 11-story tall building at Tsinghua University, Beijing, China, for over one month. This building experiences periodic nearby passing trains that are the target of this study. In particular, we focus on both the detection and the directional estimation accuracy of our algorithm under different conditions.

In this work, we have three key contributions as follows:

- We introduce a train event monitoring method using commonly deployed vibration sensors for building health monitoring;
- We analyze noisy building vibration patterns using an information theoretic approach and mining the information conveyed in the noisy vibration signals.
- We evaluate the algorithm through field experiments conducted for more than one month in a 11-story real building under train events.

The rest of this chapter is organized as follows: Section 5.2 discussed the related work of indirect sensing for urban infrastructure monitoring. Section 5.3 focuses on physical insights of vibration propagation and information exchanges inside building. Section 5.4 introduces our algorithm to detect when the train event happens and infer the direction of train in further steps. Section 5.5 shows the implementation of sensor system, results and analysis of vibration signals collected in a 11-floor building with periodic passing trains. Finally, Section 5.6 summarizes the conclusion.

5.2 Related Work on Indirect Sensing for Urban Infrastructure Monitoring

The indirect sensing for infrastructure health monitoring is first proposed by [228, 230]. The work extracted a beam natural frequency by using a model of spring mass travelling on the beam. The

model was latter tested in real-world experiment where the researchers drove an pre-installed car over a bridge to infer the bridge model parameters [126]. There are many following research conducted since then. The general objective of researcher is to find useful information extraction method for analyzing the indirect sensed data to detect the damage of the target infrastructure.

In the setup of indirect sensing, it is difficult to directly install sensors in the target infrastructures, e.g. a bridge/a light rail track. People instrument and sense the data from the infrastructures interacting with the target to acquire part of information about the target infrastructures. To improve the performance of indirect sensing, researchers have developed various approaches to model the interaction between the related infrastructures and target infrastructures, and extract key indicators from the collected sensing data to infer the conditions of the target infrastructures.

Recently, the idea of indirectly monitoring transportation infrastructures become a hot topic [140]. [229] applied empirical mode decomposition on the vibration response of a passing vehicle to generate the intrinsic mode functions to extract bridge frequencies for higher modes. The proposed method showed that the designed features can successfully extract the higher mode frequency of the target bridges. [22] proposed a method to assess the damage states of the bridge deck using the dynamic response of a vehicle passing through the bridge. The stiffness reduction in the bridge is identified using dynamic response sensitivity analysis with a regularization on the measured vehicle vibration signals. The method combined the vehicle acceleration noise, the roughness of bridge road surface, and uncertainty of the models. The damage identification results are acceptable. [145] investigated a simplified car-bridge interaction model for theoretical simulations to extract the natural frequency of the bridge from the spectrum of vehicle vibration signals. The researchers observed other structural properties like structural damping. [232] presented a new method of bridge damage detection by analyzing the vertical dynamic response of a vehicle passing through the bridge. This method utilized finite-element method to firstly simulate the interaction system, and then approximate the vertical vibration interaction between the bridge and vehicle to identify the damage indicator. The paper also discussed about the limitation of the proposed method. [201] tried to separate the influence of the velocity of driving, the natural frequency of vehicles and natural frequency of bridges to better identify the bridge damage state indirectly from the vehicle vibration response. The authors firstly described an analytical formulation and the respective finite element model to give some physical insights about the model. Then real-world experiments were conducted revealing a reasonable accuracy estimation of first natural frequency of the bridge.

Except the physical-model based approach to monitor bridge infrastructure using vehicle passing by, [114, 115, 116] also proposed to use moving light-rail to monitor the damage state of the light rail track. The researchers explored data-driven approach, instead of pure physical-model based method, to monitor the railway infrastructure using the dynamic response of the train on-

service. Instead of time-consuming and dangerous manual inspections, this work tried to find a more economical way for track inspection. By detecting the changes of high frequency component collected from the light-rail, the proposed method could provide preliminary information for the dedicated inspection vehicles about which areas should be focused on. The authors using data-driven approach to predict the rail infrastructure damage state. The results showed the accuracy of detecting two types of track changes, replacement and tamping was in satisfactory level.

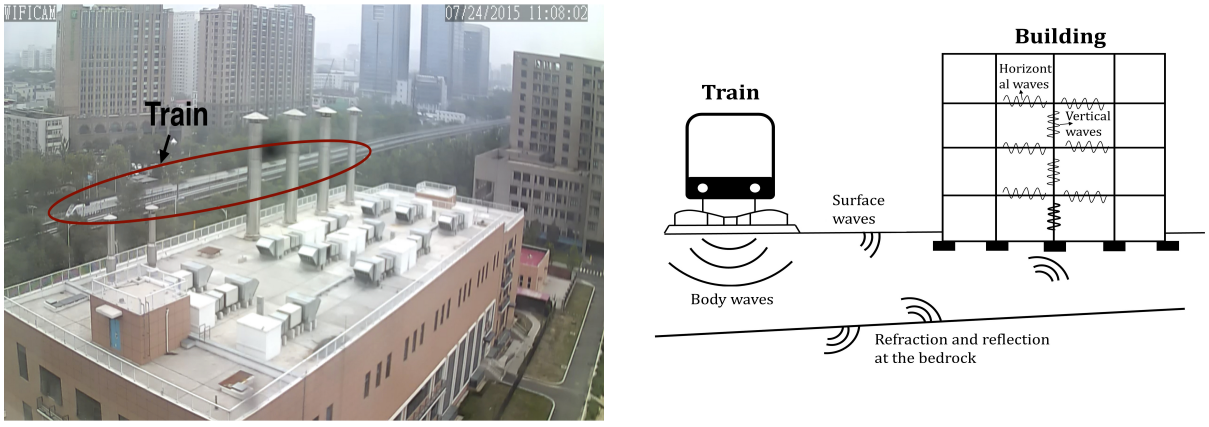
5.3 Physical Insight for Causal Analysis of Building Vibration

To better understand the problem, we discuss how building vibrations are induced by passing trains and propagate inside building structures, and then explain how these vibrations can be analyzed as information exchanges using information theoretic approaches. Here we assume that a train moves with a constant speed when passing by the building, and dynamic characteristics of the building structure remain constant (i.e., no damage is incurred by the train induced vibrations).

On the physical side of wave propagation, current studies on train-induced building vibration have focused on analyzing and predicting the effects of train-induced excitations on building structures using two main approaches: physics-based methods and data-driven methods. Physics-based methods use the physical principles of wave propagation to analytically or numerically analyze the effects of train-induced vibrations on buildings [3, 4, 70, 96]. However, these methods often make simplified assumptions to reduce the number of parameters (such as homogeneous propagation medium or require detailed prior knowledge about the ground and building structures). They may not be suitable in practice due to complex urban space and building structure. Data-driven methods are developed for analyzing structural response to train events based on historical train-induced vibration data. These approaches do not require detailed priori structural information, but may lack physical insights for observed phenomena (i.e. we can observe correlation but not physical causation) [13, 222].

In our scenario, the train excitation, which is a moving load source (as Figure 5.1(a) shows), generates Rayleigh waves in the ground that travel to nearby buildings [14, 34, 70, 72, 106, 125], as shown in Figure 5.1(b). The waves propagate inside a building in both vertical and horizontal directions. In each story, the vertical waves propagating through the building consist of two parts: up-going and down-going components [13]. The vibration propagates through columns to the adjacent stories, but there exists time difference between arrivals of waves from columns near the train and away from the train. As soon as these shear waves arrive at the next story, the horizontal waves begin to disperse through the floor from the column location. These waves are mixed with the reflected waves from the boundaries. The amplitudes, frequencies, phases, and other dynamic

features of these waves vary with the characteristics of train excitations and building structure medium that they have traveled through. Similar inference techniques have been used in other fields [114, 116, 153, 154, 155].



(a) (b)

Figure 5.1: (a) Train is passing by Rohm building (photo taken by the camera installed on the top of Rohm building); (b) Physical insights of wave propagation from train to building

Thus, the wave propagation from one place to another can be interpreted as an information exchange (or information flow) about building structures and train excitations between these two places inside a building. If the characteristics of the train excitation change (e.g., direction), the generated waves also change (e.g., time-frequency contents, phases, delays). Since different waves have different properties (e.g., speed, attenuation, reflection), the corresponding information exchange pattern between the two locations may also change. As mentioned above, this study assumes that the structural characteristics remain the same over time, and thus does not contribute to the changes in information exchange.

This work exploits the idea of representing wave propagation as information exchanges to introduce a new method that infers train event characteristics from building vibration response. This method uses an information theoretic approach to characterize the information exchange relationship between the wave signals collected from each pair of sensors, referred to as the causality between the signals. In particular, the concept of directed information is introduced to quantify the amount of exchanged information from one location to another, through investigating the wave propagation between them. This work is the first work to infer ambient traffic conditions based on building vibration response. This approach allows buildings to understand surrounding events in a cost-effective way using vibration monitoring systems that commonly exist in buildings (e.g., for seismic activity or serviceability monitoring). In addition, the advantages of data-driven approach

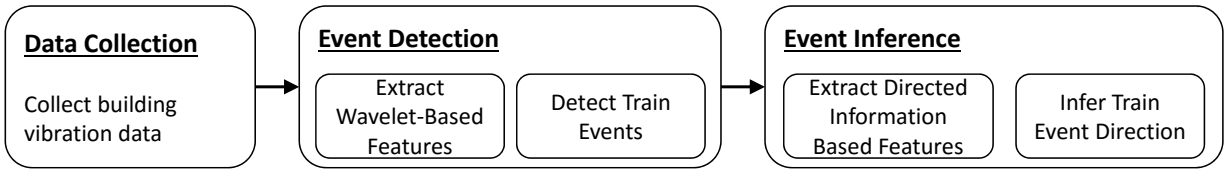


Figure 5.2: Framework of the proposed indirect train event monitoring algorithm

lie in the less requirement of a priori knowledge about physical properties of building structure. For different buildings, the absolute value of directed information between sensors will change with different natural frequency and damping ratio. However, our method focuses on the changes in the directed information value, instead of their absolute values, to infer train events. Thus, the algorithm is robust across different buildings.

5.4 Causal Analysis based Train traffic monitoring Algorithm

We introduce an information-theoretic algorithm to infer ambient traffic events using the causal relationship between vibration responses of building structures. This algorithm consists of three steps as shown in Figure 5.2: 1) collect building vibration data from different locations (Section 5.4.1), 2) detect the time interval of a train passing by the building using wavelet features (Section 5.4.2), and 3) infer the event characteristics based on directed information (Section 5.4.3).

5.4.1 Data Collection

In this section, building vibration responses are collected to characterize nearby train activities. Vertical acceleration responses are measured using accelerometers installed at various locations of building floors. We need at least two sensors for two reasons: 1) The algorithm for event detection combines data from multiple sensors to mitigate the effect of noise from specific concentrated areas; 2) The directed information represents the information exchanges between each pair of signals. More sensors provide higher spatial resolution for information flow patterns, which lead to more accurate and robust event inference. The locations of sensor deployment are suggested as: 1) deploy on the upper floors: upper floors tend to have stronger train-induced vibrations [222]; 2) spread out sensors: this mitigates the noise from concentrated areas; and 3) avoid areas with machines (e.g., elevator, fan) to reduce machine-induced vibration. Each measurement of sensor i at time t is denoted as $x_i(t)$. These vibration signals are pre-processed by being quantized into S levels for computational efficiency when calculating the directed information in Section 5.4.3. The exchanged information between different locations of building are extracted from pairs of

pre-processed vibration data.

5.4.2 Event Detection

To characterize the train event, the algorithm first detects the time interval during which a train passes by the building, referred to as an event. The wavelet transform is used to capture the characteristics of train-induced building vibration. We extract the features by combining the wavelet coefficients of vibration signals collected from all sensors on each floor. Based on the wavelet-based features, the train event is detected using machine learning techniques.

Extract Wavelet-Based Features

The building vibration signals are analyzed using wavelet transform to extract features that are sensitive to train excitations. Train-induced building vibrations are often non-stationary in nature, meaning that their statistical characteristics change over time. Thus, conventional signal analysis methods such as Fourier transform and auto-regressive modeling are not suitable. In contrast, wavelet transform uses wavelet bases that are localized in both time and frequency to represent signals. This allows wavelet transform to represent the time evolution of the frequency contents of the non-stationary signals. In addition, many classes of functions can be represented by wavelets in a compact way [35]. This compactness results in easier event detection because fewer features can represent the event of interest. Wavelet analysis has been widely applied as a promising tool to extract structural dynamic characteristics in structural health monitoring [25, 75, 163, 164, 165, 209, 234]. Similarly, we use wavelet to extract structural dynamic characteristics that change with train activities.

The wavelets are generated from a mother wavelet, $\Psi(t)$, by scaling and time-shifting:

$$\Psi_{s,\tau}(t) = \frac{1}{\sqrt{s}}\Psi(\frac{t-\tau}{s}) \tag{5.1}$$

where s is the scale factor and τ is the time-shift factor. Then the wavelet transform of the vibration signal $x_i(t)$ with respect to the wavelet function $\Psi(t)$ is defined as:

$$\gamma(s,\tau) = \int x_i(t)\Psi^*_{s,\tau}(t)dt \tag{5.2}$$

where $\Psi^*_{s,\tau}(t)$ represents the conjugate function of $\Psi_{s,\tau}(t)$

Since the signals are oscillatory, the average of wavelet coefficients over a small time window in each scale are used as features for each sensor data. For every time window, the feature is a

vector with the length of the number of scales used for the analysis multiplied by the number of sensors. The size of the time window and the number of scales are determined empirically and discussed with more details in Section 5.5.2.

Detect Train Events

Machine learning algorithms are used to classify the extracted features at every time window as train event or no event. In particular, we considered Support Vector Machine (SVM) and Random Forest as classifiers. SVM maps the features in the original finite-dimensional space into a higher-dimensional space, presumably making the classification easier in the new space when the decision boundary is non-linear [38]. SVM is also robust to noise by allowing outliers to be misclassified (i.e., ignore some data for classification if they are likely to be outliers). This allows us to classify data while rejecting outliers at the same time without using additional outlier detection methods. Random Forest is a decision tree style algorithm that uses randomized subsets of features to train a forest of decision trees and chooses the mode of their outputs as classification result [123]. The wavelet-based features of each time window are input to classifiers. The corresponding training labels are whether the train events happen in each time window. The classifiers learn the mapping between the features and the labels using the training dataset. Then with input feature of a new testing sample, the classifiers determine whether the corresponding building response in the time window is influenced by the train event.

5.4.3 Event Inference

When a train event is detected, directed information between each pair of sensor signals is used for inferring the characteristics of the train event. The directed information quantifies information exchange patterns or causality relationship among vibration signals at multiple locations in the building as features. Then classification method is used to categorize the event characteristics.

Extract Directed Information Based Features

The causality relationship between vibrations at different sensor pairs represent the information exchange patterns between these signals. Conventional vibration signal analysis methods often extract vibration characteristics from each sensor individually. But these methods demand dense implementation of sensors for fine-grained understanding of the structure and events [101, 133, 187, 223]. In contrast, the causal analysis extracts relational information between different sensors' measurements, which provides richer information than that from only a single sensor measurement. The causality inferred from vibration signal represents the directionality of wave propagation in

the structure under various loading conditions. Analyzing the vibration data directly to infer wave propagation information is difficult in practice due to a high level of noise. To this end, we utilize information theory to quantify the causal relationship. The information theory is based on the probabilistic modeling of information (or uncertainties) in random variables (or processes). Thus, this approach is suitable for dealing with noisy vibration signals with high uncertainties.

Directed information quantifies the causality between two stochastic processes (i.e., a sequence of random variables) using an information theoretic approach. A key measure in information theory is entropy which quantifies the amount of uncertainties (or lack of information) in random variables [10, 89]. For example, let X and Y be random numbers representing the numbers from tossing fair 4-sided and 12-sided dies, respectively. X has higher predictability than Y. Thus, the entropy of X is smaller than the entropy of Y. When the two random variables or processes are dependent (i.e., knowledge of one process provides information about the other process and vice versa), mutual information can quantify the shared information between them [37, 200]. For example, let another random variable $Z = 1$ if $Y = 1$ and $Z = 0$ otherwise. Then knowing Y increases the predictability of Z and vice versa because of the shared information between them. This shared information is computed as the gained information (or reduction in uncertainties) about one variable due to knowing the other variable, typically referred to as mutual information. Mutual information is a symmetric measure (i.e., the gained information about Y by knowing Z is same as the gained information about Z by knowing Y). On the other hand, directed information is an asymmetric measure that quantifies the directionality in the dependency between two stochastic processes, quantitatively depicting the causality (or feedback) relationship between them [142].

In our problem, we define two stochastic processes $X_i(t)$ and $X_j(t)$ to represent building vibrations at sensor locations i and j, respectively, and then define the directed information between them using their joint probability density function (PDF). If $X_i(t)$ and $X_j(t)$ are independent, their joint PDF is equivalent to the product of their marginal distributions for all times $n_1, \cdots, n_N$,

$$P(X_{i,n_1:n_N}; X_{j,n_1:n_N}) = P(X_{i,n_1:n_N})P(X_{j,n_1:n_N}), \tag{5.3}$$

where $X_{i,n_1:n_N}$ and $X_{j,n_1:n_N}$ represent the collections of $X_i(t)$ and $X_j(t)$, respectively, for $t = n_1, \cdots, n_N$. Then, the mutual information of X_i and X_j ($I(X_i; X_j)$) is quantified as the distance (or information discrepancy) between the joint PDF and the product of the marginals by using the concept of Kullback-Leibler divergence [47, 215] (i.e., the mutual information measures the degree of dependency). The distance here represents the information gain that we revised our belief from

X_i and X_j are independent to that X_i and X_j are dependent:

$$I(X_{i,n_1:n_N}; X_{j,n_1:n_N}) = E[log\frac{P(X_{i,n_1:n_N}; X_{j,n_1:n_N})}{P(X_{i,n_1:n_N})P(X_{j,n_1:n_N})}]. \tag{5.4}$$

Mutual information is always non-negative. And mutual information will be zero when $X_i(t)$ and $X_j(t)$ are independent. This mutual information does not represent any directionality in information flow. To introduce directionality, an alternative factorization in terms of the joint PDF is proposed. This factorization can represent the information feedforward and feedback [142]:

$$P(X_{i,n_1:n_N}; X_{j,n_1:n_N}) = \overleftarrow{P}(X_{i,n_1:n_N}|X_{j,n_1:n_N})\overrightarrow{P}(X_{j,n_1:n_N}|X_{i,n_1:n_N}), \tag{5.5}$$

where $\overleftarrow{P}(X_{i,n_1:n_N}|X_{j,n_1:n_N}) = \prod_{k=1}^{N} P(X_{i,n_k}|X_{i,n_1:n_{k-1}}; X_{j,n_1:n_{k-1}})$ and $\overrightarrow{P}(X_{j,n_1:n_N}|X_{i,n_1:n_N}) = \prod_{k=1}^{N} P(X_{j,n_k}|X_{i,n_1:n_k}; X_{j,n_1:n_{k-1}})$.

Similar to the definition in Chapter 2, the directed information from X_i to X_j is defined as the information discrepancy between the true joint PDF and the PDF computed as if X_i depends on X_j but not vice versa. Thus, the directed information is defined as:

$$I(X_{i,n_1:n_N} \to X_{j,n_1:n_N}) = E[log\frac{P(X_{i,n_1:n_N}; X_{j,n_1:n_N})}{\overleftarrow{P}(X_{i,n_1:n_N}|X_{j,n_1:n_N})P(X_{j,n_1:n_N})}]. \tag{5.6}$$

The directed information is smaller than or equivalent to the mutual information. When X_j does not depend on X_i, $\overrightarrow{P}(X_{j,n_1:n_N}|X_{i,n_1:n_N}) = P(X_{j,n_1:n_N})$, and directed information is equal to the mutual information. Note that $I(X_{i,n_1:n_N} \to X_{j,n_1:n_N}) \neq I(X_{j,n_1:n_N} \to X_{i,n_1:n_N})$. With transform on equation (6), the directed information can also be expressed using entropy and conditional entropy:

$$I(X_{i,n_1:n_N} \to X_{j,n_1:n_N}) = H(X_{j,n_1:n_N}) - H(X_{j,n_1:n_N}||X_{i,n_1:n_N}) \tag{5.7}$$

Where

$$H(X_{j,n_1:n_N}||X_{i,n_1:n_N}) = \sum_{k=1}^{N} H(X_{j,n_k}|X_{j,n_1}^{n_{k-1}}, X_{i,n_1}^{n_k}) \tag{5.8}$$

The entropy $H(X_{j,n_1:n_N})$ and $H(X_{j,n_k}|X_{j,n_1}^{n_{k-1}}, X_{i,n_1}^{n_k})$ here are functionals of the discrete distribution of variables $X_{j,n_1:n_N}$ and $X_{j,n_k}|X_{j,n_1}^{n_{k-1}}, X_{i,n_1}^{n_k}$ for $k \in \{1, \cdots, N\}$. When estimating directed information, we use Equation (7) for computational efficiency, instead of Equation (6) that involves estimating joint PDF. The entropy values are estimated using the minimax rate-optimal estimators under $l2$ loss [94, 95]. The minimax estimator minimizes the maximum loss function between estimator and functional of real distribution. The loss function is $l2$ norm of difference between estimator and functional of real distribution. We use empirical D-tuple joint distribution based on

the collected data to estimate the functionals of real distribution, and it has been proved that empirical joint distribution of D-tuple converges to the true joint distribution [94, 95]. The estimator converges faster and has less mean square error than conventional MLE (Maximum Likelihood Estimator) [95].

Using the building vibration data collected from different locations, directed information between all the sensor measurement pairs are computed to obtain the causality between them. Based on the known sensor locations, we can construct the building vibration pattern graph under different train events. The difference between the directed information from sensor i to j and their inverse directed information (i.e., directed information from sensor j to i) is defined as feature. If the feature has a high absolute value, the information exchange from sensor i to j has strong directionality. The feature with value close to zero represents weak directionality.

Since the directed information is computed at every time point, the feature dimension is high when data sampling frequency is high. Thus, we use windows to segment the original directed information features and take an average value for each window. The window sizes are determined empirically as investigated in Section 5.5.3.

Infer Train Event Characteristics

This work uses different train direction to illustrate the event characteristics. The train direction is classified using the directed information based features extracted from multiple sensor measurement pairs. Supervised machine learning approach is used for this purpose. The main challenge of our problem is the high dimensionality of the directed information based feature. When the number of training sample is relatively small, the classification algorithm performance may become unstable. To address this challenge, we use kernel support vector machine for classification [41, 199]. This method has an advantage in handling high-dimensional data.

5.5 Evaluation

In this section, we evaluate our algorithm by deploying our system in a 11-story building with a light rail line running 100 meters away. During a three-month period, we collected approximately 10GB of vibration data, and 450 hours of video for ground truth. The vibration data are collected from four groups of sensors deployed on the 11th floor of building. Then we extract wavelet based features and use machine learning to detect the time interval of a passing train. Finally, with the detected time interval, we extract the information exchanges inside building during train passing to infer the directions of train.

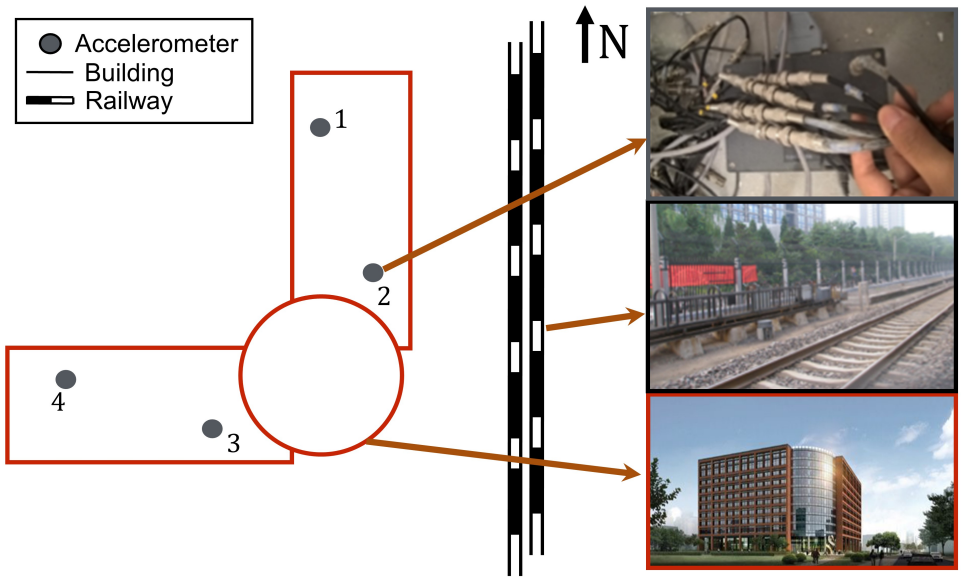


Figure 5.3: Building, location of train track, and sensor deployment (vertical view)

5.5.1 Rohm Building Experiment Setup

We evaluate our algorithm for one month on a train-building system with nearby light rails. Here we give a brief description of the data collection system.

The deployment is done in the newly completed Rohm building at the Tsinghua University campus located in Beijing China. It is a 11-story building with 3 basement levels, totaling 350,000 square feet. The building houses of Electronic Engineering Department of Tsinghua, and about 1000 occupants work in the building. This building is located around 80 to 140 meters away from a railway (10 trains per day) and Beijing light-rail line 13 (1 million daily riders, 5-11 minutes between two trains) respectively. These lines provide a regular active excitation to the building. The left side of Figure 5.3 shows a top down view of the deployed building. The center of the figure shows the train track location and the right side of the figure shows the photos of the sensor (top), light rail track (middle), and the outside view of the building (bottom).

The building is deployed with 40 high-quality accelerometers. These sensors are embedded in the floor of the building to monitor vibrations. The network of accelerometers is distributed over the building to record its vibration at different locations and directions. The accelerometers are divided into 16 groups and deployed on the 1^{st}, 5^{th}, 8^{th} and 11^{th} (top) floor of the building, with

4 groups on each floor. These sensors are deployed at the same location per floor (approximate location shown on left side of Figure 5.3). Coaxial wires are used to connect the sensors to a data collector on each floor, which sends the vibration data to a central server through local area networks. In this experiment, we utilized only the 11th floor sensors as it gives the largest swing and highest sensitivity to our measurements. We collected data on weekdays from June to August in 2015.

The precision of our single-axis sensor is $10^{-5}m/s^2$, with highest sample rate of 1024 samples per second. In our experiments, we collected data at 200Hz totaling 400 million data points through the period of the experiment due to storage limitations.

There are trains passing by building in two opposite directions daily: from North to South, and from South to North. About 70 trains with heavy loadings including cargo and thousands of passengers pass by the building daily. Throughout the experiment we observed roughly 1200 train passes. These vibrations are not strong enough to be directly perceived by humans.

5.5.2 Event Detection Results and Discussion

This section presents the train event detection results. We first characterize the wavelet-based features extracted from building vibration signals under different train events. Then we explain how to find the ground truth for evaluation using camera data. Finally the accuracies of event detection using two machine learning techniques are presented and discussed.

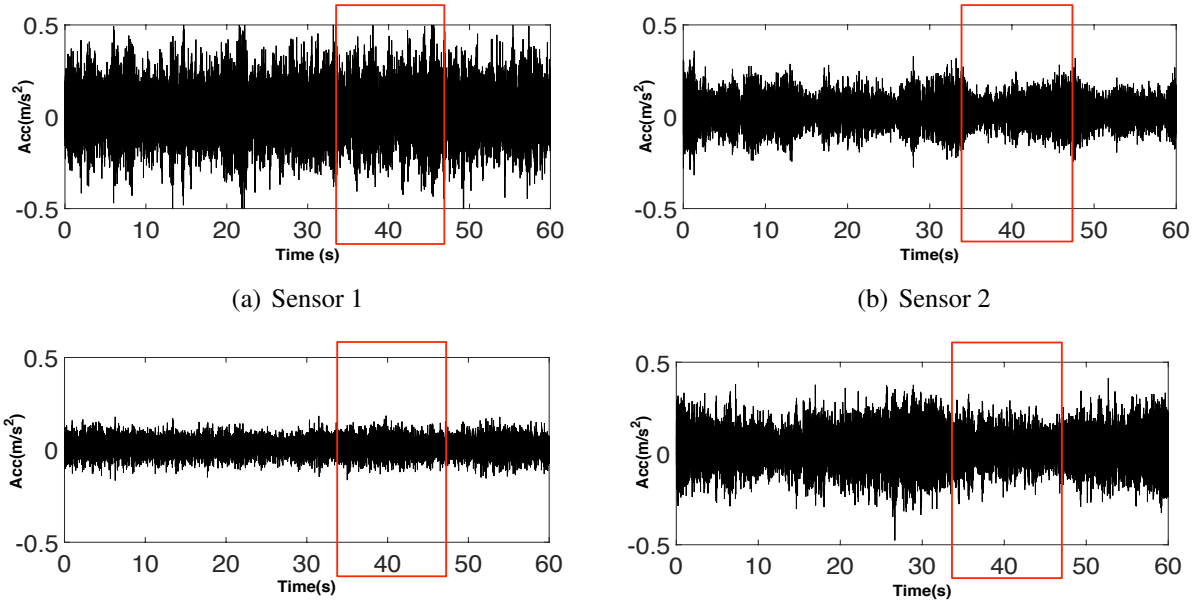
Characterization of Wavelet-Based Features

In this section, we show the characteristics of the data to better understand the signals. We first show both the time and frequency domain of vibration signals through a train event. Then we show our wavelet transform technique provides multi-resolution to show the existence of train event around building by capturing characteristics of vibration.

We give an example of signal segment in Figure 5.4 to illustrate the general nature of the captured signal. The data is collected from 4 sensor groups described above at the time when a train is passing. Based on the camera data, we manually labeled the train passing time at 32 to 48 seconds. As Figure 5.4 shows, the measured vibration signals are quite noisy, and the train event is not obvious well bellow the noise floor as to make it not immediately visible. This is due to the high amount of environmental noise in the setting including outside traffic, heating and cooling systems of the building, weather, building occupants, etc.

Figure 5.5 shows the same raw signals using Short-time Fourier Transform (STFT) with window length of 0.25 seconds. Although the general vibration frequency of current building is visible

in this figure, the train event can not be seen from the time-frequency features. This is due to the fact that short time Fourier Transform decomposes signals in fixed frequency resolution while the train generates multiple frequencies. Note that although the low frequency area (around 10Hz) in Fourier spectrum of sensor 3 signal shows high energy values indicating an event happening at around 40 - 42s, using this feature would incur frequent false positives and negatives because 1) the time duration of high energy area is short; 2) there exists many other peaks of similar magnitude and durations; and 3) the high energy area does not appear in the spectrum of any other three sensors. Although the window size for STFT may be tuned for each frequency to optimize the visibility of the peak, it is difficult to apply different window size for different frequencies for STFT, and it requires multiple parameter tunings which makes the algorithm inefficient.



(a) Sensor 1

(b) Sensor 2

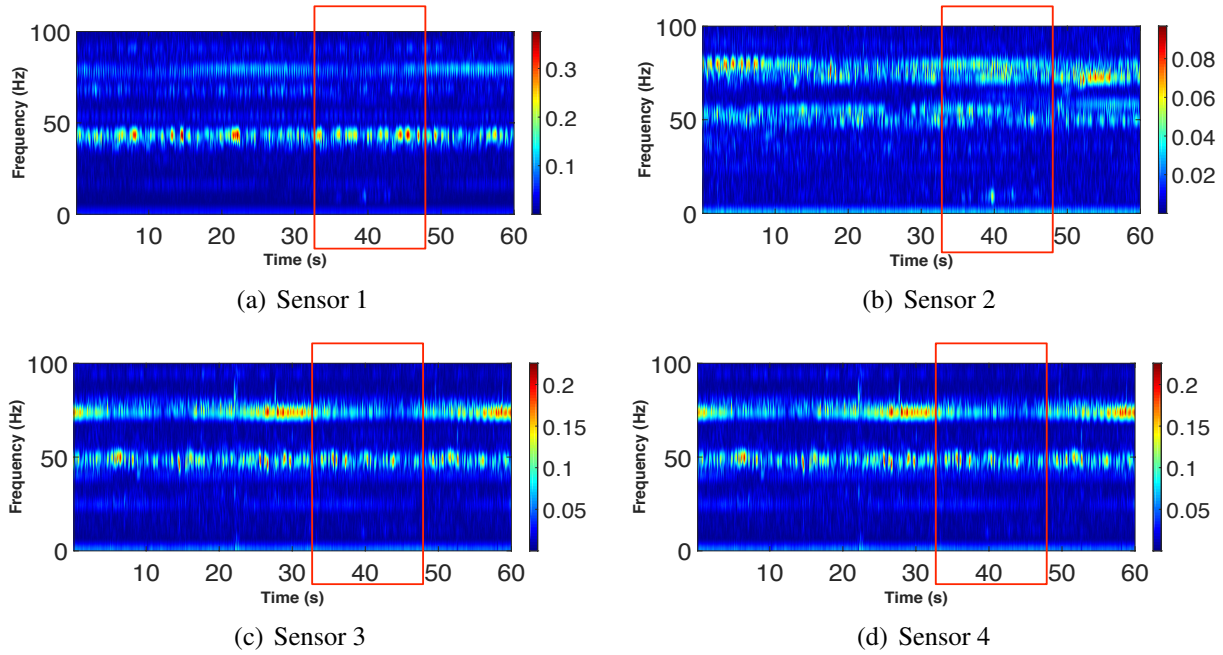
(c) Sensor 3

(d) Sensor 4

Figure 5.4: Vibration signals for 60 seconds collected from 4 sensors deployed on the 11th floor as shown in Figure 5.3. The train passed by the building at 32 to 48 seconds (as red boxes show). The building vibration due to train is not obviously visible due to the background noise.

Wavelet analysis results of two vibration data series produced by 4 sensors using Continuous Wavelet Transform (CWT) are shown in Figure 5.6. Here we choose scale from 1 to 32, which includes general frequency of Rohm building vibration of 10Hz to 80 Hz. The larger coefficients indicate stronger similarity between the signal and the analyzing function. As Figure 5.6 shows, the wavelet coefficient under the train event is significantly higher than wavelet coefficient at the time with no passing trains (compared to the raw time-domain signal in Figure 5.4 and short Fourier transform shown in Figure 5.5). In particular, the coefficient corresponding to the train passing

event is most prominent from 8 to 24 in scale. The result shows the advantage of wavelet analysis in detecting the train-induced slight changes of vibration signals.



(a) Sensor 1

(b) Sensor 2

(c) Sensor 3

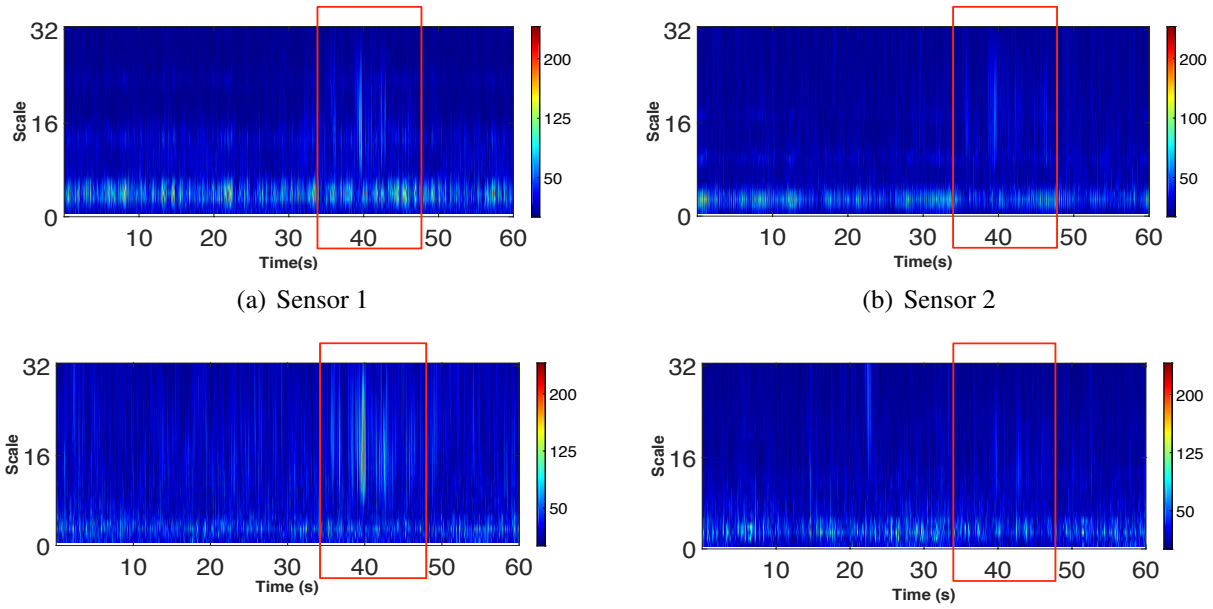
(d) Sensor 4

Figure 5.5: Short time Fourier Transform (STFT) of vibration signals shown in Figure 5.4. The red boxes show the time duration of train passing. Similar to the vibration signal, the train event is not obviously visible.

Train Event Labeling

To perform the evaluation, we need to properly label the segments of the vibration input using ground truth (video). In our case, we recorded the train passing time as the time when the head of train arrive at one side of building, until the time that the end of train leave the other side of building. In our experiments, we observed that it took 12 to 20 seconds for a train to pass by the building depending on its speed and length. This time interval recorded by visual inspection may not represent the "true" time interval when the vibration excited by train will influence the building. This is due to the difference of speed of visual light vs. ground vibration, and different strength of influence due to the ground composition. We calculate and remove this time delay interval by using cross correlation to calculate the delay time between when the train passed by the building and when the building got affected.

To obtain the ground truth of the train passing, we recorded a total of 450 hours of usable video (image capture shown in Figure 5.1(a)). Using the video, we obtained manually a binary



(a) Sensor 1

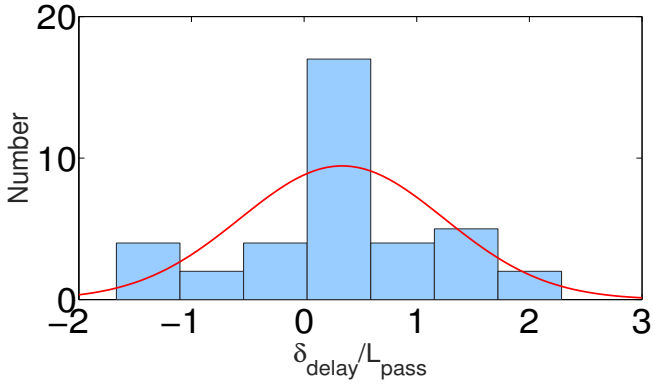
(b) Sensor 2

(c) Sensor 3

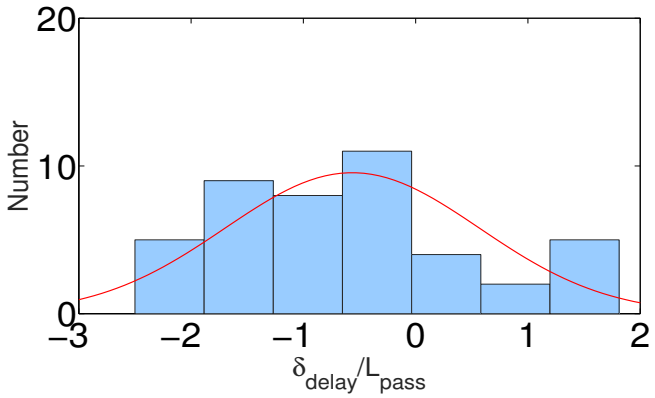
(d) Sensor 4

Figure 5.6: Wavelet coefficients (scaled from 1 to 32) of the building vibration signals shown in Figure 5.4. The red boxes show the time duration of train passing. The train event is clearly visible in the data from all four sensors between 38 and 45 second.

observation sequence indicating whether the train pass by the building at each time fragment of one second and its direction. Then, we divided the vibration data into fragments of the same length and use average wavelet coefficients over scale 8 to 24 as the indicator, which contains the information about the influence from train. Through calculating the cross-correlation between observation sequence and influence sequence. The time shift with which the maximum cross-correlation value appears is regarded as the maximum possible delay between the two sequences. Because of the velocity and length of train changes, the total time of train passing by the building are also different. Here, we calculated the ratio of delay time to total time length to see the distribution of delay time. Figure 5.7 shows the histogram of delay time ratio's distribution. Specifically, Figure 5.7(a) and Figure 5.7(b) separately show the relationship between delay time and directions of the train. As expected, the train from south to north tends to get positive delay time ratio and the train from north to south tends to get negative delay time ratio. We can use these factors to calibrate the delay time based on visual time and delay time, and get the ground truth about when the vibration of building was affected by train.



(a) Ratio of delay time from South to North



(b) Ratio of delay time from North to South

Figure 5.7: The histogram of 82 cases' ratio of delay time to total time of train passing, which is calculated by cross correlation. Y-axis represents the number of cases, X-axis represents the ratio of delay time to total time length, red line is the pdf curve of corresponding fitting normal distribution.

Train Event Detection Results

Using the wavelet results, we input the extracted wavelet coefficient as features to classification model to detect whether the train passed by or not at corresponding time interval.

After extracting the wavelet coefficient as features, we use supervised classification on time fragment for train event detection. Here, we considered Support Vector Machine (SVM) and Random Forest (RF) as classifiers.

The average of wavelet coefficients over different time fragments are calculated as features on each scale and each sensor. For every time unit, we extracted the average wavelet coefficients over scale 8 to scale 24 from each sensor. Thus, we will have $t \times 4$ dimensional features, where t is the number of time unit. For the training set, the output is a binary label sequence indicating whether there is a train passing during corresponding time, or not.

As a baseline to our algorithm described in the previous section, we extract dynamic frequency domain features to identify the effect from traffic events. This is extracted using Fourier analysis. We analyzed the peak frequencies with a moving window of 1 second on the vibration data. We calculate the corresponding power spectral density, and record the frequency and PSD of the 5 largest peaks.

Our dataset includes the vibrational data collected in the building and video ground-truth data. The dataset was down selected to only include train passing through without light rail passing within 1 minute before and after to simplify the problem. This includes 55 cases when there is a passing train, and 60 cases when there is no passing train. 90 (45 train, 45 no train) samples are used to train binary classification models, and the remaining 25 cases are used to test the accuracy of classification models. We labeled the samples by combining collected visual data and delay time.

With sample rate of 200 Hz and around 16s-long time series, the number of features is always much higher than sample number, which is unbalanced and makes it difficult to train models. As discussed in Section 5.4, we use a sliding time window to increase training accuracy. Figure 5.8 shows the classification accuracy of the two algorithms (SVM vs. RF) using both feature selection methods (wavelet and Fourier) over time. The performance of random forest is better than that of a tuned support vector machine. The optimal window length is around 0.25 seconds. We can see that when we use all wavelet coefficients during train pass by as features, the accuracy is low due to small number of training samples compared to number of feature. And when the window length is too long (e.g. 1 second) the accuracy also decreases because at this window length some information begins to be lost for event detection.

Figure 5.8 also compares the two feature selection methods, one is based on wavelet analysis, another is based on Fourier analysis. We see that the accuracy of wavelet-based feature is

Table 5.1: The True Positive, True Negative, False Positive and False Negative of Train Event Detection

		Ground Truth	
		Train Pass	No Train Pass
Prediction Result	Train Pass	93.3%	20.0%
	No Train Pass	6.7%	80.0%

consistently higher than the accuracy to use Fourier-based feature. The highest accuracy (wavelet with random forest) of 88% is achieved at window length of 0.25s based on features of wavelet coefficient and Random Forest classification models.

We further present the confusion metric of the prediction result and ground truth in Table 5.1. From the table, we see that the train passing event is correctly classified 93.3% of the time, corresponding to 6.7% false negative. Similarly, the no train is classified correctly 80% of the time, corresponding to 20% false positive. This is likely due to the location of the testing building being next to both rail and large roads. The effect of the train event is fairly similar to existence of other large traffic. Thus, during training, we only identified tests that has effect and no train and did not train the difference between other large traffic events (trucks, buses. etc.). This similarity raises the false positive events.

To see the sensitivity of the detection due to the system, we also analyzed the effect of sample rate on train event detection accuracy. By sub-sampling the vibration signals from a sample rate of 200Hz, we simulate collecting vibration data with sample rate of 100Hz, 50Hz, 25Hz and 1Hz. As Figure 5.9 shows, when the sample rate is low, e.g. 1Hz, the highest train event detection accuracy is only about 56%, which is close to 50% accuracy baseline of blind classification. When the sample rate increases to 25 Hz, the accuracy achieved 84%, which is 28% higher than the accuracy on sample rate of 1Hz. However, with sample rate increases further to 50Hz and 100Hz, the accuracy drops. This is likely due to that the majority of train vibration information exists in the lower frequencies, i.e. near 25Hz. For example this could include general rumbling created by the train, or a cart in the hallway. However, for increased accuracy, the distinctive features of the train vs. no train passing is at higher frequencies. The initial drop in accuracy could be due to the increased noise introduced initially at the mid frequencies.

5.5.3 Event Direction Inference Results and Discussion

In addition to the existence of the train events, we further explore the results of directional inference of the train through the use of directed information between the multiple sensors.

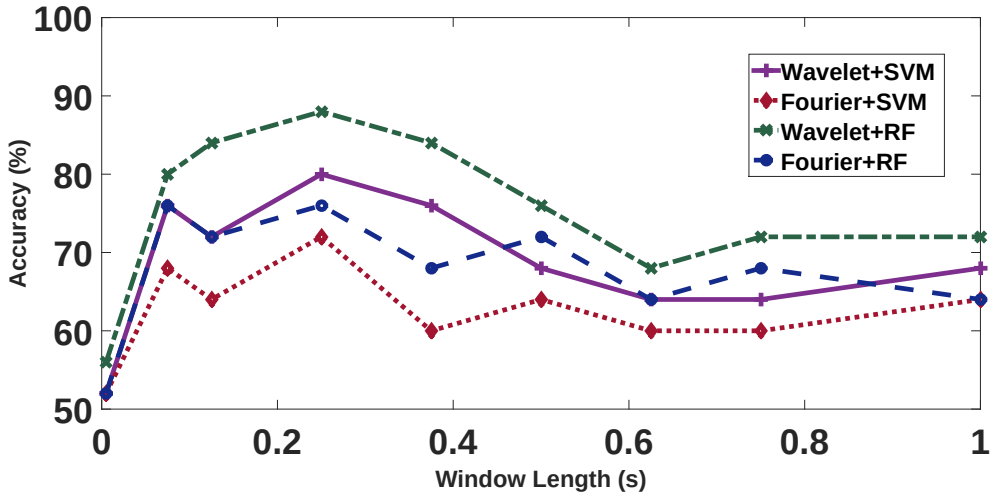


Figure 5.8: The figure shows how accuracy of different methods changes with increasing window length. "Wavelet+*" represents we use wavelet analysis to extract feature; 'Fourier+*' represents using Fourier analysis to extract feature; SVM and RF represents the classification method Support Vector Machine and Random Forest.

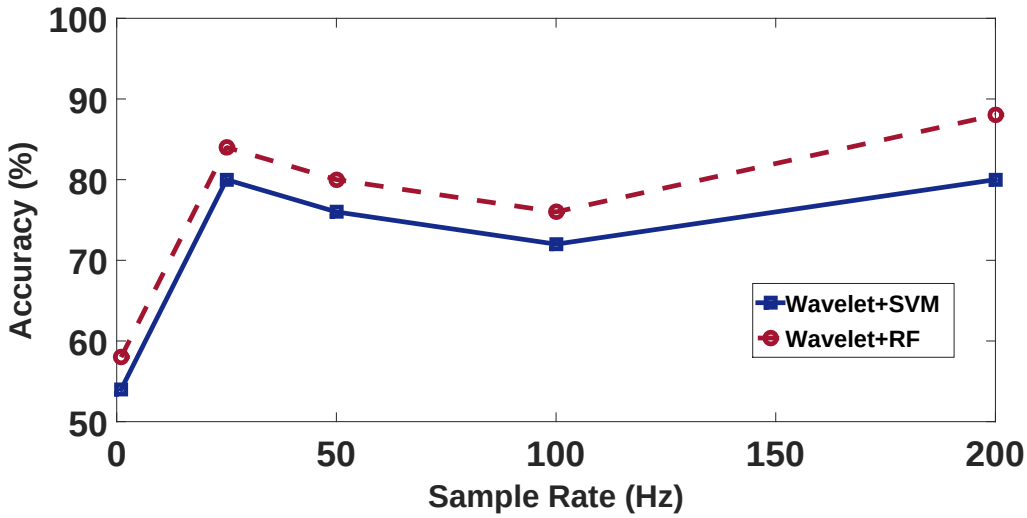
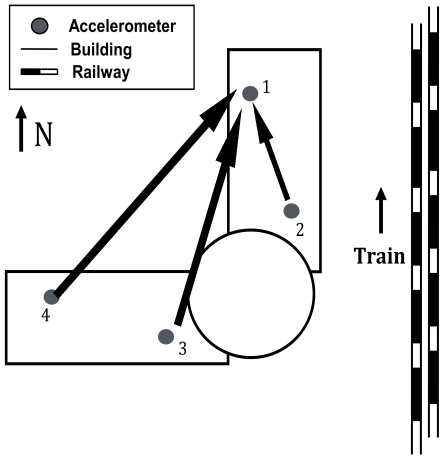


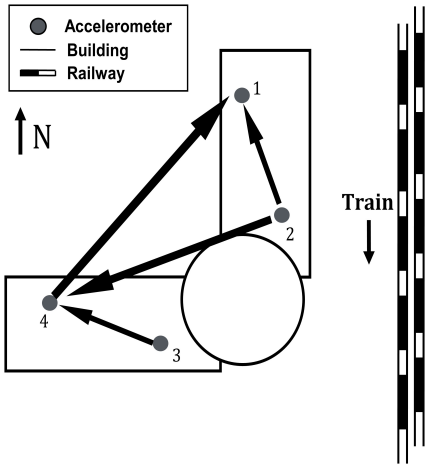
Figure 5.9: Accuracy under different sample rates based on (1) red line: wavelet coefficients as features and Support Vector Machine as classification model; (2) blue line: wavelet coefficients as features and Random Forest as classification model.

Characterization of Directed Information Features

In order to characterize beyond the existence of the train, we explore the directionality of the train event. In this evaluation, we use the same 4 sensors on the 11th floor of the building recording



(a) Train from South to North



(b) Train from North to South

Figure 5.10: (a) Directions of information propagation among 4 sensor groups on the 11th floor when a train passes from South to North; (b) Directions of information propagation among 4 sensor groups on the 11th floor when a train passes from North to South.

vertical vibration to investigate their influence on each other (see Figure 5.3). These 4 sensors will have 6 paired combinations (sensor 1 vs 2, sensor 1 vs 3, sensor 1 vs 4, sensor 2 vs 3, sensor 2 vs 4, sensor 3 vs 4). First, we found that the building vibration patterns without a passing train are rather random and show no consistent patterns in directed information results. Thus, there is no inherent bias in the information propagation without the train.

In our case, the signals will be quantized into 10 levels before calculating directed information. The algorithm introduced in Section 5.4 can detect the time and duration of train events automatically. Then we extract the vibration signals during train events and quantize these signals. In general, to effectively estimate the directed information, a large number of quantization level is desirable, since with large quantization levels, the signal amplitude range for each quantization level is small (i.e., higher signal resolution) such that more information contained in difference between signals can be extracted. However, this level cannot be too large because of the limitation of $m \approx \frac{S^{D+1}}{lnS}$. m is the sufficient sample number for calculating directed information between two signals and $m < n$, where n is the total sample number. So we decided maximum quantization level of 10 based our current total sample number and reorganized the data of sensor 1 and 2 into two $m \cdots (n-m)$ matrix. The matrix will be input to calculate the directed information along time series.

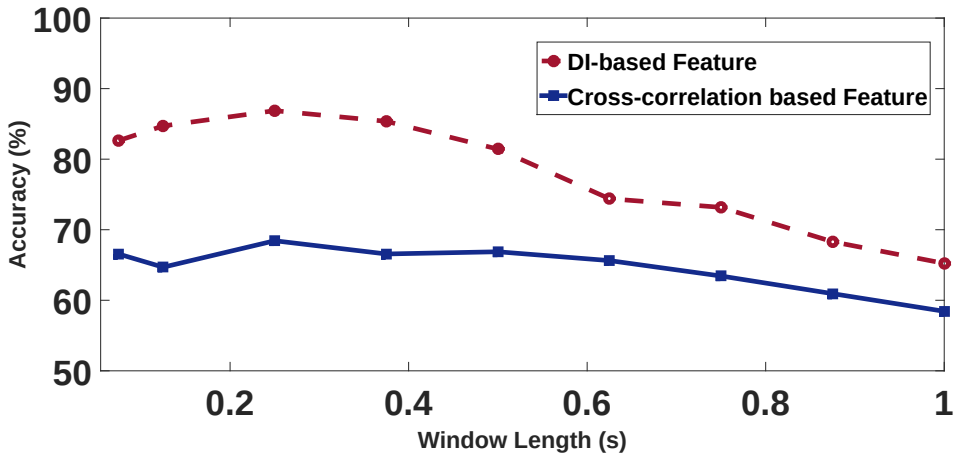


Figure 5.11: Accuracy of train direction prediction; (a) red line: using directed information as features; (b) blue line: using cross-correlation coefficients features.

We constructed the vibration pattern with train in two different directions, northbound and southbound. Here, we use simple threshold to find the direction of information exchange between sensor pairs based on the extracted directed information and inverse directed information. For

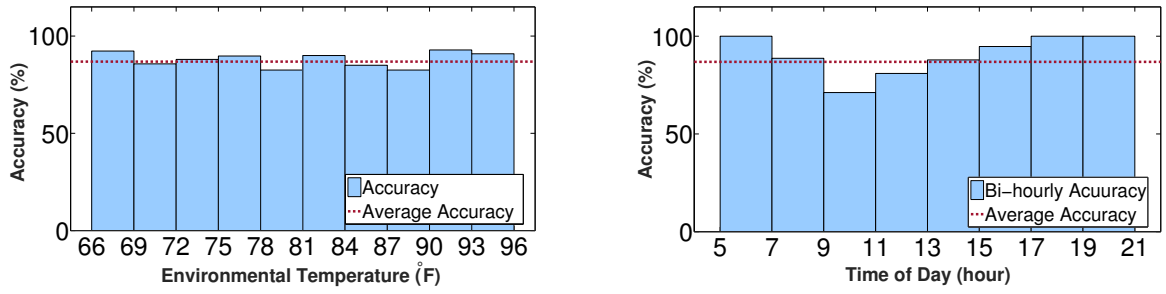
example, given sensor 1 and sensor 2, if the absolute value of $I(1 \to 2) - I(1 \to 2)$ is less than threshold, we regard the case as no directionality in the information exchanges between sensor 1 and sensor 2. If $I(1 \to 2) - I(2 \to 1)$ is large than the threshold, we regard the direction is from sensor 1 to sensor 2; If $I(2 \to 1) - I(1 \to 2)$ is larger than the threshold, the direction will be from sensor 2 to sensor 1.

Figure 5.10 shows the dominant vibration patterns when there are trains passing by the building in different directions. Figure 5.10(a) shows the north-bound train, and Figure 5.10(b) shows the vibration pattern for the south-bound train. The arrow shows the direction of the propagation of information using directed information. More than 40% N-S cases have the same pattern as Figure 5.10(a) shows, and more than 45% northbound cases have the same pattern as Figure 5.10(b). Other patterns follow this general trend. For the north-bound trains, the vibration is first observed from the southern end of the building (sensors 2, 3, and 4), and propagate to the northern part of the building. For the south-bound trains, the vibration information propagates from east (sensor 2, 3) to west (sensor 4).

Over the whole train passing process, the building vibration is a very complex process with coupling of excitation source and structure properties. In the process of wave propagation, the reflection, interference and diffraction between wave and structures are very complicated. Besides, the train here is a moving line excitation source, which make the vibration pattern extracted based on the overall process of train passing more complex. It is hard to predict the vibration patterns inside building intuitively by matching them with the train direction. However, since the time of wave arrivals on different sensors varies with trains direction, we can conclude that the vibration patterns are different under different directions of train passing by the building. The change of pattern revealed in two causal graphs demonstrates not only the event's influence on the building's vibration characteristics, but also how the building reacts to different kinds of vibration in different ways.

Train Event Inference Results

Based on extracted directed information features, we used a classification model to predict the directionality of train events. Here we label each sample using the direction of the passing train during the corresponding time interval. We also need to decide the optimal window length of generating samples: when the window length is too short, the uncertainty of value of directed information will affect the extracted feature significantly; when the window length is too long, we may lose the details about the response of information exchanges inside building to train events during the time window. In our method, support vector machine is applied to classify the direction based on the extracted directed information features.



(a) (b)

Figure 5.12: Accuracy of train direction prediction (a) under different environmental temperatures between 66 and 96 °F; (b) at different time of day from 5am to 9pm

To evaluate our classification system, Figure 5.11 shows the accuracy to infer the direction of train passing by the building using different sliding window lengths. The red line shows the results of our directed information approach, while the blue line uses cross-correlation coefficients between sensor pairs. The results show that the accuracy decreases with large window lengths. That is because longer window length loses time resolution of features. Figure 5.11 also shows that given the same classification method, our directed information based features achieve the directional accuracy of up to 86.9%, compared to 66.9% when using the cross-correlation based method. This correspond to a 2.5X reduction in error.

In this work, we evaluate the robustness of our algorithm to some factors (e.g. weather condition, temperature, indoor people activity, other traffic noise, etc.), by collecting data on different days and different time of a day with various temperature, noise level, peoples activity, and weather conditions, including rainy, sunny, and cloudy weather [112, 149, 171, 172]. Figures 5.12(a) and 5.12(b) show the algorithm accuracy for different temperature level and time of the day, respectively. Although the temperature may cause changes on the physical properties of building structure, we found that our algorithm is robust to temperature variations, as shown in Figure 5.12(a). In addition, the weather pattern was fairly consistent over a day and closely correlated with temperature, which implies that our method is also robust to weather. For Figure 5.12(b), the accuracy starts decreasing at the beginning of business hours around 7am and then starts increasing around noon time and then reaches the maximum value around 5pm. This may be due to indoor activity and outside traffic noise during business hours. On weekdays, indoor people activities are very frequent during 9am to 5pm. the traffic on the road between Rohm building and train track is particularly busy during the afternoon. There are several other factors may also affect the performance of the algorithm, such as wind speed, which can be potential future work for this project.

5.6 Conclusion

In this work we present a method to detect and characterize the train events through an information theoretic approach using a building's vibration responses. We represent the vibration propagation in the building structure as information exchanges in signals using directed information measure. Our event detection method first identifies time of train passing using wavelet-based features. Then, directed information estimators are used to to study the building's vibration characteristics under different train event conditions (southbound vs. northbound trains). We evaluated our method using an 11-story building located near a light rail line and achieved 93% true positive rate and 80% true negative rates for train event detection. The directions of passing trains were inferred with accuracy of 86.9%, which is a 2.5X reduction in error compared to conventional method.

The directed information based on causal analysis provides a new perspective in analyzing building dynamics, with which traditional physics-based analysis can be coupled, for further improvement. While this work focuses on the passing of trains, it can similarly be applied to other significant traffic events in the surrounding environment.

Chapter 6

Conclusions

In conclusion, this thesis introduce a set of methodologies which combine physical knowledge to acquire and learn high-fidelity information for urban infrastructure systems under constrained sensing capabilities. These methodologies aim to address 4 common constrained sensing capabilities in real-world urban infrastructure systems: 1) noisy data from complex physical systems, 2) lack of labeled data, 3) inefficient sensor deployment, and 4) lack of proper sensors.

To address the challenges of constrained sensing capabilities, I combine the knowledge about the physical processes inside infrastructure systems, between different infrastructure systems, between the infrastructure with the instrumented sensors, and between the infrastructure and environment. In different stages of urban infrastructure sensing and learning shown in Figure 1.1, the physical knowledge is utilized and embedded to best utilize the constrained sensing capabilities and improve the performance of high-fidelity information sensing and learning. When the data collected from complex physical systems is noisy, information-theoretic features are developed to model the wave propagation inside physical structures as information exchanges between vibrations to eliminate the influence of noise and improve the performance of structural damage diagnosis. To address the challenge of lack of labeled data, I introduce a physics-guided knowledge transfer framework to first transfer the knowledge learnt from other buildings to diagnose a building without any labeled data. To address the challenge of inefficient sensor deployment, I propose a new incentivizing algorithm to efficiently and flexibly actuate and deploy mobile sensor networks for various infrastructure monitoring tasks in the context of mobile crowd sensing systems. To address the challenge of indirect sensing, I also present an indirect sensing framework which combines the physical understanding of how traffic-induced vibrations propagate to the buildings to infer the traffic events from building data in the context of traffic event monitoring.

The development of advanced sensing technologies and big data algorithms enable the smart urban infrastructure systems. However, the gap between the physical knowledge and the data-driven

methods is obvious for smart urban infrastructure applications. It is undoubted that advanced sensing technologies provide us huge data, and the power of machine learning techniques enables the complex and highly nonlinear modeling of the collected data. But when it comes to the real-world physical infrastructure systems, many universal assumptions of these methods are violated under constrained sensing capabilities. It would be unreliable to entirely discard those classic physical principles and blindly apply these data-driven methods.

This thesis provides a new perspective to incorporate the physical knowledge into different sensing and learning stages to improve urban infrastructure monitoring. The future work of this thesis includes the extension of the proposed methodologies to different types of urban infrastructure systems and different tasks of urban infrastructure monitoring. For example, the information-theoretic building damage diagnosis and knowledge transfer framework to diagnose buildings without labeled data in earthquake scenarios can be extended to daily structural health monitoring and to bridge systems. The efficient sensor deployment scheme in urban crowd sensing systems can be extended to different types of sensor carriers and to multiple tasks of urban infrastructure monitoring. The indirect sensing framework can be extended to monitor different types of traffic events in the ambient environment.

References

[1] Sherin Abdelhamid, Hossam S Hassanein, and Glen Takahara. Reputation-aware, trajectory-based recruitment of smart vehicles for public sensing. *IEEE Transactions on Intelligent Transportation Systems*, 19(5):1387–1400, 2018. 4.2

[2] Osama Abdeljaber, Onur Avci, Serkan Kiranyaz, Moncef Gabbouj, and Daniel J Inman. Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks. *Journal of Sound and Vibration*, 388:154–170, 2017. 3.2

[3] M Adam and O Von Estorff. Reduction of train-induced building vibrations by using open and filled trenches. *Computers & structures*, 83(1):11–24, 2005. 5.3

[4] S Ahmad and TM Al-Hussaini. Simplified design for vibration screening by open and in-filled trenches. *Journal of Geotechnical Engineering*, 117(1):67–88, 1991. 5.3

[5] Vito Albino, Umberto Berardi, and Rosa Maria Dangelico. Smart cities: Definitions, dimensions, performance, and initiatives. *Journal of Urban Technology*, 22(1):3–21, 2015. 1.1

[6] Kashif Ali, Dina Al-Yaseen, Ali Ejaz, Tayyab Javed, and Hossam S Hassanein. Crowdits: Crowdsourcing in intelligent transportation systems. In *Wireless Communications and Networking Conference (WCNC), 2012 IEEE*, pages 3307–3311. IEEE, 2012. 1.1, 4.1

[7] Jude Allred, Ahmad Bilal Hasan, Saroch Panichsakul, William Pisano, Peter Gray, Jyh Huang, Richard Han, Dale Lawrence, and Kamran Mohseni. Sensorflock: an airborne wireless sensor network of micro-air vehicles. In *Proceedings of the 5th international conference on Embedded networked sensor systems*, pages 117–129. ACM, 2007. 4.1, 4.1, 4.5.2

[8] Georgios Amanatidis, Georgios Birmpas, and Evangelos Markakis. *Coverage, Matching, and Beyond: New Results on Budgeted Mechanism Design*. Springer Berlin Heidelberg, 2016. 4.2

[9] Stavros A Anagnostopoulos and Konstantinos V Spiliopoulos. An investigation of earthquake induced pounding between adjacent buildings. *Earthquake engineering & structural*

dynamics, 21(4):289–302, 1992. 3.3.1

[10] David R Anderson. *Information Theory and Entropy*. Springer, 2008. 5.4.3

[11] Roberto ARAYA. Earthquake accelerogram destructiveness potential factor. In *Proc. 8th World Conference on Earthquake Engineering, 1985. 7*, volume 11, pages 835–843, 1985. 3.3.1

[12] Massimo Livi Bacci. *A concise history of world population*. John Wiley & Sons, 2017. 1.1

[13] Mehdi Bahrekazemi. *Train-induced ground vibration and its prediction*. PhD thesis, Royal Institute of Technology, 2004. 5.3

[14] T Balendra, KH Chua, KW Lo, and SL Lee. Steady–state vibration of subway-soil-building system. *Journal of engineering mechanics*, 115(1):145–162, 1989. 5.3

[15] Elizabeth Bales, Nima Nikzad, Nichole Quick, Celal Ziftci, Kevin Patrick, and William Griswold. Citisense: Mobile air quality sensing for individuals and communities design and deployment of the citisense mobile air-quality system. In *Pervasive Computing Technologies for Healthcare (PervasiveHealth), 2012 6th International Conference on*, pages 155–158. IEEE, 2012. 4.1

[16] Roberto Battiti. Using mutual information for selecting features in supervised neural net learning. *IEEE Transactions on neural networks*, 5(4):537–550, 1994. 2.4.1

[17] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144, 2007. 3.4.1

[18] John Blitzer, Mark Dredze, and Fernando Pereira. Biographies, bollywood, boom-boxes and blenders: Domain adaptation for sentiment classification. In *Proceedings of the 45th annual meeting of the association of computational linguistics*, pages 440–447, 2007. 3.2

[19] Thomas Blumensath and Mike E Davies. Iterative thresholding for sparse approximations. *Journal of Fourier analysis and Applications*, 14(5-6):629–654, 2008. 4.4.1

[20] Jérôme Bolte, Shoham Sabach, and Marc Teboulle. Proximal alternating linearized minimization or nonconvex and nonsmooth problems. *Mathematical Programming*, 146(1-2): 459–494, 2014. 4.4.1

[21] Kurt M Bretthauer and Bala Shetty. The nonlinear knapsack problem–algorithms and applications. *European Journal of Operational Research*, 138(3):459–472, 2002. 4.4.1, 4.4.1

[22] JQ Bu, SS Law, and XQ Zhu. Innovative bridge condition assessment from dynamic response of a passing vehicle. *Journal of engineering mechanics*, 132(12):1372–1379, 2006.

5.2

[23] E Peter Carden and James MW Brownjohn. Arma modelled time-series classification for structural health monitoring of civil infrastructure. *Mechanical systems and signal processing*, 22(2):295–314, 2008. 2.2

[24] Ozan Cem Celik and Bruce R Ellingwood. Seismic fragilities for non-ductile reinforced concrete frames–role of aleatoric and epistemic uncertainties. *Structural Safety*, 32(1):1–12, 2010. 2.1

[25] Fu-Kuo Chang. *Structural health monitoring 2000*. CRC Press, 1999. 5.4.2

[26] Kai-Wei Chang, Cho-Jui Hsieh, and Chih-Jen Lin. Coordinate descent method for large-scale l2-loss linear support vector machines. *Journal of Machine Learning Research*, 9 (Jul):1369–1398, 2008. 4.4.3

[27] Siyu Chen, Yong Li, Wenyu Ren, Depeng Jin, and Pan Hui. Location prediction for large scale urban vehicular mobility. In *Wireless Communications and Mobile Computing Conference (IWCMC), 2013 9th International*, pages 1733–1737. IEEE, 2013. 4.3.1, 4.3.4, 4.5.1

[28] Xinlei Chen, Aveek Purohit, Carlos Ruiz Dominguez, Stefano Carpin, and Pei Zhang. Drunkwalk: Collaborative and adaptive planning for navigation of micro-aerial sensor swarms. In *Proceedings of the 13th ACM Conference on Embedded Networked Sensor Systems*, pages 295–308. ACM, 2015. 4.1

[29] Xinlei Chen, Xiangxiang Xu, Xinyu Liu, Hae Young Noh, Lin Zhang, and Pei Zhang. Hap: Fine-grained dynamic air pollution map reconstruction by hybrid adaptive particle filter. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 336–337. ACM, 2016. 4.1, 4.2, 5.1

[30] Xinlei Chen, Aveek Purohit, Shijia Pan, Carlos Ruiz, Jun Han, Zheng Sun, Frank Mokaya, Patric Tague, and Pei Zhang. Design experiences in minimalistic flying sensor node platform through sensorfly. *ACM Transactions on Sensor Networks (TOSN)*, 13(4):33, 2017. 4.1

[31] Xinlei Chen, Xiangxiang Xu, Xinyu Liu, Shijia Pan, Jiayou He, Hae Young Noh, Lin Zhang, and Pei Zhang. Pga: Physics guided and adaptive approach for mobile fine-grained air pollution estimation. In *Proceedings of the 2018 ACM International Joint Conference and 2018 International Symposium on Pervasive and Ubiquitous Computing and Wearable Computers*, pages 1321–1330. ACM, 2018. 4.1

[32] Xinlei Chen, Susu Xu, Haohao Fu, Carlee Joe-Wong, Lin Zhang, Hae Young Noh, and Pei Zhang. Asc: Actuation system for city-wide crowdsensing with ride-sharing vehicular plat-

form. In *14th International Science of Smart City Operations and Platforms Engineering*. IEEE, 2019. 4.2

[33] N Cheraghi, GP Zou, and F Taheri. Piezoelectric-based degradation assessment of a pipe using fourier and wavelet analyses. *Computer-Aided Civil and Infrastructure Engineering*, 20(5):369–382, 2005. 2.2

[34] KH Chua, KW Lo, and T Balendra. Building response due to subway train traffic. *Journal of Geotechnical Engineering*, 121(11):747–754, 1995. 5.3

[35] Ronald R Coifman, Yves Meyer, and Victor Wickerhauser. Wavelet analysis and signal processing. In *In Wavelets and their Applications*, pages 153–178. Citeseer, 1992. 5.4.2

[36] Sinem Coleri, Sing Yiu Cheung, and Pravin Varaiya. Sensor networks for monitoring traffic. In *Allerton conference on communication, control and computing*, pages 32–40, 2004. 1.1

[37] Pierre Comon. Independent component analysis, a new concept? *Signal processing*, 36(3): 287–314, 1994. 2.1, 2.2, 2.4.1, 5.4.3

[38] Corinna Cortes and Vladimir Vapnik. Support vector machine. *Machine learning*, 20(3): 273–297, 1995. 5.4.2

[39] Edoardo Cosenza and Gaetano Manfredi. Damage indices and damage measures. *Progress in Structural Engineering and Materials*, 2(1):50–59, 2000. 3.3.1

[40] Applied Technology Council. *Improvement of nonlinear static seismic analysis procedures*. FEMA Region II, 2005. 2.5.3, 3.5.1

[41] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000. 2.5.3, 5.4.3

[42] Graham Currie. Gap analysis of public transport needs: measuring spatial distribution of public transport needs and identifying gaps in the quality of public transport provision. *Transportation Research Record: Journal of the Transportation Research Board*, 1(1895): 137–146, 2004. 1.1, 4.1

[43] Wenyuan Dai, Qiang Yang, Gui-Rong Xue, and Yong Yu. Self-taught clustering. In *Proceedings of the 25th international conference on Machine learning*, pages 200–207. ACM, 2008. 3.2

[44] Hal Daume III and Daniel Marcu. Domain adaptation for statistical classifiers. *Journal of artificial Intelligence research*, 26:101–126, 2006. 3.2

[45] Srinivas Devarakonda, Parveen Sevusu, Hongzhang Liu, Ruilin Liu, Liviu Iftode, and Badri Nath. Real-time air quality monitoring through mobile sensing in metropolitan areas. In *Pro-*

ceedings of the 2nd ACM SIGKDD international workshop on urban computing, page 15. ACM, 2013. 4.2, 4.5.2

[46] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018. 3.4.3

[47] Inderjit S Dhillon, Subramanyam Mallela, and Dharmendra S Modha. Information-theoretic co-clustering. In *Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 89–98. ACM, 2003. 2.2, 2.4.1, 5.4.3

[48] Scott W Doebling, Charles R Farrar, Michael B Prime, and Daniel W Shevitz. Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: a literature review. Technical report, Los Alamos National Lab., NM (United States), 1996. 1.1, 2.1, 5.1

[49] Jeff Donahue, Yangqing Jia, Oriol Vinyals, Judy Hoffman, Ning Zhang, Eric Tzeng, and Trevor Darrell. Decaf: A deep convolutional activation feature for generic visual recognition. In *International conference on machine learning*, pages 647–655, 2014. 3.2

[50] Zhuojun Duan, Mingyuan Yan, Zhipeng Cai, Xiaoming Wang, Meng Han, and Yingshu Li. Truthful incentive mechanisms for social cost minimization in mobile crowdsourcing systems. *Sensors*, 16(4):481, 2016. 4.1

[51] Brace Ellingwood and Andrew Tallin. Structural serviceability: floor vibrations. *Journal of structural engineering*, 110(2):401–418, 1984. 5.1

[52] Jeffrey Erochko, Constantin Christopoulos, Robert Tremblay, and Hyunhoon Choi. Residual drift response of smrfs and brb frames in steel buildings designed according to asce 7-05. *Journal of Structural Engineering*, 137(5):589–599, 2010. 2.1

[53] Estados Unidos Federal Emergency Management Agency. FEMA. *Prestandard and commentary for the seismic rehabilitation of buildings*. FEMA, 2000. 2.5.3

[54] Jeffrey Fulmer. What in the world is infrastructure. *PEI Infrastructure investor*, 1(4):30–32, 2009. 1.1

[55] Chuang Gan, Tianbao Yang, and Boqing Gong. Learning attributes equals multi-source domain generalization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 87–97, 2016. 3.2

[56] Yaroslav Ganin and Victor Lempitsky. Unsupervised domain adaptation by backpropagation. *arXiv preprint arXiv:1409.7495*, 2014. 3.2

[57] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle,

François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030, 2016. 3.2, 3.4.4, 3.5.2

[58] Raghu K Ganti, Fan Ye, and Hui Lei. Mobile crowdsensing: current state and future challenges. *IEEE Communications Magazine*, 49(11), 2011. 4.1

[59] C Gentile and A Saisi. Ambient vibration testing of historic masonry towers for structural identification and damage assessment. *Construction and Building Materials*, 21(6):1311–1321, 2007. 5.1

[60] Muhammad Ghifary, W Bastiaan Kleijn, Mengjie Zhang, David Balduzzi, and Wen Li. Deep reconstruction-classification networks for unsupervised domain adaptation. In *European Conference on Computer Vision*, pages 597–613. Springer, 2016. 3.1

[61] Ahmed Ghobarah. Performance-based design in earthquake engineering: state of development. *Engineering structures*, 23(8):878–884, 2001. 3.1

[62] Xavier Glorot, Antoine Bordes, and Yoshua Bengio. Domain adaptation for large-scale sentiment classification: A deep learning approach. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 513–520, 2011. 3.2

[63] Boqing Gong, Yuan Shi, Fei Sha, and Kristen Grauman. Geodesic flow kernel for unsupervised domain adaptation. In *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 2066–2073. IEEE, 2012. 3.2

[64] Pinghua Gong, Changshui Zhang, Zhaosong Lu, Jianhua Huang, and Jieping Ye. A general iterative shrinkage and thresholding algorithm for non-convex regularized optimization problems. In *International Conference on Machine Learning*, pages 37–45, 2013. 4.4.1

[65] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 3.4.4

[66] James-Alexandre Goulet, Clotaire Michel, and A Der Kiureghian. Data-driven post-earthquake rapid structural safety assessment. *Earthquake Engineering & Structural Dynamics*, 44(4):549–562, 2015. 3.1, 3.2

[67] Jayavardhana Gubbi, Rajkumar Buyya, Slaven Marusic, and Marimuthu Palaniswami. Internet of things (iot): A vision, architectural elements, and future directions. *Future Generation Computer Systems*, 29(7):1645–1660, 2013. 5.1

[68] Guoqing Gui, Hong Pan, Zhibin Lin, Yonghua Li, and Zhijun Yuan. Data-driven support vector machine with optimization techniques for structural health monitoring and damage

detection. *KSCE Journal of Civil Engineering*, 21(2):523–534, 2017. 3.2

[69] Philip Haeusser, Thomas Frerix, Alexander Mordvintsev, and Daniel Cremers. Associative domain adaptation. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2765–2773, 2017. 3.1

[70] Lars Hall. Simulations and analyses of train-induced ground vibrations in finite element models. *Soil Dynamics and Earthquake Engineering*, 23(5):403–413, 2003. 5.3

[71] Jian-Gang Han, Wei-Xin Ren, and Zeng-Shou Sun. Wavelet packet based damage identification of beam structures. *International Journal of Solids and Structures*, 42(26):6610–6627, 2005. 2.2

[72] Hong Hao, TC Ang, and Jay Shen. Building vibration to traffic-induced ground motion. *Building and Environment*, 36(3):321–336, 2001. 5.3

[73] Makarand Hastak and Earl J Baim. Risk factors affecting management and maintenance cost of urban infrastructure. *Journal of Infrastructure Systems*, 7(2):67–76, 2001. 1.1

[74] Zongjian He, Jiannong Cao, and Xuefeng Liu. High quality participant recruitment in vehicle-based crowdsourcing using predictable mobility. In *Computer Communications (INFOCOM), 2015 IEEE Conference on*, pages 2542–2550. IEEE, 2015. 4.1, 4.2, 4.5.1

[75] Adriana Hera and Zhikun Hou. Application of wavelet approach for asce structural health monitoring benchmark studies. *Journal of Engineering Mechanics*, 130(1):96–104, 2004. 5.4.2

[76] Judy Hoffman, Brian Kulis, Trevor Darrell, and Kate Saenko. Discovering latent domains for multisource domain adaptation. In *European Conference on Computer Vision*, pages 702–715. Springer, 2012. 3.2

[77] Zhikun Hou, Adriana Hera, and Abhijeet Shinde. Wavelet-based structural health monitoring of earthquake excited structures. *Computer-Aided Civil and Infrastructure Engineering*, 21(4):268–279, 2006. 2.2

[78] Weiming Hu, Xuejuan Xiao, Zhouyu Fu, Dan Xie, Tieniu Tan, and Steve Maybank. A system for learning statistical motion patterns. *IEEE transactions on pattern analysis and machine intelligence*, 28(9):1450–1464, 2006. 4.5.3

[79] Norden E Huang, Zheng Shen, Steven R Long, Manli C Wu, Hsing H Shih, Quanan Zheng, Nai-Chyuan Yen, Chi Chao Tung, and Henry H Liu. The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis. In *Proceedings of the Royal Society of London A: mathematical, physical and engineering sciences*, volume 454, pages 903–995. The Royal Society, 1998. 2.2

[80] Seong-Hoon Hwang and Dimitrios G Lignos. Earthquake-induced loss assessment of steel frame buildings with special moment frames designed in highly seismic regions. *Earthquake Engineering & Structural Dynamics*, 46(13):2141–2162, 2017. 2.6.1, 3.2, 3.5.1

[81] Seong-Hoon Hwang and Dimitrios G Lignos. Assessment of structural damage detection methods for steel structures using full-scale experimental data and nonlinear analysis. *Bulletin of Earthquake Engineering*, 16(7):2971–2999, 2018. 2.6.1

[82] Seong-Hoon Hwang and Dimitrios G Lignos. Nonmodel-based framework for rapid seismic risk and loss assessment of instrumented steel buildings. *Engineering Structures*, 156:417–432, 2018. 3.1, 3.2, 3.5.1

[83] Toshihide Ibaraki. Approximate algorithms for the multiple-choice continuous knapsack problems. *Journal of the Operations Research Society of Japan*, 23(1):28–63, 1980. 4.4.1

[84] Toshihide Ibaraki, Toshiharu Hasegawa, Katsumi Teranaka, and Jiro Iwase. The multiple-choice knapsack problem. *Journal of the Operations Research Society of Japan*, 21(1): 59–95, 1978. 4.4.1

[85] Yoshiki Ikeda. Verification of system identification utilizing shaking table tests of a full-scale 4-story steel building. *Earthquake Engineering & Structural Dynamics*, 45(4):543–562, 2016. 2.1

[86] Luis G Jaimes, Idalides Vergara-Laurens, and Miguel A Labrador. A location-based incentive mechanism for participatory sensing systems with budget constraints. In *Pervasive Computing and Communications (PerCom), 2012 IEEE International Conference on*, pages 103–108. IEEE, 2012. 4.2

[87] Luis G. Jaimes, Idalides J. Vergara-Laurens, and Andrew Raij. A survey of incentive techniques for mobile crowd sensing. *IEEE Internet of Things Journal*, 2(5):370–380, 2015. 4.2

[88] Abhinav Jauhri, Brian Foo, Jerome Berclaz, Chih Chi Hu, Radek Grzeszczuk, Vasu Parameswaran, and John Paul Shen. Space-time graph modeling of ride requests based on real-world data. *arXiv preprint arXiv:1701.06635*, 2017. 4.3.4, 4.5.1

[89] Edwin T Jaynes. Information theory and statistical mechanics. *Physical review*, 106(4):620, 1957. 2.2, 2.4.1, 5.4.3

[90] Elaina Jennings, John W van de Lindt, Ershad Ziaei, Gary Mochizuki, Weichiang Pang, and Xiaoyun Shao. Retrofit of a soft-story woodframe building using sma devices with full-scale hybrid test verification. *Engineering Structures*, 80:469–485, 2014. 1.2, 1.2.2, 3.1

[91] Shenggong Ji, Yu Zheng, and Tianrui Li. Urban sensing based on human mobility. In

Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing, pages 1040–1051. ACM, 2016. 4.2, 4.3.2, 4.5.2, 4.5.2

[92] Xiaodong Ji, Gregory L Fenves, Kouichi Kajiwara, and Masayoshi Nakashima. Seismic damage detection of a full-scale shaking table test structure. *Journal of Structural Engineering*, 137(1):14–21, 2010. 2.1, 3.2

[93] Zhenhua Jia, Musaab Alaziz, Xiang Chi, Richard E Howard, Yanyong Zhang, Pei Zhang, Wade Trappe, Anand Sivasubramaniam, and Ning An. Hb-phone: a bed-mounted geophone-based heartbeat monitoring system. In *Information Processing in Sensor Networks (IPSN), 2016 15th ACM/IEEE International Conference on*, pages 1–12. IEEE, 2016. 5.1

[94] Jiantao Jiao, Haim H Permuter, Lei Zhao, Young-Han Kim, and Tsachy Weissman. Universal estimation of directed information. *IEEE Transactions on Information Theory*, 59(10): 6220–6242, 2013. 2.2, 2.4.1, 2.4.1, 2.5.2, 5.4.3

[95] Jiantao Jiao, Kartik Venkat, Yanjun Han, and Tsachy Weissman. Minimax estimation of functionals of discrete distributions. *IEEE Transactions on Information Theory*, 61(5): 2835–2885, 2015. 2.2, 2.4.1, 2.4.1, 5.4.3

[96] Shen-Haw Ju. Finite element investigation of traffic induced vibrations. *Journal of Sound and Vibration*, 321(3):837–853, 2009. 5.3

[97] Motoki Kazama and Toshihiro Noda. Damage statistics (summary of the 2011 off the pacific coast of tohoku earthquake damage). *Soils and Foundations*, 52(5):780–792, 2012. 2.1

[98] Hans Kellerer, Ulrich Pferschy, and David Pisinger. Introduction to np-completeness of knapsack problems. In *Knapsack problems*, pages 483–493. Springer, 2004. 4.4.1, 4.4.1

[99] Junaid Ahmed Khan, Yacine Ghamri-Doudane, and Dmitri Botvich. Autonomous identification and optimal selection of popular smart vehicles for urban sensingan information-centric approach. *IEEE Transactions on Vehicular Technology*, 65(12):9529–9541, 2016. 4.2

[100] Kavi K Khedo, Rajiv Perseedoss, Avinash Mungur, et al. A wireless sensor network air pollution monitoring system. *arXiv preprint arXiv:1005.1737*, 2010. 4.1, 4.1, 4.2

[101] Sukun Kim, Shamim Pakzad, David Culler, James Demmel, Gregory Fenves, Steven Glaser, and Martin Turon. Health monitoring of civil infrastructures using wireless sensor networks. In *2007 6th International Symposium on Information Processing in Sensor Networks*, pages 254–263. IEEE, 2007. 5.4.3

[102] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 3.5.3, 3.5.4

[103] Charles Kircher, Gregory Deierlein, John Hooper, Helmut Krawinkler, Steve Mahin, Benson Shing, and John Wallace. Evaluation of the fema p-695 methodology for quantification of building seismic performance factors— nist. Technical report, NIST, 2010. 2.5.3, 3.5.1

[104] Philipp Koehn and Josh Schroeder. Experiments in domain adaptation for statistical machine translation. In *Proceedings of the second workshop on statistical machine translation*, pages 224–227, 2007. 3.2

[105] Vassilis Kostakos, Timo Ojala, and Tomi Juntunen. Traffic in the smart city: Exploring city-wide sensing for traffic control center augmentation. *Internet Computing, IEEE*, 17(6): 22–29, 2013. 5.1

[106] Georges Kouroussis, David P Connolly, and Olivier Verlinden. Railway-induced ground vibrations–a review of vehicle effects. *International Journal of Rail Transportation*, 2(2): 69–110, 2014. 5.3

[107] GERHARD KRAMER. *Directed Information for Channels with Feedback*. PhD thesis, Citeseer, 1998. 2.3

[108] Alexander Kraskov, Harald Stögbauer, and Peter Grassberger. Estimating mutual information. *Physical review E*, 69(6):066138, 2004. 2.4.1

[109] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*, pages 1097–1105, 2012. 3.4.3

[110] Solomon Kullback and Richard A Leibler. On information and sufficiency. *The annals of mathematical statistics*, 22(1):79–86, 1951. 4.3.1, 4.3.2

[111] Elizabeth L Labuz, Minwoo Chang, and Shamim N Pakzad. Local damage detection in beam-column connections using a dense sensor network. In *Structures Congress 2010*, pages 3143–3154, 2010. 2.1

[112] Mike Lam, Mostafa Mirshekari, Shijia Pan, Pei Zhang, and Hae Young Noh. Robust occupant detection through step-induced floor vibration by incorporating structural characteristics. In *Dynamics of Coupled Structures, Volume 4*, pages 357–367. Springer, 2016. 5.5.3

[113] Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436, 2015. 3.4.3

[114] George Lederman, Siheng Chen, James Garrett, Jelena Kovačević, Hae Young Noh, and Jacobo Bielak. Track-monitoring from the dynamic response of an operational train. *Mechanical Systems and Signal Processing*, 87:1–16, 2017. 5.2, 5.3

[115] George Lederman, Siheng Chen, James H Garrett, Jelena Kovačević, Hae Young Noh, and

Jacobo Bielak. A data fusion approach for track monitoring from multiple in-service trains. *Mechanical Systems and Signal Processing*, 95:363–379, 2017. 5.2

[116] George Lederman, Siheng Chen, James H Garrett, Jelena Kovačević, Hae Young Noh, and Jacobo Bielak. Track monitoring from the dynamic response of a passing train: A sparse approach. *Mechanical Systems and Signal Processing*, 90:141–153, 2017. 5.2, 5.3

[117] Jungwhee Lee and Sungkon Kim. Structural damage detection in the frequency domain using neural networks. *Journal of Intelligent Material Systems and Structures*, 18(8):785–792, 2007. 2.2

[118] Juong-Sik Lee and Baik Hoh. Sell your experiences: a market mechanism based incentive for participatory sensing. In *Pervasive Computing and Communications (PerCom), 2010 IEEE International Conference on*, pages 60–68. IEEE, 2010. 4.2

[119] Uichin Lee, Biao Zhou, Mario Gerla, Eugenio Magistretti, Paolo Bellavista, and Antonio Corradi. Mobeyes: smart mobs for urban monitoring with a vehicular sensor network. *IEEE Wireless Communications*, 13(5), 2006. 4.1, 4.1

[120] Stéphanie Lefèvre, Dizan Vasquez, and Christian Laugier. A survey on motion prediction and risk assessment for intelligent vehicles. *Robomech Journal*, 1(1):1, 2014. 4.3.1

[121] QS Li, JR Wu, SG Liang, YQ Xiao, and CK Wong. Full-scale measurements and numerical evaluation of wind-induced vibration of a 63-story reinforced concrete tall building. *Engineering structures*, 26(12):1779–1794, 2004. 5.1

[122] Xiaohua Li, Masahiro Kurata, and Akiko Suzuki. Decoupling algorithm for evaluating multiple beam damages in steel moment-resisting frames. *Earthquake Engineering & Structural Dynamics*, 46(7):1045–1064, 2017. 2.1

[123] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R news*, 2 (3):18–22, 2002. 5.4.2

[124] DG Lignos, H Krawinkler, and AS Whittaker. Shaking table collapse tests of two scale models of a 4-story moment resisting steel frame. In *14th World Conference on Earthquake Engineering*, pages 12–17, 2008. 3.2, 3.5.1

[125] CC Lin, Jer-Fu Wang, and BL Chen. Train-induced vibration control of high-speed railway bridges equipped with multiple tuned mass dampers. *Journal of Bridge Engineering*, 10(4): 398–414, 2005. 5.3

[126] CW Lin and YB Yang. Use of a passing vehicle to scan the fundamental bridge frequencies: An experimental verification. *Engineering Structures*, 27(13):1865–1878, 2005. 5.2

[127] Jen-Hao Liu, Yu-Fan Chen, Tzu-Shiang Lin, Da-Wei Lai, Tzai-Hung Wen, Chih-Hong Sun,

Jehn-Yih Juang, and Joe-Air Jiang. Developed urban air quality monitoring system based on wireless sensor networks. In *Sensing technology (icst), 2011 fifth international conference on*, pages 549–554. IEEE, 2011. 4.1

[128] Jianbo Liu, Xinwei Wang, Shenfang Yuan, and Gang Li. On hilbert-huang transform approach for structural health monitoring. *Journal of Intelligent Material Systems and Structures*, 17(8-9):721–728, 2006. 2.2

[129] Yazhi Liu, Jianwei Niu, and Xiting Liu. Comprehensive tempo-spatial data collection in crowd sensing using a heterogeneous sensing vehicle selection method. *Personal and Ubiquitous Computing*, 20(3):397–411, 2016. 4.2

[130] Yu Liu, Chaogui Kang, Song Gao, Yu Xiao, and Yuan Tian. Understanding intra-urban trip patterns from taxi trajectory data. *Journal of geographical systems*, 14(4):463–483, 2012. 4.1

[131] Mingsheng Long, Jianmin Wang, Guiguang Ding, Jiaguang Sun, and Philip S Yu. Transfer feature learning with joint distribution adaptation. In *Proceedings of the IEEE international conference on computer vision*, pages 2200–2207, 2013. 3.2

[132] Mingsheng Long, Jianmin Wang, Jiaguang Sun, and S Yu Philip. Domain invariant transfer kernel learning. *IEEE Transactions on Knowledge and Data Engineering*, 27(6):1519–1532, 2014. 3.2

[133] Jerome P Lynch and Kenneth J Loh. A summary review of wireless sensors and sensor networks for structural health monitoring. *Shock and Vibration Digest*, 38(2):91–130, 2006. 1.1, 5.1, 5.4.3

[134] Jerome P Lynch, Charles R Farrar, and Jennifer E Michaels. Structural health monitoring: technological advances to practical implementations [scanning the issue]. *Proceedings of the IEEE*, 104(8):1508–1512, 2016. 2.1

[135] Jerome Peter Lynch. An overview of wireless structural health monitoring for civil structures. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, 365(1851):345–372, 2006. 2.1

[136] Huadong Ma, Dong Zhao, and Peiyan Yuan. Opportunities in mobile crowd sensing. *IEEE Communications Magazine*, 52(8):29–35, 2014. 1.2, 4.1, 4.2

[137] Wei Ma and Zhen Sean Qian. Estimating multi-year 24/7 origin-destination demand using high-granular multi-source traffic data. *Transportation Research Part C: Emerging Technologies*, 96:96–121, 2018. 4.3.1

[138] Yajie Ma, Mark Richards, Moustafa Ghanem, Yike Guo, and John Hassard. Air pollution

monitoring and mining based on sensor grid in london. *Sensors*, 8(6):3601–3623, 2008. 4.2

[139] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 3.3

[140] Abdollah Malekjafarian, Eugene J OBrien, and Fatemeh Golpayegani. Indirect monitoring of critical transport infrastructure: Data analytics and signal processing. In *Data Analytics for Smart Cities*, pages 157–176. Auerbach Publications, 2018. 5.2

[141] Hans Marko. The bidirectional communication theory–a generalization of information theory. *Communications, IEEE Transactions on*, 21(12):1345–1351, 1973. 5.1

[142] J Massey. Causality, feedback and directed information. In *Proc. Int. Symp. Inf. Theory Applic.(ISITA-90)*, pages 303–305. Citeseer, 1990. 2.2, 2.3, 2.4.1, 2.4.1, 5.4.3, 5.4.3

[143] Steven G Mattson and Sudhakar M Pandit. Statistical moments of autoregressive model residuals for damage localisation. *Mechanical Systems and Signal Processing*, 20(3):627–645, 2006. 2.6

[144] David McCallen, Floriana Petrone, Jason Coates, and Nicholas Repanich. A laser-based optical sensor for broad-band measurements of building earthquake drift. *Earthquake spectra*, 33(4):1573–1598, 2017. 1.2, 1.2.2, 3.1

[145] Patrick J McGetrick, A Gonzlez, and Eugene J OBrien. Theoretical investigation of the use of a moving vehicle to identify bridge dynamic parameters. *Insight-Non-Destructive Testing and Condition Monitoring*, 51(8):433–438, 2009. 5.2

[146] Francis Thomas Mckenna. *Object-oriented finite element programming: Frameworks for analysis, algorithms and parallel computing*. PhD thesis, University of California, Berkeley, 1999. 3.5.1

[147] Ricardo Antonio Medina and Helmut Krawinkler. Seismic demands for nondeteriorating frame structures and their dependence on ground motions. Technical report, Department of CEE, Stanford University, 2003. 2.6.1

[148] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*, 2010. 3.4.3

[149] Mostafa Mirshekari, Shijia Pan, Adeola Bannis, Yan Pui Mike Lam, Pei Zhang, and Hae Young Noh. Step-level person localization through sparse sensing of structural vibration. In *proceedings of the 14th international conference on information processing in sensor networks*, pages 376–377. ACM, 2015. 5.5.3

[150] Mostafa Mirshekari, Shijia Pan, Pei Zhang, and Hae Young Noh. Characterizing wave

propagation to improve indoor step-level person localization using floor vibration. In *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, pages 980305–980305. International Society for Optics and Photonics, 2016. 5.1

[151] Mostafa Mirshekari, Pei Zhang, and Hae Young Noh. Non-intrusive occupant localization using floor vibrations in dispersive structure: Poster abstract. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 378–379. ACM, 2016. 5.1

[152] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013. 3.4.3

[153] Frank Mokaya, Brian Nguyen, Cynthia Kuo, Quinn Jacobson, Anthony Rowe, and Pei Zhang. Mars: a muscle activity recognition system enabling self-configuring musculoskeletal sensor networks. In *Information Processing in Sensor Networks (IPSN), 2013 ACM/IEEE International Conference on*, pages 191–202. IEEE, 2013. 5.3

[154] Frank Mokaya, Roland Lucas, Hae Young Noh, and Pei Zhang. Myovibe: Vibration based wearable muscle activation detection in high mobility exercises. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 27–38. ACM, 2015. 5.3

[155] Frank Mokaya, Roland Lucas, Hae Young Noh, and Pei Zhang. Burnout: a wearable system for unobtrusive skeletal muscle fatigue estimation. In *Information Processing in Sensor Networks (IPSN), 2016 15th ACM/IEEE International Conference on*, pages 1–12. IEEE, 2016. 5.3

[156] Brendan Tran Morris and Mohan M Trivedi. Trajectory learning for activity understanding: Unsupervised, multilevel, and long-term adaptive approach. *IEEE transactions on pattern analysis and machine intelligence*, 33(11):2287–2301, 2011. 4.3.1

[157] Masato Motosaka and Kazuya Mitsuji. Building damage during the 2011 off the pacific coast of tohoku earthquake. *Soils and Foundations*, 52(5):929–944, 2012. 2.1

[158] Tamer Nadeem, Sasan Dashtinezhad, Chunyuan Liao, and Liviu Iftode. Trafficview: A scalable traffic monitoring system. In *Mobile Data Management, 2004. Proceedings. 2004 IEEE International Conference on*, pages 13–26. IEEE, 2004. 5.1

[159] K Krishnan Nair, Anne S Kiremidjian, and Kincho H Law. Time series-based damage detection and localization algorithm with application to the asce benchmark structure. *Journal of Sound and Vibration*, 291(1-2):349–368, 2006. 2.2

[160] Sang-Hyeok Nam, Ha-Won Song, Keun-Joo Byun, and Koichi Maekawa. Seismic analysis of underground reinforced concrete structures considering elasto-plastic interface element with thickness. *Engineering Structures*, 28(8):1122–1131, 2006. 3.3.1

[161] Yu Nesterov. Efficiency of coordinate descent methods on huge-scale optimization problems. *SIAM Journal on Optimization*, 22(2):341–362, 2012. 4.4.3

[162] Zhaolong Ning, Feng Xia, Noor Ullah, Xiangjie Kong, and Xiping Hu. Vehicular social networks: Enabling smart mobility. *IEEE Communications Magazine*, 55(5):16–55, 2017. 4.1

[163] H Noh and AS Kiremidjian. On the use of wavelet coefficient energy for structural damage diagnosis. In *Proc. the 10th International Conference on Structural Safety and Reliability, Osaka, Japan*, pages 3942–3947, 2009. 5.4.2

[164] Hae Young Noh, Dimitrios G Lignos, K Krishnan Nair, and Anne S Kiremidjian. Development of fragility functions as a damage classification/prediction method for steel moment-resisting frames using a wavelet-based damage sensitive feature. *Earthquake Engineering & Structural Dynamics*, 41(4):681–696, 2012. 1.1, 2.1, 3.1, 3.2, 5.4.2

[165] HY Noh, DG Lignos, KK Nair, and AS Kiremidjian. Application of wavelet based damage sensitive features for structural damage diagnosis. In *Proceedings of the 7th International Workshop on Structural Health Monitoring, Stanford, CA, USA*, pages 2011–2018, 2009. 5.4.2

[166] David Sousa Nunes, Pei Zhang, and Jorge Sá Silva. A survey on human-in-the-loop applications towards an internet of all. *IEEE Communications Surveys & Tutorials*, 17(2):944–965, 2015. 5.1

[167] Robert Ighodaro Ogie. Adopting incentive mechanisms for large-scale participation in mobile crowdsensing: from literature review to a conceptual framework. *Human-centric Computing and Information Sciences*, 6(1):24, 2016. 4.2

[168] Shamim N Pakzad and Gregory L Fenves. Statistical analysis of vibration modes of a suspension bridge using spatially dense wireless sensor network. *Journal of structural engineering*, 135(7):863–872, 2009. 3.1, 3.2

[169] Shijia Pan, Amelie Bonde, Jie Jing, Lin Zhang, Pei Zhang, and Hae Young Noh. Boes: building occupancy estimation system using sparse ambient vibration monitoring. In *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, pages 90611O–90611O. International Society for Optics and Photonics, 2014. 5.1

[170] Shijia Pan, Mostafa Mirshekari, Hae Young Noh, and Pei Zhang. Structural sensing system

with networked dynamic sensing configuration. In *Proceedings of the 14th International Conference on Information Processing in Sensor Networks*, pages 344–345. ACM, 2015. 5.1

[171] Shijia Pan, Ningning Wang, Yuqiu Qian, Irem Velibeyoglu, Hae Young Noh, and Pei Zhang. Indoor person identification through footstep induced structural vibration. In *Proceedings of the 16th International Workshop on Mobile Computing Systems and Applications*, pages 81–86. ACM, 2015. 5.5.3

[172] Shijia Pan, Kent Lyons, Mostafa Mirshekari, Hae Young Noh, and Pei Zhang. Multiple pedestrian tracking through ambient structural vibration sensing: Poster abstract. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 366–367. ACM, 2016. 5.5.3

[173] Shijia Pan, Mostafa Mirshekari, Pei Zhang, and Hae Young Noh. Occupant traffic estimation through structural vibration sensing. In *SPIE Smart Structures and Materials+ Nondestructive Evaluation and Health Monitoring*, pages 980306–980306. International Society for Optics and Photonics, 2016. 5.1

[174] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on knowledge and data engineering*, 22(10):1345–1359, 2010. 3.1, 3.1, 3.2

[175] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2010. 3.2, 3.5.2

[176] Sinno Jialin Pan, Ivor W Tsang, James T Kwok, and Qiang Yang. Domain adaptation via transfer component analysis. *IEEE Transactions on Neural Networks*, 22(2):199–210, 2011. 3.1, 3.2

[177] WeiChiang Pang, DV Rosowsky, JW van de Lindt, and Shiling Pei. Simplified direct displacement design of six-story neeswood capstone building and pre-test seismic performance assessment. *NEESWood Report NW-05*, 2009. 3.1

[178] Young-Ji Park, Alfredo H-S Ang, and Yi Kwei Wen. Seismic damage analysis of reinforced concrete buildings. *Journal of Structural Engineering*, 111(4):740–757, 1985. 3.3.1

[179] Chathurdara Sri Nadith Pathirage, Jun Li, Ling Li, Hong Hao, Wanquan Liu, and Pinghe Ni. Structural damage identification based on autoencoder neural networks and deep learning. *Engineering Structures*, 172:13–28, 2018. 3.2

[180] Eric Paulos, Richard J Honicky, and Elizabeth Goodman. Sensing atmosphere. *Human-Computer Interaction Institute*, page 203, 2007. 4.1, 4.1, 4.5.2

[181] Haim H Permuter, Young-Han Kim, and Tsachy Weissman. On directed information and gambling. In *2008 IEEE International Symposium on Information Theory*, pages 1403–1407. IEEE, 2008. 5.1

[182] Haim H Permuter, Young-Han Kim, and Tsachy Weissman. Interpretations of directed information in portfolio theory, data compression, and hypothesis testing. *IEEE Transactions on Information Theory*, 57(6):3248–3259, 2011. 5.1

[183] Vitaly Petrov, Andrey Samuylov, Vyacheslav Begishev, Dmitri Moltchanov, Sergey Andreev, Konstantin Samouylov, and Yevgeni Koucheryavy. Vehicle-based relay assistance for opportunistic crowdsensing over narrowband iot (nb-iot). *IEEE Internet of Things Journal*, 2017. 4.1

[184] Xidong Pi and Zhen Sean Qian. A stochastic optimal control approach for real-time traffic routing considering demand uncertainties and travelers choice heterogeneity. *Transportation Research Part B: Methodological*, 104:710–732, 2017. 4.3.1

[185] Ricardo Piedrahita, Yun Xiang, Nick Masson, John Ortega, Ashley Collier, Yifei Jiang, Kun Li, Robert P Dick, Qin Lv, Micahel Hannigan, et al. The next generation of low-cost personal air quality sensors for quantitative exposure monitoring. *Atmospheric Measurement Techniques*, 7(10):3325–3336, 2014. 4.1

[186] Darryll Pines and Liming Salvino. Structural health monitoring using empirical mode decomposition and the hilbert phase. *Journal of sound and vibration*, 294(1-2):97–124, 2006. 2.2

[187] Darryll J Pines and Philip A Lovell. Conceptual framework of a remote wireless health monitoring system for large civil structures. *Smart materials and Structures*, 7(5):627, 1998. 5.4.3

[188] FEMA Prestandard. commentary for the seismic rehabilitation of buildings (fema356). *Washington, DC: Federal Emergency Management Agency*, 7, 2000. 2.5.3, 3.5.1

[189] Aveek Purohit, Zheng Sun, Frank Mokaya, and Pei Zhang. Sensorfly: Controlled-mobile sensing platform for indoor emergency response applications. In *Information Processing in Sensor Networks (IPSN), 2011 10th International Conference on*, pages 223–234. IEEE, 2011. 5.1

[190] Aveek Purohit, Zheng Sun, Shijia Pan, and Pei Zhang. Sugartrail: Indoor navigation in retail environments without surveys and maps. In *Sensor, Mesh and Ad Hoc Communications and Networks (SECON), 2013 10th Annual IEEE Communications Society Conference on*, pages 300–308. IEEE, 2013. 5.1

[191] Aveek Purohit, Zheng Sun, and Pei Zhang. Sugarmap: Location-less coverage for micro-aerial sensing swarms. In *Information Processing in Sensor Networks (IPSN), 2013 ACM/IEEE International Conference on*, pages 253–264. IEEE, 2014. 5.1

[192] Ser-Tong Quek, Quan Wang, Liang Zhang, and Kian-Keong Ang. Sensitivity analysis of crack detection in beams by wavelet technique. *International journal of mechanical sciences*, 43(12):2899–2910, 2001. 2.2

[193] Christopher J Quinn, Todd P Coleman, Negar Kiyavash, and Nicholas G Hatsopoulos. Estimating the directed information to infer causal relationships in ensemble neural spike train recordings. *Journal of computational neuroscience*, 30(1):17–44, 2011. 5.1

[194] Carlos Marcelo Ramirez. *Building-specific loss estimation methods & tools for simplifiedperformance-based earthquake engineering*. Stanford University, 2009. 2.5.3, 3.5.1

[195] Arvind Rao, Alfred O Hero III, David J States, and James Douglas Engel. Using directed information to build biologically relevant influence networks. In *Computational Systems Bioinformatics: (Volume 6)*, pages 145–156. World Scientific, 2007. 2.4.1

[196] Jorge Ruiz-García and Carlos Chora. Evaluation of approximate methods to estimate residual drift demands in steel framed buildings. *Earthquake Engineering & Structural Dynamics*, 44(15):2837–2854, 2015. 2.1

[197] Jorge Ruiz-García and Eduardo Miranda. Probabilistic estimation of residual drift demands for seismic assessment of multi-story framed buildings. *Engineering Structures*, 32(1):11–20, 2010. 2.1

[198] C Scawthorn, H Iemura, and Y Yamada. Seismic damage estimation for low-and mid-rise buildings in japan. *Earthquake Engineering & Structural Dynamics*, 9(2):93–115, 1981. 3.3.1

[199] Bernhard Scholkopf and Alexander J Smola. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001. 2.5.3, 5.4.3

[200] Thomas Schreiber. Measuring information transfer. *Physical review letters*, 85(2):461, 2000. 2.2, 2.4.1, 5.4.3

[201] Dionysius M Siringoringo and Yozo Fujino. Estimating bridge fundamental frequency from vibration response of instrumented passing vehicle: analytical and experimental study. *Advances in Structural Engineering*, 15(3):417–433, 2012. 5.2

[202] Asim Smailagic, Hae Young Noh, Pedro Costa, Devesh Walawalkar, Kartik Khandelwal, Mostafa Mirshekari, Jonathon Fagert, Adrián Galdrán, and Susu Xu. Medal: Deep active

learning sampling method for medical image analysis. *arXiv preprint arXiv:1809.09287*, 2018. 3.4.3

[203] Hoon Sohn, Charles R Farrar, Francois M Hemez, Devin D Shunk, Daniel W Stinemates, Brett R Nadler, and Jerry J Czarnecki. A review of structural health monitoring literature: 1996–2001. *Los Alamos National Laboratory, USA*, 2003. 1.1

[204] Hoon Sohn, Charles R Farrar, Francois M Hemez, Devin D Shunk, Daniel W Stinemates, Brett R Nadler, and Jerry J Czarnecki. A review of structural health monitoring literature: 1996-2001, 2004. 1.1, 5.1

[205] Gangbing Song, Haichang Gu, and Yi-Lung Mo. Smart aggregates: multi-functional sensors for concrete structuresa tutorial and a review. *Smart materials and structures*, 17(3):033001, 2008. 2.1

[206] Zheng Song, Chi Harold Liu, Jie Wu, Jian Ma, and Wendong Wang. Qoi-aware multitask-oriented dynamic participant selection with budget constraints. *IEEE Transactions on Vehicular Technology*, 63(9):4618–4632, 2014. 4.2

[207] RJS Spence, AW Coburn, A Pomonis, and S Sakai. Correlation of ground motion with building damage: The definition of a new damage-based seismic intensity scale. In *Proceedings of the Tenth World Conference on Earthquake Engineering, Madrid, Spain*, volume 1, pages 551–56, 1992. 3.3.1

[208] Zheng Sun, Aveek Purohit, Shijia Pan, Frank Mokaya, Raja Bose, and Pei Zhang. Polaris: getting accurate indoor orientations for mobile devices using ubiquitous visual patterns on ceilings. In *Proceedings of the Twelfth Workshop on Mobile Computing Systems & Applications*, page 14. ACM, 2012. 5.1

[209] MM Reda Taha, A Noureldin, JL Lucero, and TJ Baca. Wavelet transform for structural health monitoring: a compendium of uses and features. *Structural Health Monitoring*, 5(3): 267–295, 2006. 5.4.2

[210] Maria I Todorovska and Mihailo D Trifunac. Impulse response analysis of the van nuys 7-storey hotel during 11 earthquakes and earthquake damage detection. *Structural Control and Health Monitoring: The Official Journal of the International Association for Structural Control and Monitoring and of the European Association for the Control of Structures*, 15 (1):90–116, 2008. 2.3, 2.3

[211] MD Trifunac and AG Brady. On the correlation of seismic intensity scales with the peaks of recorded strong ground motion. *Bulletin of the Seismological Society of America*, 65(1): 139–162, 1975. 3.3.1

[212] Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4068–4076, 2015. 3.2

[213] Eric Tzeng, Judy Hoffman, Kate Saenko, and Trevor Darrell. Adversarial discriminative domain adaptation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 7167–7176, 2017. 3.2, 3.4.4

[214] Dimitrios Vamvatsikos and C Allin Cornell. Incremental dynamic analysis. *Earthquake Engineering & Structural Dynamics*, 31(3):491–514, 2002. 3.1, 3.5.1

[215] Nicolas Veyrat-Charvillon and François-Xavier Standaert. Mutual information analysis: how, when and why? In *Cryptographic Hardware and Embedded Systems-CHES 2009*, pages 429–443. Springer, 2009. 2.2, 2.4.1, 5.4.3

[216] Xiumin Wang, Weiwei Wu, and Deyu Qi. Mobility-aware participant recruitment for vehicle-based mobile crowdsensing. *IEEE Transactions on Vehicular Technology*, 67(5): 4415–4426, 2018. 4.1, 4.2

[217] Zheng Wang, Yangqiu Song, and Changshui Zhang. Transferred dimensionality reduction. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 550–565. Springer, 2008. 3.2

[218] Michael Wibral, Raul Vicente, and Joseph T Lizier. *Directed information measures in neuroscience*. Springer, 2014. 2.4.1

[219] Fang-Jing Wu, Yu-Fen Kao, and Yu-Chee Tseng. From wireless sensor networks towards cyber physical systems. *Pervasive and Mobile computing*, 7(4):397–413, 2011. 4.2

[220] Tong Tong Wu, Kenneth Lange, et al. Coordinate descent algorithms for lasso penalized regression. *The Annals of Applied Statistics*, 2(1):224–244, 2008. 4.4.3

[221] Zhaohua Wu and Norden E Huang. Ensemble empirical mode decomposition: a noise-assisted data analysis method. *Advances in adaptive data analysis*, 1(01):1–41, 2009. 2.2

[222] He Xia, Nan Zhang, and YM Cao. Experimental study of train-induced vibrations of environments and buildings. *Journal of Sound and Vibration*, 280(3):1017–1029, 2005. 5.3, 5.4.1

[223] Susu Xu, Lin Zhang, Pei Zhang, and Hae Young Noh. An indirect traffic monitoring approach using building vibration sensing system. In *Proceedings of the 14th ACM Conference on Embedded Network Sensor Systems CD-ROM*, pages 374–375. ACM, 2016. 5.4.3

[224] Susu Xu, Lin Zhang, Pei Zhang, and Hae Young Noh. An information-theoretic approach for indirect train traffic monitoring using building vibration. *Frontiers in Built Environment*,

3:22, 2017. 3.1

[225] Susu Xu, Weiguang Mao, Yue Cao, Hae Young Noh, and Nihar B Shah. An incentive mechanism for crowd sensing with colluding agents. *arXiv preprint arXiv:1809.05161*, 2018. 4.2

[226] Dejun Yang, Guoliang Xue, Xi Fang, and Jian Tang. Incentive mechanisms for crowdsensing: crowdsourcing with smartphones. *IEEE/ACM Transactions on Networking*, 24(3): 1732–1744, 2016. 4.2

[227] Shuguan Yang and Zhen Sean Qian. Turning meter transactions data into occupancy and payment behavioral information for on-street parking. *Transportation Research Part C: Emerging Technologies*, 78:165–182, 2017. 4.1

[228] Y-B Yang, CW Lin, and JD Yau. Extracting bridge frequencies from the dynamic response of a passing vehicle. *Journal of Sound and Vibration*, 272(3-5):471–493, 2004. 5.2

[229] YB Yang and KC Chang. Extraction of bridge frequencies from the dynamic response of a passing vehicle enhanced by the emd technique. *Journal of sound and vibration*, 322(4-5): 718–739, 2009. 5.2

[230] YB Yang and CW Lin. Vehicle–bridge interaction dynamics and potential applications. *Journal of sound and vibration*, 284(1-2):205–226, 2005. 5.2

[231] Yeong-Bin Yang, JD Yau, Zhongda Yao, and YS Wu. *Vehicle-bridge interaction dynamics: with applications to high-speed railways*. World Scientific, 2004. 1.2

[232] Shih-Hsun Yin and Chung-Yu Tang. Identifying cable tension loss and deck damage in a cable-stayed bridge using a moving vehicle. *Journal of Vibration and Acoustics*, 133(2): 021007, 2011. 5.2

[233] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? In *Advances in neural information processing systems*, pages 3320–3328, 2014. 3.2

[234] Hae Young Noh, K Krishnan Nair, Dimitrios G Lignos, and Anne S Kiremidjian. Use of wavelet-based damage-sensitive features for structural damage diagnosis using strong motion data. *Journal of Structural Engineering*, 137(10):1215–1228, 2011. 1.1, 2.1, 3.1, 3.2, 5.4.2

[235] Mohamed Younis and Kemal Akkaya. Strategies and techniques for node placement in wireless sensor networks: A survey. *Ad Hoc Networks*, 6(4):621–655, 2008. 4.2

[236] Junyan Yu and Long Wang. Group consensus of multi-agent systems with directed information exchange. *International Journal of Systems Science*, 43(2):334–348, 2012. 2.3

[237] Ka-Veng Yuen and Lambros S Katafygiotis. Model updating using noisy response measurements without knowledge of the input spectrum. *Earthquake engineering & structural dynamics*, 34(2):167–187, 2005. 2.2

[238] Bianca Zadrozny. Learning and evaluating classifiers under sample selection bias. In *Proceedings of the twenty-first international conference on Machine learning*, page 114. ACM, 2004. 3.2

[239] Dimitri Zarzhitsky, Diana F Spears, William M Spears, and David R Thayer. A fluid dynamics approach to multi-robot chemical plume tracing. In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pages 1476–1477. IEEE Computer Society, 2004. 4.1, 4.1, 4.5.2

[240] Kun Zhang, Bernhard Schölkopf, Krikamol Muandet, and Zhikun Wang. Domain adaptation under target and conditional shift. In *International Conference on Machine Learning*, pages 819–827, 2013. 3.1

[241] Pei Zhang, Christopher M Sadler, Stephen A Lyon, and Margaret Martonosi. Hardware design experiences in zebranet. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 227–238. ACM, 2004. 5.1

[242] Feng Zhao, Jie Liu, Juan Liu, Leonidas Guibas, and James Reich. Collaborative signal and information processing: an information-directed approach. *Proceedings of the IEEE*, 91(8): 1199–1209, 2003. 2.4.1, 5.1

[243] Han Zhao, Shanghang Zhang, Guanhang Wu, José MF Moura, Joao P Costeira, and Geoffrey J Gordon. Adversarial multiple source domain adaptation. In *Advances in Neural Information Processing Systems*, pages 8559–8570, 2018. 3.2, 3.4.1, 3.4.4, 3.5.2

[244] Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J Gordon. On learning invariant representation for domain adaptation. *arXiv preprint arXiv:1901.09453*, 2019. 3.2

[245] Pengjun Zhao, Bin Lü, and Gert de Roo. Urban expansion and transportation: the impact of urban form on commuting patterns on the city fringe of beijing. *Environment and Planning A*, 42(10):2467–2486, 2010. 4.5.2

[246] Xuefeng Zhao, Niannian Wang, Ruicong Han, Botao Xie, Yan Yu, Mingchu Li, and Jinping Ou. Urban infrastructure safety system based on mobile crowdsensing. *International journal of disaster risk reduction*, 27:427–438, 2018. 1.1

[247] Yu Zheng, Furui Liu, and Hsun-Ping Hsieh. U-air: When urban air quality inference meets big data. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1436–1444. ACM, 2013. 4.1

[248] Yu Zheng, Tong Liu, Yilun Wang, Yanmin Zhu, Yanchi Liu, and Eric Chang. Diagnosing new york city's noises with ubiquitous data. In *Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing*, pages 715–725. ACM, 2014. 4.5.2

[249] Yu Zheng, Xiuwen Yi, Ming Li, Ruiyuan Li, Zhangqing Shan, Eric Chang, and Tianrui Li. Forecasting fine-grained air quality based on big data. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 2267–2276. ACM, 2015. 4.5.2

[250] Zhenzhe Zheng, Fan Wu, Xiaofeng Gao, Hongzi Zhu, Guihai Chen, and Shaojie Tang. A budget feasible incentive mechanism for weighted coverage maximization in mobile crowdsensing. *IEEE Transactions on Mobile Computing*, PP(99):1–1, 2017. 4.2

Appendix A

Supplementary Information for Chapter 4

A.1 Proof of Theorem 1

Proof. The incentive $B(c)$ is defined as follows

$$B(c) = \max(r_{\min}, \min(r_{\max}, r_{\max} - r_u(R^c_{ctrl} - R^c_{rand}))).$$

Where

$$R^c_{rand} = \sum_{i,j}^{a,b} Re(i,j,T) Pr_c(i,j,T)$$
$$R^c_{ctrl} = \sum_{i,j}^{a,b} Re(i,j,T) D_c(i,j,T).$$

Case 1: When $R^c_{ctrl} < R^c_{rand}$, that is, when the incentivizing assignment does not improve the probability of task request, we will pay $B(c) = r_{\max}$. The $r_{\max}$ equals to the revenue obtained by taking a general task request. In general situation, the amount of $r_{\max}$ is determined by the market and represents the lower threshold that a vehicle agent is willing to take the task, which means that the seller side and buyer side in the market, including the vehicle agent and the task request end (e.g. passenger or the crowdsourcer), all agree to accept/pay $r_{\max}$ to finish the task for time length of T.

Case 2: When $R^c_{ctrl} \geq R^c_{rand}$, to simplify the problem, here we assume the vehicle agents focus on the profit during incentivizing period T and the profit in the first time point after the incentivizing period, $T + 1$. That is, from the perspective of vehicle agents, at the beginning time point $t = 1$ before one incentivizing period, the estimation of the profit is based on the time period

from $t = 1$ to $t = T + 1$. If an unoccupied vehicle agent rejects the incentivizing assignment, since there is no task nor assignment now, the profit is the expected revenue from $t = 1$ to $t = T + 1$ subtracting the cost. For each free vehicle agent, the expected revenue at time point $t + 1$ is based on the task request obtained at time t, which is

$$\sum_{i,j}^{a,b} r_u Re(i, j, t) Pr_c(i, j, t).$$

Meanwhile, from the perspective of vehicle agents, the importance of future revenue degrades with time. We assume this degrading coefficient is γ, where $0 \leq \gamma \leq 1$. The effect of future revenue on current estimation of utility degrade in a multiplicative way. Thus, if rejecting the incentivizing assignment D_c, the utility for vehicle agent c is

$$\begin{aligned} &u(c \text{ rejects } D_c) \\ &= \sum_{t=1}^{T+1} \sum_{i,j}^{a,b} \gamma^{t-1} r_u Re(i, j, t-1) Pr_c(i, j, t-1). \end{aligned}$$

For incentivized vehicle agents, $B(c)$ is the revenue from the beginning to the end of incentivizing period. Since during $t = 1$ to $t = T$, the revenue is deterministic and known, the expected revenue from $t = 1$ to $t = T$ is $B(c)/T$. After T, task request is no longer deterministic for the vehicle agent, and the expected revenue depends on the destination of the incentivizing assignment D_c. Thus, if accepting the incentivizing assignment, the utility of vehicle agent c is

$$\begin{aligned} &u(c \text{ accepts } D_c) \\ &= \sum_{t=1}^{T} \gamma^{t-1} B(c)/T + \sum_{i,j}^{a,b} \gamma^{T+1-1} r_u Re(i, j, T) D_c(i, j, T). \end{aligned}$$

Compared to rejecting the incentivizing assignment, the cost from the beginning of incentivizing period does not change, which incurs regardless of accepting the assignment or not. Assuming all vehicle agents are rational, a vehicle agent c will accept an incentivizing assignment D_c if and only if

$$u(c \text{ rejects } D_c) \leq u(c \text{ accepts } D_c). \tag{A.1}$$

By substituting the definition of $u(c \text{ rejects } D_c)$ and $u(c \text{ accepts } D_c)$, the Inequation A.1 is equiv-

alent to

$$
\begin{aligned}
& B(c) \cdot (\sum_{t=1}^{T} \gamma^{t-1})/T \\
& \geq \sum_{t=1}^{T+1} \sum_{i,j}^{a,b} \gamma^{t-1} r_u Re(i,j,t-1) Pr_c(i,j,t-1) \\
& \quad - \sum_{i,j}^{a,b} \gamma^{T} r_u Re(i,j,T) D_c(i,j,T) \\
& = \sum_{t=1}^{T} \sum_{i,j}^{a,b} \gamma^{t-1} r_u Re(i,j,t) Pr_c(i,j,t) + \gamma^{T} r_u (R^c_{rand} - R^c_{ctrl}).
\end{aligned}
$$

Since $\forall t$, there is always

$$
\sum_{i,j}^{a,b} Pr_c(i,j,t) = 1, Re(i,j,t) \leq 1.
$$

Thus, we can have $\forall t$,

$$
\begin{aligned}
& \sum_{i,j}^{a,b} Re(i,j,t) Pr(i,j,t) \leq 1 \\
\Rightarrow & \sum_{i,j}^{a,b} r_u Re(i,j,t) Pr(i,j,t) \leq r_u \\
\Rightarrow & \sum_{t=1}^{T} \sum_{i,j}^{a,b} \gamma^{t-1} r_u Re(i,j,t-1) Pr(i,j,t-1) \leq r_u \sum_{t=1}^{T} \gamma^{t-1}.
\end{aligned}
$$

Also, since $\gamma \in [0,1]$, and $T \geq 1$, there is $\forall t \leq T$

$$
\gamma^{t-1} \geq \gamma^{T} \Rightarrow \sum_{t=1}^{T} \gamma^{t-1} \geq T\gamma^{T} \Rightarrow \gamma^{T} \leq (\sum_{t=1}^{T} \gamma^{t-1})/T
$$

Comparing with $B(c)$,

$$\begin{aligned}
&\sum_{t=1}^{T}\sum_{i,j}^{a,b}\gamma^{t-1}r_u Re(i,j,t)Pr_c(i,j,t)+\gamma^T r_u(R^c_{rand}-R^c_{ctrl})\\
\leq& r_u\sum_{t=1}^{T}\gamma^{t-1}+\gamma^T r_u(R^c_{rand}-R^c_{ctrl})\\
\leq& r_u T\cdot(\sum_{t=1}^{T}\gamma^{t-1})/T+(\sum_{t=1}^{T}\gamma^{t-1})/T\cdot r_u(R^c_{rand}-R^c_{ctrl})\\
=&[r_{\max}-r_u(R^c_{ctrl}-R^c_{rand})]\cdot(\sum_{t=1}^{T}\gamma^{t-1})/T\\
\leq& B(c)\cdot(\sum_{t=1}^{T}\gamma^{t-1})/T \text{ when } R^c_{ctrl}\geq R^c_{rand}.
\end{aligned}$$

Therefore, when $R^c_{ctrl}\geq R^c_{rand}$, the Ineuqation A.1 is always satisfied. That is, the utility obtained by accepting the incentivizing assignment is always larger or equal to the utility obtained by rejecting the assignment. When the vehicle agent is rational, the vehicle agent will always choose to accept the assignment. □

A.2 Proof of Lemma 1

Proof. In the formal definition of multiple choice knapsack problem, there are m classes $N_l,\cdots,N_m$ of items to be packed into a knapsack of capacity β. Each item $j\in N_i$ has a profit value v_{ij} and a size W_{ij}, and the problem is to choose exactly one item from each class such that the total value $\sum_{i,j}f(v_{ij})x_{ij}$ is minimized/maximized without exceeding the capacity. The mathematical formulation is

$$\begin{aligned}
&\max_{x_{ij}}\sum_{i=1}^{m}\sum_{j=1}^{N_i}f(v_{ij})x_{ij}\\
&s.t.\sum_{i=1}^{m}\sum_{j=1}^{N_i}W_{ij}x_{ij}\leq\beta\\
&\sum_{j=1}^{N_i}x_{ij}=1\\
&x_{ij}\in\{0,1\}
\end{aligned}$$

In the classic linear multiple choice knapsack problem, the value function is linear which means $f(v_{ij}) = v_{ij}$. If $f(\cdot)$ is a non-linear function of v_{ij}, the problem is a non-linear multiple choice knapsack problem.

Similarly in our problem, the capacity constraint refers to the total budget B. We have C classes, which is the C vehicle agents. Each class has K_c items, which belongs to the candidate trajectories set $\{D_c^k : 1 \leq k \leq K_c\}$. If a vehicle agent c is occupied, its candidate trajectory set only has one item, which is the trajectory determined by the passenger. If a vehicle agent c is unoccupied, its candidate trajectory include deterministic trajectories given starting location and incentivizing period and one probabilistic trajectory which is calculated based on the mobility transition matrix and represents cruising without any passengers or incentivization. When a vehicle agent is neither incentivized nor occupied, it still keeps sensing data. Therefore, choosing not to incentivize an unoccupied vehicle agent is equivalent to choosing to let the vehicle agent run as usual. Each item has a size $B(c,k)$, which is the corresponding cost determined by the vehicle agent c and respective trajectory k. $B(c,k)$ is obtained by the cost $B(c)$ of incentivizing the vehicle agent c to its kth trajectory. The value of each item is a vector v_{ck}

$$v_{ck} = [\frac{D_c^k(i=1,j=1,t=1)}{CT}, \cdots, \frac{D_c^k(i=a,j=b,t=T)}{CT}]$$

Our problem can be stated as: given C classes, each class has K_c items, each item $D_c^k \in \{D_c^k : 1 \leq k \leq K_c\}$ has a profit value v_{ck} and size $B(c,k)$, under the capacity constraint B, selecting exactly one item from each class to maximize the negative ***KL-divergence***. The mathematical formulation in the format of multiple choice knapsack problem is

$$\begin{aligned}
&\max_{x_{ck}} \sum_{i,j,t} P(i,j,t) \log \frac{O(i,j,t)}{P(i,j,t)} \\
&\text{where } P(i,j,t) = \sum_{c=1}^{C} \sum_{k=1}^{K_c} \frac{D_c^k(i,j,t)}{CT} x_{ck} \\
&s.t. \sum_{c=1}^{C} \sum_{k=1}^{K_c} B(c,k) x_{ck} \leq B \\
&\sum_{c=1}^{C} \sum_{k=1}^{K_c} x_{ck} = 1 \\
&x_{ck} \in \{0,1\}
\end{aligned}$$

Since different vehicle agent's final selected trajectories can be overlapped, the objective function is a non-separable nonlinear function with respect to each item D_c^k. Therefore, our problem is a

non-linear multiple choice knapsack problem.

□

A.3 Proof of Theorem 2

Proof. The classic linear knapsack problem, quadratic knapsack problem, linear multiple-choice knapsack problem are proved to be NP-complete. In our problem, the objective function involves with integrated logarithmic of integrated variables, which makes it more difficult to prove the NP-hardness. Although proving the NP-hardness is beyond our research scope in this study, here we give a proof of the NP-completeness of the decision version of our problem. The decision version of our problem can be stated as: *Does there exist a vehicle incentivizing solution such that the* KL-divergence *between integrated sensing data distribution and target distribution is smaller than a specific value h while the constraints are satisfied?* To show the NP-completeness, we firstly showed that the problem is NP, then showed that it is NP-hard. The proof is as follows.

To show the problem is NP, given a vehicle incentivizing solution $\mathcal{S}$ for C vehicles, we can validate the correctness of the solution in polynomial time. In detail, we can calculate the *KL-divergence* by aggregating all vehicles' trajectory to obtain the final collected data distribution and thus get the *KL-divergence* with the time complexity of $\mathcal{O}(C)$.

To show the problem is NP-hard, we show a special case of the problem is equivalent to a known NP-hard problem: linear multiple-choice knapsack problem. The special case is defined as: in the given candidate trajectories, the trajectories of any two different vehicles are not overlapped at any time point t where $1 \leq t \leq T$. Mathematically, non-overlapping means that $\forall c_1, c_2$ with the sets of candidate trajectories $\{D_1^k\}$ and $\{D_2^k\}$, we have $\forall D_1^{k_1} \in \{D_1^k\}$ and $D_2^{k_2} \in \{D_2^k\}$, and $\forall 1 \leq i \leq a, 1 \leq j, \leq b, 1 \leq t \leq T$,

$$D_1^{k_1}(i,j,t) \cdot D_2^{k_2}(i,j,t) = 0$$

Since any two vehicles' trajectories are non-overlapping, the objective function can now be separable by different vehicle c, which means that,

$$\begin{aligned} &KL(P||O) \\ &= -\sum_{c} \sum_{i,j,t} \frac{D_c(i,j,t)}{CT} [\log(O(i,j,t) \cdot CT) - \log D_c(i,j,t)] \end{aligned}$$

The special case is much simpler to solve than our original problem where the candidate trajectories could be overlapped. Meanwhile, this special case does belong to our original problem.

Therefore, if we can prove the special case is NP-hard, it is reasonable to claim that our problem is NP-hard.

In this special case, the problem is: Given C vehicles' candidate trajectory sets $\{D_c^k : 1 \leq k \leq K_c\}$ where $c = 1, \cdots, C$, can we select exactly one trajectory from each set such that the obtained ***KL-divergence*** between collected data distribution and target distribution is less than an exact value h,

$$KL(P||O) \leq h$$

while satisfying the budget constraints.

This problem is equivalent to a classic multiple choice knapsack problem: given C classes where each class has items $\{D_c^k : 1 \leq k \leq K_c\}$, each item in each class has a profit value v_{ck} where,

$$v_{ck} = \sum_{i,j,t} \frac{D_c^k(i,j,t)}{CT} [\log(O(i,j,t) \cdot CT) - \log D_c^k(i,j,t)]$$

Meanwhile each item has its own size which is $B(c,k)$, can we select one item from each class, where selecting the item k of class c means $x_{ck} = 1$ and vice versa, such that the total value $\sum_{c=1}^{C} \sum_{k=1}^{K_c} v_{ck} x_{ck} \geq h$ while the total capacity $\sum_{c=1}^{C} \sum_{k=1}^{K_c} B(c,k) x_{ck}$ does not exceed B.

Note that the value v_{ck} of an item D_c^k is only determined by the given item and the given fixed target distribution $O(i,j,t)$. $B(c,k)$ is also only determined by the class and the item itself, and represents the cost $B(c)$ of incentivizing the vehicle c to its kth trajectory in our problem.

The classic linear multiple choice knapsack problem is already proved to be NP-hard. Since our problem under a special setting is equivalent to the classic linear multiple choice knapsack problem, the general case of our problem is also NP-hard.

Therefore, the decision version of our problem is NP-complete. □